

Jan Lunze

**Automatisierungstechnik**

De Gruyter Studium

## Weitere empfehlenswerte Titel



*Künstliche Intelligenz für Ingenieure – Methoden zur Lösung ingenieurtechnischer Probleme mit Hilfe von Regeln, logischen Formeln und Bayesnetzen*

Jan Lunze, 2016

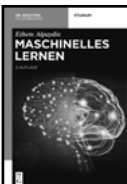
ISBN 978-3-11-044896-2, e-ISBN 978-3-11-044897-9



*Ereignisdiskrete Systeme – Modellierung und Analyse dynamischer Systeme mit Automaten, Markovketten und Petrinetzen*

Jan Lunze, 2017

ISBN 978-3-11-048467-0, e-ISBN 978-3-11-048471-7



*Maschinelles Lernen*

Ethem Alpaydin, 2019

ISBN 978-3-11-061788-7, e-ISBN 978-3-11-061789-4



*Communication in Vehicles – Cultural Variability in Speech Systems*

Brion Van Over, Ute Winter, Elizabeth Molina-Markham, Sunny Lie, Donal Carbaugh, 2020

ISBN 978-3-11-051891-7, e-ISBN 978-3-11-051900-6



*Signale und Systeme*

Fernando Puentes León, Holger Jäkel, 2019

ISBN 978-3-11-062631-5, e-ISBN 978-3-11-062632-2

Jan Lunze

# Automatisierungstechnik

---

Methoden für die Überwachung und Steuerung  
kontinuierlicher und ereignisdiskreter Systeme

5., überarbeitete Auflage

mit 413 Abbildungen, 94 Anwendungsbeispielen  
und 100 Übungsaufgaben

**DE GRUYTER**  
OLDENBOURG

**Autor**

Prof. Dr.-Ing. Jan Lunze  
Ruhr-Universität Bochum  
Lehrstuhl für Automatisierungstechnik und Prozessinformatik  
44780 Bochum  
email: Lunze@atp.rub.de

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. The MathWorks Publisher Logo identifies books that contain MATLAB and Simulink content. Used with permission. The MathWorks does not warrant the accuracy of the text or exercises in this book. This book's use or discussion of MATLAB and Simulink software or related products does not constitute endorsement or sponsorship by The MathWorks of a particular use of the MATLAB and Simulink software or related products.

For MATLAB® and Simulink® product information, or information on other related products, please contact:

The MathWorks, Inc.  
3 Apple Hill Drive  
Natick, MA, 01760-2098 USA  
Tel: 508-647-7000  
Fax: 508-647-7001  
E-mail: [info@mathworks.com](mailto:info@mathworks.com)  
Web: [www.mathworks.com](http://www.mathworks.com)

ISBN 978-3-11-068072-0  
e-ISBN (PDF) 978-3-11-068352-3  
e-ISBN (EPUB) 978-3-11-068357-8

**Library of Congress Control Number: 2020931595**

**Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.dnb.de> abrufbar.

© 2020 Walter de Gruyter GmbH, Berlin/Boston  
Umschlagabbildung: AleksandarGeorgiev/E+/gettyimages.de  
Druck und Bindung: CPI books GmbH, Leck

[www.degruyter.com](http://www.degruyter.com)



# Vorwort

Die Automatisierungstechnik zeichnet sich durch eine große Methodenvielfalt und ein breites Anwendungsfeld aus. Neben der Steuerung und Regelung dynamischer Prozesse haben die Prozessüberwachung und die Fehlerdiagnose in den letzten Jahren eine zunehmende Bedeutung erlangt. Da diese Methoden auf einer systemtheoretischen Betrachtung des zu automatisierenden Prozesses beruhen, sind sie in allen ingenieurtechnischen Gebieten und darüber hinaus in vielen nichttechnischen Bereichen einsetzbar. Verfahrenstechnische Prozesse, die Energieversorgung, fahrzeug- und verkehrstechnische Systeme, Fertigungsprozesse und die Gebäudetechnik sind ohne Automatisierungstechnik undenkbar.

Dieses Buch gibt eine breite Einführung in die grundlegenden Aufgaben und Methoden der Automatisierungstechnik und veranschaulicht diese an zahlreichen Anwendungsbeispielen. Die Palette der Themen reicht von der Modellbildung über die Vorhersage des zukünftigen Systemverhaltens, den Entwurf von Regelungen und Steuerungen bis zur Zustandsbeobachtung und Prozessdiagnose.

Mit der gleichwertigen Behandlung kontinuierlicher und ereignisdiskreter Systeme betritt das Buch Neuland. Der Autor kennt kein vergleichbares Lehrbuch, das wie dieses alle wichtigen Automatisierungsaufgaben für beide Systemklassen betrachtet.

Bisher wurden die Methoden für die Überwachung und Regelung kontinuierlicher Systeme weitgehend getrennt von denen für ereignisdiskrete Systeme entwickelt und gelehrt. Mit der Einführung der Zustandsraumdarstellung entstand bereits um 1960 die Grundlage für eine durchgängige Theorie kontinuierlicher Systeme, während die breite Entwicklung automatisierungstechnischer Methoden für ereignisdiskrete Systeme erst nach 1980 einsetzte und deshalb noch nicht so weit gediehen ist. Dennoch ist für das Verständnis der Automatisierungstechnik mit ihrem umfangreichen Methodenspektrum und breiten Anwendungsgebiet die weitgehend gemeinsame Behandlung beider Systemklassen unabdingbar. Dieses Buch soll einen Anstoß geben, beide Teilgebiete besser als bisher in der Lehre zu verknüpfen.

**Inhalt.** Das Buch ist aus dem Manuskript einer Pflichtvorlesung für den Studiengang Elektrotechnik und Informationstechnik der Ruhr-Universität Bochum entstanden. Die Lehrveranstaltung „Automatisierungstechnik“ ersetzt seit dem Wintersemester 2001 die frühere Vorlesung „Regelungstechnik“, die sich wie ähnliche Pflichtveranstaltungen ingenieurtechnischer Studiengänge an anderen Universitäten im Wesentlichen mit dem einschleifigen linearen Regelkreis beschäftigte. Wichtige regelungstechnische Begriffe und Methoden wie das Zustandsraummodell, die Stabilitätsanalyse rückgekoppelter Systeme und Einstellregeln für PID-Regler werden auch hier behandelt. Darüber hinaus gibt das Buch aber auch eine Einführung in die Zustandsbeobachtung und die Fehlerdiagnose kontinuierlicher Systeme, die traditionell erst in späteren Lehrveranstaltungen geboten werden bzw. bisher überhaupt noch nicht in die Lehre

eingeflossen sind. Die genannten Themen werden anschließend für ereignisdiskrete Systeme behandelt, wobei offensichtlich wird, dass die Automatisierungsaufgaben für beide Systemklassen sehr ähnlich sind, sich die Lösungsmethoden aber auf Grund des unterschiedlichen Charakters beider Systemklassen wesentlich voneinander unterscheiden.

Dieser Stoffauswahl entsprechend ist das Buch in drei Teile gegliedert:

### 1. Einführung

Im ersten Teil werden die Ziele und Aufgaben der Automatisierungstechnik behandelt und Methoden für die strukturelle Analyse dynamischer Systeme eingeführt. Ein Grundprinzip der Automatisierungstechnik besteht in der Rückkopplung von Informationen. Hierzu wird der Unterschied zwischen den Steuerungen in der offenen Wirkungskette und im geschlossenen Wirkungskreis erläutert. Alle im ersten Teil behandelten Methoden sind unabhängig davon, ob es sich bei den betrachteten Systemen um kontinuierliche oder ereignisdiskrete handelt.

### 2. Automatisierung kontinuierlicher Systeme

Der zweite Teil befasst sich mit der Automatisierung von Systemen, deren Verhalten durch reellwertige Signale dargestellt und die deshalb durch Differentialgleichungen beschrieben werden. Der gemeinsame Ausgangspunkt für die Lösung aller Automatisierungsaufgaben ist das Zustandsraummodell des zu automatisierenden Prozesses, das für lineare und nichtlineare Systeme behandelt wird. Nach der Verhaltensanalyse kontinuierlicher Systeme werden wichtige Systemeigenschaften wie die Steuerbarkeit, die Beobachtbarkeit und die Stabilität eingeführt, Verfahren für die Reglereinstellung erläutert und schließlich Methoden für die Zustandsbeobachtung und die Fehlerdiagnose behandelt.

### 3. Automatisierung ereignisdiskreter Systeme

Der dritte Teil beschäftigt sich mit der Automatisierung von Systemen, deren Verhalten durch Ereignisfolgen beschrieben wird. Als Modellformen werden deterministische, nicht-deterministische und stochastische Automaten sowie Petrinetze eingeführt und für die Verhaltensanalyse, den Steuerungsentwurf, die Zustandsbeobachtung sowie die Fehlerdiagnose eingesetzt.

Diese Stoffauswahl verfolgt zwei Ziele. Einerseits ist das Buch für Studenten gedacht, die eine breite Einführung in das Gebiet der Automatisierungstechnik erhalten sollen, um ihre spätere Zusammenarbeit mit Fachleuten der Automatisierungstechnik vorzubereiten. Für sie ist Spezialwissen über Regelkreise nur von zweitrangiger Bedeutung. Dies gilt auch für Studenten der gegenwärtig neu entstehenden Studiengänge mit ingenieurtechnischer Ausrichtung wie z. B. Informatikingenieurwesen, Informationssystemtechnik, Technomathematik oder Wirtschaftsingenieurwesen.

Andererseits soll der Stoff dieses Buches eine breite Basis für das Vertiefungsstudium der Automatisierungstechnik schaffen. Die darauf aufbauenden Spezialvorlesungen umfassen in Bochum Pflichtveranstaltungen, die nach den bereits erschienenen Lehrbüchern<sup>1</sup> des Autors das regelungstechnische Fachwissen bzw. die Kenntnisse über die Beschreibung und Analyse ereignisdiskreter Systeme vertiefen.

<sup>1</sup> J. Lunze: *Regelungstechnik* (2 Bände), Springer-Verlag;

J. Lunze: *Ereignisdiskrete Systeme*, de Gruyter Oldenbourg

Die Automatisierungstechnik ist eine Methodenwissenschaft. Deshalb konzentriert sich das Buch auf die methodischen Grundlagen und behandelt gerätetechnische und rechentechnische Probleme der Realisierung von Überwachungs- und Steuereinrichtungen nur am Rande.

Jede Automatisierungsaufgabe soll ein konkretes technisches Problem lösen. Trotz vieler Unterschiede in den Details kann man typische Aufgabenklassen angeben, deren Lösungsmethoden hier in Verbindung mit einer breiten Palette von **Anwendungsbeispielen** (in einer für die Lehre notwendigen Vereinfachung) erläutert werden. Dabei wird auch das große Anwendungspotenzial automatisierungstechnischer Methoden gezeigt.

Die angeführte Breite des Stoffes kann in einer einsemestrigen Lehrveranstaltung nur auf Kosten der Tiefe vermittelt werden. Die Erfahrungen des Autors zeigen, dass es durch eine Konzentration auf die wichtigsten Fragestellungen möglich ist, alle angesprochenen Themen in jeweils ein bis drei Vorlesungsstunden zu behandeln. Da durch das Zustandsraummodell in seiner kontinuierlichen und diskreten Form eine Brücke zwischen den beiden Systemklassen geschlagen wird, mit der sämtliche Automatisierungsaufgaben formuliert und gelöst werden können, entsteht dabei ein breites, zusammenhängendes Bild der Automatisierungstechnik.

**Leser.** Bei den Lesern werden gute Kenntnisse auf den Gebieten der Matrizenrechnung und der linearen gewöhnlichen Differentialgleichungen vorausgesetzt. Die Systeme werden ausschließlich im Zeitbereich behandelt, so dass keine Kenntnisse über die Funktionaltransformationen erforderlich sind. Die erwarteten Vorkenntnisse aus der diskreten Mathematik beschränken sich auf Grundbegriffe der Graphentheorie, der Mengenlehre und der Wahrscheinlichkeitsrechnung.

Das Buch wendet sich auch an die in der Praxis stehenden Fachleute, die eine fundierte Einführung in die methodischen Grundlagen der Automatisierungstechnik suchen. Zahlreiche **Übungsaufgaben** mit ausführlichen Lösungen geben Anregungen für das selbstständige Erarbeiten des Stoffes. Alle Kapitel schließen mit einem Ausblick auf weiterführende Themen und mit Literaturhinweisen. Die Zusammenstellung wichtiger englischer Fachbegriffe im Anhang soll den Einstieg in die Fachliteratur erleichtern.

**Danksagung.** Die Idee, kontinuierliche und ereignisdiskrete Systeme in weitgehender Analogie zu behandeln, verfolgt mich seit meinem Studium an der Technischen Universität Ilmenau, bei dem ich durch eine Vorlesung von Prof. Dr. MICHAEL KRAPP in die Automatentheorie eingeführt wurde. Aufgrund dieser Lehrveranstaltung habe ich die kontinuierliche Sicht auf Regelungs- und Steuerungssysteme nie als die einzig mögliche Betrachtungsweise angesehen und hatte schon lange geplant, eine möglichst parallele Behandlungsweise beider Systemklassen in die Lehre einzuführen.

Gemeinsam mit meinem ehemaligen Mitarbeiter Dr.-Ing. JOCHEN SCHRÖDER habe ich mehrere Jahre lang untersucht, wie man die Probleme der Zustandsbeobachtung und der Prozessdiagnose für Automaten lösen kann. Die dabei erarbeiteten Methoden haben eine Lücke geschlossen und eine durchgängige Behandlung aller Automatisierungsaufgaben für kontinuierliche und ereignisdiskrete Systeme möglich gemacht. Der häufig plakativ gebrauchte Hinweis auf die Einheit von Forschung und Lehre hat hier seine Berechtigung.

Die Herren Dr.-Ing. JÖRG NEIDIG und Dr.-Ing. PHILIPP PLANCHON haben wesentlich bei der Einführung der Lehrveranstaltung „Automatisierungstechnik“ in Bochum mitgewirkt und wertvolle Hinweise für die Gestaltung der Übungsaufgaben gegeben. Schließlich gilt mein

Dank Frau ANDREA MARSCHALL für das Zeichnen vieler Bilder sowie dem Verlag De Gruyter Oldenbourg für die schnelle Herausgabe dieses Lehrbuches.

**Fünfte Auflage.** Die Überarbeitung des Textes für die neue Auflage führte auf viele Verbesserungen der Bilder und der Erläuterungen. In den Übungsaufgaben und Beispielen wurden neue Ideen meines Doktoranden M. Sc. MICHAEL SCHWUNG aufgenommen, der in den letzten Jahren die Lehrveranstaltung zu diesem Buch betreut hat. Die für Bilder verwendeten MATLAB-Skripte wurden der aktuellen Version (Release R2019b) angepasst.

Münster, im Januar 2020

JAN LUNZE

# Inhaltsverzeichnis

<b>Verzeichnis der Anwendungsbeispiele</b> .....	xvii
<b>Hinweise zum Gebrauch des Buches</b> .....	xxv

## Teil 1: Einführung

<b>1 Ziele und Aufgaben der Automatisierungstechnik</b> .....	3
1.1 Ziele der Automatisierungstechnik .....	3
1.2 Anwendungsbeispiele .....	6
1.2.1 Prozessautomatisierung .....	6
1.2.2 Fertigungsautomatisierung .....	11
1.2.3 Gebäudeautomatisierung .....	12
1.2.4 Überwachung und Steuerung von Energiesystemen .....	13
1.2.5 Automatisierungstechnik in Fahrzeugen .....	14
1.2.6 Überwachung und Steuerung des Flugverkehrs .....	16
1.2.7 Automatisierungsaufgaben in der Informations- und Kommunikationstechnik .....	17
1.2.8 Zusammenfassung: Notwendigkeit der Automatisierung technischer Systeme .....	18
1.3 Grundstruktur automatisierter Systeme .....	19
1.3.1 Beziehungen zwischen der Automatisierungseinrichtung und dem zu automatisierenden Prozess .....	19
1.3.2 Das Rückkopplungsprinzip .....	21
1.3.3 Die Rolle des Menschen in automatisierten Systemen .....	22
1.3.4 Spezielle Klassen automatisierter Systeme .....	23
1.4 Automatisierungsaufgaben .....	25
1.4.1 Modellbildung dynamischer Systeme .....	26
1.4.2 Vorhersage des Systemverhaltens .....	28
1.4.3 Planung von Steuereingriffen .....	29
1.4.4 Zustandsbeobachtung .....	30
1.4.5 Prozessdiagnose .....	31
1.4.6 Regelung und Steuerung .....	32

1.4.7	Kombination von Automatisierungsfunktionen.....	35
1.4.8	Automatisierungshierarchie.....	36
1.5	Realisierung von Automatisierungseinrichtungen.....	40
1.5.1	Methoden und Geräte.....	40
1.5.2	Lösungsweg für Automatisierungsaufgaben.....	41
1.5.3	Beziehungen zwischen der Automatisierungstechnik und angrenzenden Fachdisziplinen.....	42
	Literaturhinweise.....	43
<b>2</b>	<b>Grundlegende Eigenschaften dynamischer Systeme.....</b>	<b>45</b>
2.1	Grundbegriffe der Systemtheorie.....	45
2.1.1	Signal, Prozess, System.....	45
2.1.2	Kontinuierliche und diskrete Signale und Systeme.....	47
2.1.3	Statische und dynamische Systeme.....	50
2.1.4	Autonome und gesteuerte Systeme.....	52
2.1.5	Gemeinsamkeiten und Unterschiede kontinuierlicher und diskreter Systeme.....	53
2.2	Blockschaltbild und Signalflussgraph.....	54
2.2.1	Struktur dynamischer Systeme.....	54
2.2.2	Blockschaltbild.....	55
2.2.3	Signalflussgraph.....	62
2.3	Dekomposition und Aggregation von Systemen.....	64
2.4	Kopplungsanalyse.....	66
2.5	Steuerungen in der offenen Wirkungskette und im geschlossenen Kreis.....	72
	Literaturhinweise.....	80

## Teil 2: Automatisierung kontinuierlicher Systeme

<b>3</b>	<b>Beschreibung kontinuierlicher Systeme.....</b>	<b>83</b>
3.1	Modellbildungsaufgabe.....	83
3.2	Systembeschreibung durch lineare Differentialgleichungen.....	84
3.3	Zustandsraummodell linearer Systeme.....	90
3.3.1	Zustandsgleichung und Ausgabegleichung.....	90
3.3.2	Zustandsbegriff.....	93
3.3.3	Normierung der Signale und Parameter.....	95
3.3.4	Blockschaltbild und Signalflussgraph des Zustandsraummodells.....	97
3.3.5	Zustandsraumdarstellung von Mehrgrößensystemen.....	99
3.3.6	Gleichgewichtszustand linearer Systeme.....	100
3.4	Zustandsraummodell nichtlinearer Systeme.....	102
3.5	Linearisierung.....	111
3.6	Kompositionale Modellbildung kontinuierlicher Systeme.....	116
	Literaturhinweise.....	121

<b>4</b>	<b>Verhalten kontinuierlicher Systeme</b>	123
4.1	Vorhersage des Systemverhaltens	123
4.2	Verhalten linearer Systeme	124
4.2.1	Lösung der Zustandsgleichung	124
4.2.2	Berechnung der Ausgangsgröße	127
4.2.3	Anmerkungen zum Zustandsbegriff	133
4.2.4	Bewegungsgleichung in kanonischer Form	136
4.2.5	Übergangsfunktion	141
4.2.6	Berechnung der Matrixexponentialfunktion	143
4.3	Verhalten nichtlinearer Systeme	144
4.3.1	Lösung der nichtlinearen Modellgleichungen	144
4.3.2	Rechnergestützte Analyse nichtlinearer Systeme	146
4.4	Kennwertermittlung	151
	Literaturhinweise	153
<b>5</b>	<b>Steuerbarkeit und Beobachtbarkeit</b>	155
5.1	Definition der Steuerbarkeit und der Beobachtbarkeit	155
5.2	Steuerbarkeit linearer Systeme	157
5.2.1	Steuerbarkeitskriterium	157
5.2.2	Eigenschaften vollständig steuerbarer Systeme	158
5.3	Beobachtbarkeit linearer Systeme	163
5.3.1	Beobachtbarkeitskriterium	163
5.3.2	Berechnung des Anfangszustands aus $n$ Messwerten	164
5.3.3	Bestimmung des Anfangszustands mit Hilfe der gramscen Beobachtbarkeitsmatrix	166
5.4	Strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit	168
5.4.1	Strukturgraph	168
5.4.2	Definition und Kriterien für die strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit	170
5.4.3	Strukturelle Analyse nichtlinearer Systeme	175
5.5	Systemzerlegung entsprechend den Steuerbarkeits- und Beobachtbarkeitseigenschaften	176
	Literaturhinweise	180
<b>6</b>	<b>Stabilität</b>	181
6.1	Stabilitätsdefinition	181
6.2	Stabilitätsanalyse linearer Systeme	184
6.2.1	Stabilitätsanalyse anhand der Eigenwerte der Systemmatrix	184
6.2.2	Hurwitzkriterium	187
6.3	Stabilitätsanalyse nichtlinearer Systeme	190
6.3.1	Lösungswege	190
6.3.2	Stabilitätsprüfung mit dem linearisierten Modell	192
6.3.3	Direkte Methode von Ljapunow	195
6.3.4	Anwendung der Direkten Methode auf lineare Systeme	201

6.4	Stabilität von Regelkreisen .....	203
6.4.1	Stabilität der Regelstrecke und des Regelkreises .....	203
6.4.2	Robuste Stabilität .....	207
6.5	Ausblick: Weitere Verfahren für die Stabilitätsprüfung .....	209
	Literaturhinweise .....	209
<b>7</b>	<b>Einschleifige Regelkreise .....</b>	<b>211</b>
7.1	Regelungsaufgabe für kontinuierliche Systeme .....	211
7.2	Modell des Standardregelkreises .....	214
7.2.1	Linearer Regelkreis .....	214
7.2.2	Nichtlinearer Regelkreis .....	217
7.3	Wichtige Eigenschaften von Regelkreisen .....	219
7.3.1	Störkompensation und Sollwertfolge .....	219
7.3.2	Erreichbare Regelgüte .....	228
7.3.3	Robustheit .....	230
7.4	Reglertypen .....	234
7.4.1	PID-Regler .....	234
7.4.2	Weitere Reglerkomponenten .....	237
	Literaturhinweise .....	239
<b>8</b>	<b>Einstellregeln für PID-Regler .....</b>	<b>241</b>
8.1	Entwurfsschritte .....	241
8.2	Einstellregeln von ZIEGLER und NICHOLS .....	243
8.3	Robuste PI-Regelung .....	250
8.3.1	Gegenkopplungsbedingung für I-Regler .....	250
8.3.2	Reglereinstellung .....	253
8.3.3	Erweiterung auf PI-Regler .....	255
8.4	Ausblick: Verfahren für den Reglerentwurf .....	259
	Literaturhinweise .....	260
<b>9</b>	<b>Zustandsbeobachtung kontinuierlicher Systeme .....</b>	<b>261</b>
9.1	Beobachtungsaufgabe .....	261
9.2	Luenbergerbeobachter .....	263
9.2.1	Grundidee .....	263
9.2.2	Beobachterstruktur .....	265
9.2.3	Wahl der Beobachterrückführung .....	267
9.2.4	Verhalten des Beobachters bei Störungen und Modellunsicherheiten .....	273
9.3	Beobachter für nichtlineare Systeme .....	278
9.4	Anwendungen der Zustandsbeobachtung .....	280
9.4.1	Beobachtung eines Teilsystems .....	280
9.4.2	Online-Vorhersage des Systemverhaltens .....	281
9.4.3	Regelung unter Verwendung einer beobachteten Regelgröße .....	283
	Literaturhinweise .....	285



<b>10 Diagnose kontinuierlicher Systeme</b>	287
10.1 Diagnoseaufgabe und Lösungswege	287
10.1.1 Diagnoseaufgabe	287
10.1.2 Diagnoseschritte	290
10.1.3 Signalbasierte und modellbasierte Diagnose	290
10.1.4 Modelle des fehlerfreien und des fehlerhaften Systems	292
10.1.5 Diagnose mit statischen und dynamischen Modellen	295
10.1.6 Prinzip der konsistenzbasierten Diagnose	295
10.2 Fehlererkennung mit einem Zustandsbeobachter	298
10.3 Sensorüberwachung	305
10.3.1 Aufgabenstellung	305
10.3.2 Fehlerlokalisierung mit dedizierten Beobachtern	306
10.3.3 Erweiterung	314
10.4 Fehleridentifikation	316
10.4.1 Fehleridentifikation mit einer Beobachterbank	316
10.4.2 Fehleridentifikation unter Verwendung von Fehlermodellen	317
10.4.3 Entwurf beobachtergestützter Diagnosesysteme	319
10.5 Ausblick: Diagnose und fehlertolerante Steuerung	328
Literaturhinweise	329

### Teil 3: Automatisierung ereignisdiskreter Systeme

<b>11 Beschreibung diskreter Systeme</b>	333
11.1 Modellbildungsaufgabe	333
11.1.1 Diskrete Signale und Ereignisse	333
11.1.2 Modellbildungsziel und Modellbildungsschritte	340
11.2 Deterministische Automaten	344
11.2.1 Autonome deterministische Automaten	344
11.2.2 Deterministische Automaten mit Eingang und Ausgang	349
11.3 Nichtdeterministische Automaten	355
11.3.1 Deterministische und nichtdeterministische Systeme	355
11.3.2 Autonome nichtdeterministische Automaten	357
11.3.3 Nichtdeterministische Automaten mit Eingang und Ausgang	367
11.4 Stochastische Automaten	372
11.4.1 Stochastische Prozesse	372
11.4.2 Autonome stochastische Automaten	376
11.4.3 Stochastische Automaten mit Eingang und Ausgang	382
11.4.4 Markoveigenschaft dynamischer Systeme	385
11.5 Petrinetze	389
11.5.1 Autonome Petrinetze	389
11.5.2 Petrinetze mit Eingang und Ausgang	400
11.5.3 Beziehungen zwischen Petrinetzen und Automaten	405

11.6	Kompositionale Modellbildung diskreter Systeme .....	407
11.6.1	Modellbildungsaufgabe .....	407
11.6.2	Synchronisation von Automaten .....	408
11.6.3	Reihenschaltung .....	413
11.6.4	Rückführautomat .....	417
	Literaturhinweise .....	425
<b>12</b>	<b>Verhalten diskreter Systeme .....</b>	<b>427</b>
12.1	Vorhersage des Systemverhaltens .....	427
12.2	Verhalten deterministischer Automaten .....	428
12.2.1	Berechnung der Zustands- und Ausgabefolge .....	428
12.2.2	Erreichbarkeitsanalyse des Automatengraphen .....	429
12.2.3	Strukturelle Analyse deterministischer Automaten .....	431
12.2.4	Steuerbarkeit deterministischer Automaten .....	433
12.3	Verhalten nichtdeterministischer Automaten .....	434
12.3.1	Berechnung der Zustands- und Ausgabefolgen .....	434
12.3.2	Strukturelle Analyse nichtdeterministischer Automaten .....	437
12.3.3	Steuerbarkeit nichtdeterministischer Automaten .....	438
12.4	Verhalten stochastischer Automaten .....	441
12.4.1	Berechnung der Zustandsfolgen autonomer stochastischer Automaten .....	441
12.4.2	Strukturelle Analyse stochastischer Automaten .....	444
12.4.3	Erweiterung auf Automaten mit Eingang und Ausgang .....	444
12.5	Verhalten von Petrinetzen .....	446
12.5.1	Berechnung der Markierungsfolgen .....	446
12.5.2	Strukturelle Analyse von Petrinetzen .....	447
12.5.3	Invarianten .....	448
	Literaturhinweise .....	455
<b>13</b>	<b>Steuerung diskreter Systeme .....</b>	<b>457</b>
13.1	Steuerungsaufgaben für diskrete Systeme .....	457
13.2	Verknüpfungssteuerungen .....	460
13.3	Ablaufsteuerungen .....	462
13.3.1	Reglergesetz .....	462
13.3.2	Zeitplansteuerungen .....	465
13.4	Speicherprogrammierbare Steuerungen .....	472
	Literaturhinweise .....	474
<b>14</b>	<b>Entwurf diskreter Steuerungen .....</b>	<b>475</b>
14.1	Entwurfsschritte .....	475
14.1.1	Systematischer Steuerungsentwurf .....	476
14.1.2	Heuristische Festlegung des Steuerungsgesetzes .....	477
14.2	Reglerentwurf für Automaten .....	480
14.2.1	Entwurfsproblem .....	480
14.2.2	Reglerentwurf für deterministische Automaten .....	480

14.2.3	Erweiterungen .....	483
14.2.4	Reglerentwurf für nichtdeterministische Automaten .....	487
14.3	Reglerentwurf für Petrinetze .....	488
14.3.1	Entwurf mit Hilfe des Erreichbarkeitsgraphen .....	488
14.3.2	Steuerungsentwurf unter Nutzung von S-Invarianten .....	490
14.4	Verifikation diskreter Steuerungen .....	497
	Literaturhinweise .....	500
<b>15</b>	<b>Zustandsbeobachtung diskreter Systeme .....</b>	<b>501</b>
15.1	Beobachtungsaufgabe .....	501
15.2	Beobachtung deterministischer und nichtdeterministischer Automaten .....	502
15.2.1	Beobachtungsalgorithmus für nichtdeterministische Automaten .....	502
15.2.2	Darstellung des Beobachtungsalgorithmus als Automat .....	509
15.2.3	Anwendung des Beobachtungsalgorithmus auf deterministische Automaten .....	511
15.2.4	Konsistenz von E/A-Paaren mit nichtdeterministischen Automaten ...	513
15.3	Beobachtung stochastischer Automaten .....	515
	Literaturhinweise .....	523
<b>16</b>	<b>Diagnose diskreter Systeme .....</b>	<b>525</b>
16.1	Diagnoseaufgabe .....	525
16.2	Diagnose nichtdeterministischer Automaten .....	526
16.2.1	Modellierung fehlerbehafteter Systeme .....	526
16.2.2	Detektion konstanter Fehler .....	528
16.2.3	Identifikation konstanter Fehler .....	529
16.2.4	Identifikation zeitabhängiger Fehler .....	536
16.3	Diagnose stochastischer Automaten .....	538
16.3.1	Beschreibung fehlerbehafteter Systeme durch stochastische Automaten	538
16.3.2	Grundidee der Diagnose stochastischer Systeme .....	539
16.3.3	Diagnosealgorithmus .....	545
	Literaturhinweise .....	552
<b>17</b>	<b>Ausblick: Überwachung und Steuerung hybrider dynamischer Systeme .....</b>	<b>553</b>
17.1	Automatisierung kontinuierlicher und ereignisdiskreter Systeme .....	553
17.2	Hybride dynamische Systeme .....	555
	Literaturhinweise .....	561
	<b>Literaturverzeichnis .....</b>	<b>563</b>
 <b>Anhänge</b> 		
	<b>Anhang 1: Lösung der Übungsaufgaben .....</b>	<b>567</b>
	<b>Anhang 2: Fachwörter deutsch – englisch .....</b>	<b>665</b>
	<b>Sachwortverzeichnis .....</b>	<b>669</b>



# Verzeichnis der Anwendungsbeispiele

## Prozessautomatisierung

Aufgaben der Prozessautomatisierung (Abschn. 1.2.1) .....	6
Batchreaktor und Paketlager (Beispiel 2.1) .....	49
Steuerung einer Screening-Anlage (Aufgabe 11.17) .....	406
Überwachung eines automatischen Qualitätskontrollsystems (Aufgabe 16.4) .....	544
<b>• Analyse und Steuerung einer Abfüllanlage</b>	
Steuerung einer Abfüllanlage (Aufgabe 2.5 mit Lösung) .....	79, 576
Blockschaltbild und Kopplungsstruktur (Beispiel 2.4) .....	59
Veränderung der Kopplungsstruktur (Beispiel 2.9) .....	67
Analyse der erweiterten Abfüllanlage (Aufgabe 2.3 mit Lösung) .....	70, 574
<b>• Modellierung und Analyse eines Wärmeübertragers</b>	
Zustandsraummodell eines Wärmeübertragers (Aufgabe 3.7 mit Lösung) .....	114, 590
Kennwertermittlung (Aufgabe 4.4 mit Lösung) .....	152, 594
Stabilitätsanalyse (Aufgabe 6.1 mit Lösung) .....	189, 604
<b>• Steuerung und Diagnose von Rührkesselreaktoren</b>	
Temperatur- und Füllstandsregelung eines Reaktors (Beispiel 1.2) .....	7
Temperaturverhalten eines Rührkesselreaktors (Aufgabe 4.2 mit Lösung) .....	140, 591
Stabilität des Temperaturregelkreises (Aufgabe 6.7 mit Lösung) .....	206, 610
Einstellung der Temperaturregelung (Beispiel 8.1) .....	246
Entwurf eines Beobachters für zwei gekoppelte Behälter (Aufgabe 9.1 mit Lösung) .....	272, 616
Fehlerdiagnose eines Reaktors mit Füllstandsregelung (Aufgabe 10.2 mit Lösung) .....	303, 617
Erprobung des Diagnosealgorithmus aus Aufgabe 10.2 (Aufgabe 10.3) .....	304
Diagnose zweier gekoppelter Reaktoren mit einem ereignisdiskreten Modell (Aufgabe 16.2 mit Lösung) .....	535, 663
<b>• Steuerung und Diagnose von Batchprozessen</b>	
Steuerung eines Batchprozesses (Beispiel 1.3) .....	8
Vergleich von Zeitplansteuerung und Ablaufsteuerung (Beispiel 2.11) .....	75

Kontinuierliches Modell eines Behältersystems (Aufgabe 3.4 mit Lösung) .....	111, 584
Verhalten eines Batchprozesses (Beispiel 12.5) .....	439
Rezeptsteuerung (Beispiel 14.1) .....	478
Wertefolgen und Ereignisse beim Füllen eines Reaktors (Beispiel 11.1) .....	337
Beschreibung eines Batchprozesses durch ein Petrinetz (Aufgabe 11.13 mit Lösung) .....	400, 632
Invarianten des Petrinetzes (Beispiel 12.9) .....	452
Kompositionale Modellbildung eines geregelten Vorratsbehälters (Aufgabe 11.24 mit Lösung) .....	424, 643
Steuerung eines Batchprozesses (Aufgabe 13.4 mit Lösung) .....	471, 651
Diagnose mit ereignisdiskretem Modell (Beispiel 16.1) .....	531
Modellierung eines Batchreaktors für die Fehlerdiagnose (Aufgabe 16.6 mit Lösung) .....	552, 663
Anwendung des Diagnosealgorithmus (Aufgabe 16.1) .....	534
Erweiterung der Diagnose unter Berücksichtigung der Fehlerwahrscheinlichkeit (Beispiel 16.5) .....	550
Darstellung eines Behältersystems als hybrider Automat (Aufgabe 17.1) .....	561
<b>• Diagnose von Servoventilen</b>	
Fehlerdiagnose eines Servoventils (Beispiel 10.4) .....	320
Diagnostizierbarkeitsanalyse (Aufgabe 10.5 mit Lösung) .....	327, 621
<b>• Überwachung von pH-Sensoren</b>	
Überwachung eines pH-Sensors (Beispiel 10.3) .....	308
Intelligenter pH-Sensor (Aufgabe 10.4 mit Lösung) .....	313, 620
<b>• Überwachung und Regelung eines Industrieofens</b>	
Steuerbarkeit eines Industrieofens (Beispiel 5.1) .....	158
Beobachtbarkeit (Beispiel 5.2) .....	166
Strukturelle Analyse (Beispiel 5.3) .....	172
Beobachterentwurf für den Industrieofen (Beispiel 9.2) .....	270
Verhalten des Beobachters bei Störungen (Beispiel 9.3) .....	275
Temperaturregelung des Industrieofens (Aufgabe 9.2) .....	284
<b>Fertigungsautomatisierung</b>	
Transportsysteme in der Schaltkreisfertigung (Beispiel 1.4) .....	11
<b>• Robotersteuerungen</b>	
Grundstruktur von Robotersteuerungen (Abschn. 1.2.2) .....	11
Steuerung eines elastischen Roboters (Beispiel 2.12) .....	78

## • Automatisierung von Fertigungszellen

Hierarchische Steuerung einer Fertigungszelle (Beispiel 1.10) .....	38
Modellierung der Fertigungszelle (Aufgabe 11.7 mit Lösung) .....	366, 627
Zustandsbeobachtung (Aufgabe 15.2 mit Lösung) .....	508, 658
Beschreibung der Zuverlässigkeit einer Werkzeugmaschine (Beispiel 11.10) .....	381
Kompositionale Modellbildung von drei Werkzeugmaschinen (Beispiel 11.14) .....	414
Beschreibung von vier Werkzeugmaschinen durch einen nichtdeterministischen Automaten (Beispiel 11.4) .....	363
Beschreibung durch ein Petrinetz (Beispiel 11.10) .....	397
Invarianten des Petrinetzes (Beispiel 12.8) .....	451
Beschreibung der vier Werkzeugmaschinen für die Materialflusssteuerung (Beispiel 11.11) .....	403
Analyse der vier Werkzeugmaschinen (Aufgabe 12.1) .....	438
Materialflusssteuerung für vier Werkzeugmaschinen (Aufgabe 14.4) .....	499
Zustandsbeobachtung der vier Werkzeugmaschinen (Aufgabe 15.3) .....	512

## • Modellierung einer Stanze

Ereignisdiskrete Beschreibung einer Stanze (Beispiel 11.2) .....	348
Modellerweiterung zur Berücksichtigung von Fehlermöglichkeiten (Beispiel 11.3) .....	360
Modellerweiterung zur Erfassung der Fehlerwahrscheinlichkeiten (Beispiel 11.7) .....	380
Verhaltensanalyse (Beispiel 12.6) .....	442
Analyse der gesteuerten Stanze (Aufgabe 14.3) .....	499

## Automatisierung von Kraftfahrzeugen

Automatische Automobile (Aufgabe 1.1) .....	6
Regelungsaufgaben bei Fahrzeugen (Abschn. 1.2.5) .....	14
Autofahren als Problem der Mehrebenensteuerung (Aufgabe 1.5 mit Lösung) .....	39, 569
Ereignisdiskrete Beschreibung des Autofahrens (Aufgabe 11.3 mit Lösung) .....	355, 625
Komfortverbesserung in Kraftfahrzeugen (Aufgabe 13.1) .....	469
Aktive Federung (Beispiel 1.9) .....	22

## • Modellierung der Längsbewegung

Blockschaltbild des Antriebsstrangs eines Kraftfahrzeugs (Aufgabe 2.1 mit Lösung) .....	61, 570
Längsbewegung eines Fahrzeugs (Beispiel 4.1) .....	129
Modellunbestimmtheiten und Störungen der Fahrzeugbewegung (Beispiel 7.5) .....	231

## • Antiblockiersystem

Bremsassistent und Antiblockiersystem (Beispiel 1.8) .....	15
Modell für den Abbremsvorgang eines Fahrzeugs (Beispiel 3.9) .....	107

Linearisierte Beschreibung des Abbremsvorgangs (Aufgabe 3.6 mit Lösung) .....	114, 587
Simulation des Abbremsvorgangs (Beispiel 4.2) .....	147
Strukturelle Analyse des Modells für den Abbremsvorgang (Aufgabe 5.6 mit Lösung) ....	178, 602
Stabilitätsanalyse des Schlupfverhaltens (Aufgabe 6.4 mit Lösung) .....	194, 606
Robustheit eines Antiblockiersystems (Aufgabe 6.8 mit Lösung) .....	207, 611
Realisierung eines Antiblockiersystems (Beispiel 7.7) .....	238
Diskrete Beschreibung eines Antiblockiersystems (Aufgabe 11.23) .....	423
<b>• Onboard-Diagnose</b>	
Diagnose von Kraftfahrzeugen (Beispiel 10.1) .....	288
Diagnose eines Motorkühlsystems (Beispiel 16.2) .....	542
Einklemmschutz für einen Fensterheber (Beispiel 10.2) .....	299
Strukturelle Steuerbarkeit und Beobachtbarkeit des Fensterhebers (Aufgabe 5.7) .....	179
<b>• Motorsteuerung</b>	
Aufgaben der Motorsteuerung (Beispiel 1.7) .....	14
Blockschaltbild einer Motorsteuerung (Aufgabe 2.2 mit Lösung) .....	62, 574
Beschreibung der Arbeitsweise eines Verbrennungsmotors (Beispiel 11.16) .....	406
Ereignisdiskrete Beschreibung des Startvorgangs eines Fahrzeugs (Aufgabe 11.6) .....	365
<b>• Modellierung eines Regensors</b>	
Beschreibung eines Regensors durch einen nichtdeterministischen Automaten (Beispiel 11.5) .....	368
Modellerweiterung zum stochastischen Automaten (Aufgabe 11.12 mit Lösung) .....	385, 631
Verhaltensanalyse (Aufgabe 12.2 mit Lösung) .....	446, 645
<b>• Autarke Unterwasserfahrzeuge</b>	
Nichtlineares Zustandsraummodell eines Unterwasserfahrzeugs (Beispiel 3.8) .....	103
Linearisierung des Modells (Aufgabe 3.5 mit Lösung) .....	114, 586
Stabilität des Unterwasserfahrzeugs (Aufgabe 6.3 mit Lösung) .....	194, 605
<b>Verkehrsleittechnik</b>	
Steuerung einer Schiffsschleuse (Aufgabe 14.2 mit Lösung) .....	496, 655
<b>• Ampelsteuerung</b>	
Steuerung der Ampel an einer Garageneinfahrt (Beispiel 13.1) .....	461
Steuerung einer „Bremsampel“ (Aufgabe 13.3 mit Lösung) .....	470, 648



## • Automatisierung von U-Bahn, Eisenbahn und Flugzeug

Aufgaben der Flugregelung (Abschn. 1.2.6) .....	16
Steuerung der vollautomatischen U-Bahn in Lille (Aufgabe 13.5) .....	471
Diskrete und kontinuierliche Elemente der U-Bahn-Steuerung (Beispiel 17.1) .....	555
Beschreibung eines Streckenabschnitts der Eisenbahn (Aufgabe 11.15 mit Lösung) .....	404, 636

## Automatisierungsaufgaben in der Energietechnik

### • Überwachung von Elektroenergienetzen

Automatisierungsaufgaben in Energiesystemen (Abschn. 1.2.4) .....	13
Überwachung und Steuerung von Elektroenergienetzen (Beispiel 1.6) .....	13
Kopplungsanalyse eines Elektroenergieversorgungsnetzes (Aufgabe 2.4 mit Lösung) .....	70, 575

### • Modellierung und Analyse von Gleichstrommotoren

Aufstellung der Differentialgleichung für einen Gleichstrommotor (Beispiel 3.1) .....	86
Zustandsraummodell (Beispiel 3.2) .....	90
Zustandstrajektorie (Beispiel 3.3) .....	94
Normierung des Zustandsraummodells (Beispiel 3.4) .....	96
Signalflussgraph (Beispiel 3.5) .....	98
Statisches Verhalten (Beispiel 3.6) .....	100
Verhalten bei Störungen und Modellunbestimmtheiten (Aufgabe 7.3 mit Lösung) .....	234, 613
Zustandsbeobachtung (Aufgabe 9.3) .....	284
Blockschaltbild eines drehzahlgeregelten Gleichstrommotors (Beispiel 2.3) .....	58
Hierarchische Modellbildung des drehzahlgeregelten Motors (Beispiel 2.7) .....	65
Zustandsraummodell des geregelten Motors (Beispiel 7.1) .....	215
Bleibende Regelabweichung beim drehzahlgeregelten Gleichstrommotor (Beispiel 7.3) ..	222
Verschlechterung der Regelgüte durch Messverzögerungen (Beispiel 7.4) .....	228

## Modellierung und Regelung mechanischer Systeme

Stabilitätsanalyse eines Pendels mit Hilfe des linearisierten Modells (Beispiel 6.2) .....	192
Fehlerdiagnose eines Pendels (Aufgabe 10.1) .....	303

### • Invertiertes Pendel

Aufrichten eines invertierten Pendels (Beispiel 2.10) .....	73
Zustandsraummodell (Aufgabe 3.2 mit Lösung) .....	102, 582
Stabilitätsanalyse (Aufgabe 6.2) .....	193
Stabilitätsanalyse des geregelten Pendels (Aufgabe 6.6 mit Lösung) .....	206, 609

Robuste Stabilisierung des invertierten Pendels (Aufgabe 6.9) .....	208
Stabilisierung des invertierten Pendels (Beispiel 7.6) .....	236
<b>• Radioteleskop</b>	
Zustandsraummodell (Aufgabe 3.3 mit Lösung) .....	110, 583
Regelung (Beispiel 7.2) .....	218
<b>Automatisierung von Kommunikationsnetzen</b>	
Beschreibung einer Rechnerkommunikation (Beispiel 11.13) .....	411
Beobachtbarkeit eines Kommunikationssatelliten (Aufgabe 5.8 mit Lösung) .....	180, 603
<b>Gebäudeautomatisierung</b>	
Aufgaben der Gebäudeautomatisierung (Beispiel 1.2.3) .....	12
Wirkungsweise einer Raumtemperaturregelung (Beispiel 1.5) .....	12
Blockschaltbild einer Raumtemperaturregelung (Aufgabe 2.6 mit Lösung) .....	79, 577
Steuerung einer Sicherheitsschleuse (Beispiel 14.3) .....	493
<b>• Steuerung und Diagnose einer Raumbelichtung</b>	
Beschreibung der Funktion einer Raumbelichtung durch einen Automaten (Aufgabe 11.8 mit Lösung) .....	372, 628
Modellerweiterung zum stochastischen Automaten (Aufgabe 11.11 mit Lösung) .....	385, 630
Diagnose einer Raumbelichtung (Aufgabe 16.3) .....	536
Steuerung einer Treppenhausbeleuchtung (Aufgabe 13.2 mit Lösung) .....	470, 647
<b>• Steuerung von Rolltreppen</b>	
Modellierung einer Rolltreppe (Aufgabe 11.18 mit Lösung) .....	412, 637
Beschreibung der gesteuerten Rolltreppe durch einen Rückführautomaten (Beispiel 11.15) .....	418
Regelung der Rolltreppe (Beispiel 13.3) .....	466
<b>• Modellierung und Steuerung eines Personenaufzugs</b>	
Beschreibung eines Personenaufzugs durch gekoppelte Automaten und ein Petrinetz (Aufgabe 11.22 mit Lösung) .....	422, 640
Steuerung eines Personenaufzugs (Beispiel 14.2) .....	483
Verhinderung verbotener Zustände (Aufgabe 14.1 mit Lösung) .....	496, 653

**Automatisierungstechnik im täglichen Leben**

Automatisierungstechnik im täglichen Leben (Aufgabe 1.2 mit Lösung) .....	24, 567
Automatiktoaster und andere automatische Geräte (Aufgabe 1.3 mit Lösung) .....	24, 568
Waschautomat (Beispiel 1.1) .....	5
Beschreibung eines Kartentelefons (Aufgabe 11.9) .....	372
Steuerung einer Digitalkamera (Aufgabe 11.14 mit Lösung) .....	404, 633
Modell eines Getränkeautomaten (Aufgabe 11.1 mit Lösung) .....	354, 624
Steuerung eines Getränkeautomaten (Beispiel 13.2) .....	462
Beschreibung eines Bestellvorgangs (Aufgabe 11.4 mit Lösung) .....	355, 626
Analyse des Elbehochwassers (Beispiel 2.5) .....	59
Rezeptsteuerung in einer Bäckerei (Aufgabe 1.4) .....	25
Automatentheoretische Beschreibung der Fußball-Weltmeisterschaft (Aufgabe 11.5) .....	355



# Hinweise zum Gebrauch des Buches

**Formelzeichen.** Die Wahl der Formelzeichen hält sich an folgende Konventionen: Kleine Buchstaben bezeichnen Skalare, z. B.  $x$ ,  $a$ ,  $t$ . Vektoren sind durch halbfette Kleinbuchstaben ( $\boldsymbol{x}$ ,  $\boldsymbol{a}$ ) und Matrizen durch halbfette Großbuchstaben ( $\boldsymbol{A}$ ,  $\boldsymbol{X}$ ) dargestellt. Entsprechend dieser Festlegung werden die Elemente der Matrizen und Vektoren durch kursive Kleinbuchstaben (mit Indizes) symbolisiert, beispielsweise durch  $x_1$ ,  $x_2$ ,  $x_i$  für Elemente des Vektors  $\boldsymbol{x}$  und durch  $a_{12}$ ,  $a_{ij}$  für Elemente der Matrix  $\boldsymbol{A}$ . Werden Größen, die im allgemeinen Fall als Vektor oder Matrix geschrieben werden (z. B.  $\boldsymbol{x}$ ,  $\boldsymbol{A}$ ), in einem einfachen Beispiel durch Skalare ersetzt, so wird dies durch den Übergang zu kleinen kursiven Buchstaben ( $x$  bzw.  $a$ ) verdeutlicht. Dann gelten die vorher mit Vektoren und Matrizen geschriebenen Gleichungen mit den skalaren Größen gleichen Namens.

Vektoren sind stets als Spaltenvektoren definiert. Die Transposition von Vektoren und Matrizen wird durch ein hochgestelltes „T“ gekennzeichnet ( $\boldsymbol{c}^T$ ).

Mengen sind durch kalligrafische Buchstaben dargestellt:  $\mathcal{Q}$ ,  $\mathcal{P}$ .


Bei den Indizes wird zwischen steil gesetzten Abkürzungen und kursiv gesetzten Laufindizes unterschieden. Beispielsweise kennzeichnet der Index „m“ bei  $y_m$  den *Messwert* einer Ausgangsgröße  $y$ , während er bei  $y_m$  den  $m$ -ten Systemausgang bezeichnet. Im zweiten Fall kann für  $m$  eine beliebige Zahl eingesetzt werden, und es gibt außer  $y_m$  z. B. auch die Größe  $y_{m-1}$ .

Die verwendeten Formelzeichen und Bezeichnungen orientieren sich an den international üblichen und weichen deshalb von der DIN 19299 ab. Beispielsweise werden bei kontinuierlichen Systemen für die Regel- und die Stellgröße die Buchstaben  $y$  und  $u$  verwendet.  $x$  und  $\boldsymbol{x}$  sind die international gebräuchlichen Formelzeichen für eine Zustandsvariable bzw. den Zustandsvektor.

Wenn bei einer Gleichung hervorgehoben werden soll, dass es sich um eine Forderung handelt, die durch eine geeignete Wahl von bestimmten Parametern erfüllt werden soll, wird über das Gleichheitszeichen ein Ausrufezeichen gesetzt ( $\stackrel{!}{=}$ ).

Bei den Beispielen wird mit Zahlengleichungen gearbeitet, in die die physikalischen Größen in einer zuvor festgelegten Maßeinheit einzusetzen sind. Bei den Ergebnissen werden die Maßeinheiten wieder an die Größen geschrieben. Dabei wird zur Vereinfachung der Darstellung in den Gleichungen nicht zwischen den physikalischen Größen und ihren auf eine vorgegebene Maßeinheit bezogenen Größen unterschieden (vgl. Abschn. 3.3.3).

**Übungsaufgaben.** Die angegebenen Übungsaufgaben sind ihrem Schwierigkeitsgrad entsprechend folgendermaßen gekennzeichnet:

- Aufgaben ohne Markierung dienen der Wiederholung und Festigung des unmittelbar zuvor vermittelten Stoffes. Sie können in direkter Analogie zu den behandelten Beispielen gelöst werden.
- Aufgaben, die mit einem Stern markiert sind, befassen sich mit der Anwendung des Lehrstoffes auf ein praxisnahes Beispiel. Für ihre Lösung werden vielfach außer dem unmittelbar zuvor erläuterten Stoff auch Ergebnisse und Methoden vorhergehender Kapitel genutzt. Die Leser sollen bei der Bearbeitung dieser Aufgaben zunächst den prinzipiellen Lösungsweg festlegen und erst danach die Lösungsschritte nacheinander ausführen. Die Lösungen dieser Aufgaben sind im Anhang 1 angegeben.
- Aufgaben, die umfangreiche numerische Rechnungen erfordern und deshalb zweckmäßigerweise unter Verwendung von MATLAB gelöst werden sollten, sind durch das Symbol  gekennzeichnet.

**Weitere Informationen.** Von der Homepage des Lehrstuhls für Automatisierungstechnik und Prozessinformatik der Ruhr-Universität Bochum können weitere Informationen sowie die Abbildungen dieses Buches in A4-Vergrößerung für die Verwendung in der Vorlesung bezogen werden:

<http://www.atp.rub.de/Buch/AT>

## Teil 1: Einführung





# 1

## Ziele und Aufgaben der Automatisierungstechnik

*Das Einführungskapitel veranschaulicht anhand von Beispielen aus verschiedenen Anwendungsfeldern die Ziele und die grundlegenden Aufgaben der Automatisierungstechnik und erläutert die unterschiedlichen Informationskopplungen zwischen dem zu automatisierenden System und der Automatisierungseinrichtung, die für die Lösung von automatisierungstechnischen Aufgaben typisch sind.*

### 1.1 Ziele der Automatisierungstechnik

Die Automatisierungstechnik beschäftigt sich mit der Überwachung und Steuerung technischer Systeme. Geräte und Anlagen sollen durch Automatisierungseinrichtungen so überwacht und gesteuert werden, dass sie ihre Funktionen weitgehend selbsttätig erfüllen und Sicherheitsanforderungen genügen. In Erweiterung dessen will man zukünftig sogar erreichen, dass automatisierte Systeme die aktuell zu erfüllenden Aufgaben selbstständig erkennen und daraus die auszuführenden Lösungsschritte ableiten, also autonom (ohne den Eingriff eines Menschen) arbeiten.

|| Es ist das Ziel der Automatisierungstechnik, Systeme so zu steuern, dass die automatisierten Systeme selbstständig ihre Funktion erfüllen.

Anschauliche Beispiele für autonom arbeitende Systeme sind Raumflugkörper, die ohne menschliche Hilfe Messaufgaben durchführen, ihre Position bestimmen und gegebenenfalls ihre Bewegung korrigieren. Fahrerlose Transportsysteme erkunden den Mars oder bewegen selbsttätig Werkstücke zu Fertigungseinrichtungen oder Wafer durch die Reinräume der Schaltkreisherstellung. In medizinischen Labors werden Serien von Proben ohne Eingriff eines Menschen analysiert und in der Fahrzeugindustrie schweißen mehrere Roboter gemeinsam eine Karosserie. Auch im täglichen Leben ist die Automatisierungstechnik allgegenwärtig, wenn auch nicht immer ohne Weiteres zu erkennen. Fotoapparate, Waschmaschinen, Videorecorder und viele andere Geräte enthalten komplizierte Regelungen und Steuerungen, durch die sie ihre Aufgaben weitgehend selbstständig lösen. Selbst wenn im Folgenden allgemein von technischen Anlagen gesprochen wird, sind auch solche Geräte in die Betrachtungen eingeschlossen.

Die Bedeutung der Automatisierungstechnik kann man am besten dadurch verstehen, dass man sich Anlagen ohne Steuereinrichtungen vorstellt. Ein Personenaufzug ohne Steuerung steht und reagiert nicht auf Fahrtwünsche, eine Rolltreppe schaltet sich nicht selbsttätig ein und aus, Temperatur und Luftfeuchtigkeit großer Räume werden nicht selbsttätig der wechselnden Außentemperatur, der Sonneneinstrahlung und der sich verändernden Anzahl anwesender Personen angepasst. Die Fahrzeugtechnik wäre auf dem Stand von Mitte des letzten Jahrhunderts ohne Antiblockiersysteme, geregelte Katalysatoren, Servolenkung und Klimaanlage. Nicht bei allen Beispielen führt die Automatisierung zu einer vollständig autonomen Arbeitsweise, aber in jedem Fall sorgt sie dafür, dass die Anlage wichtige Teilprozesse selbsttätig ausführt.

Die Automatisierung übernimmt in allen genannten Systemen Überwachungs- und Steuerungsaufgaben:

- *Überwachen* heißt, die wichtigsten Prozessgrößen zu messen und die Messwerte zu sammeln, auszuwerten und für das Bedienpersonal in zweckmäßiger Weise darzustellen. Die Auswertung umfasst auch die Verknüpfung der Informationen, die von unterschiedlichen Sensoren kommen. Mit Hilfe eines Modells können nicht messbare Größen berechnet und Fehler aufgedeckt werden.
- *Steuern* heißt, Prozesse zielgerichtet zu beeinflussen. Die von außen vorgebbaren Größen werden so gewählt, dass die in der Anlage ablaufenden Prozesse in einer gewünschten Folge aktiviert oder wichtige Prozessgrößen trotz des Einwirkens von Störungen auf vorgeschriebenen Werten gehalten werden. Der Begriff Steuerung in seiner allgemeinen Bedeutung umfasst sowohl kontinuierlich wirkende als auch diskret arbeitende Steuerungen und er kann sich auch auf Steuereingriffe des Bedienpersonals beziehen. Wenn die Steuerung im geschlossenen Wirkungskreis erfolgt, spricht man von einer Regelung.

Plakativ ausgedrückt heißt das:

|| Automatisieren = Überwachen + Steuern.

Dieses Buch beschäftigt sich mit den *Methoden*, die man kennen und anwenden muss, um technische Systeme durch mathematische Modelle zu beschreiben, die wichtigsten Eigenschaften herauszufinden, Automatisierungsaufgaben mit Hilfe dieser Modelle zu lösen und das Verhalten des Gesamtsystems, das aus dem zu steuernden Prozess und der Automatisierungseinrichtung

besteht, zu analysieren und zu bewerten. Diese Methoden sind das wichtigste *Know-how* der Automatisierungstechniker.

Daneben sind für die Lösung automatisierungstechnischer Aufgaben auch Automatisierungsgeräte wie Sensoren, Stellglieder (Aktoren), Rechner für die Echtzeitdatenverarbeitung und Programmmodule für die Realisierung einzelner Komponenten einer Automatisierungseinrichtung notwendig. Auf diese gerätetechnischen Fragen wird hier nur am Rande eingegangen.

**Mechanisierung und Automatisierung.** Wie der Name Automatisierungstechnik sagt, sollen die in einem Gerät oder einer Anlage ablaufenden Prozesse *automatisch*, also ohne Eingriff des Menschen, ablaufen. Häufig denkt man dabei an Tätigkeiten, die früher der Mensch *manuell* ausführen musste und die jetzt von Maschinen übernommen werden. Dabei muss man jedoch zwischen mechanisierten und automatisierten Prozessen unterscheiden. Um Tätigkeiten zu automatisieren, müssen zuerst Konstruktionen und Verfahren entworfen und realisiert werden, die die manuellen Tätigkeiten ersetzen. Dieser Schritt wird *Mechanisierung* genannt. Die entstehenden Maschinen ersetzen die körperliche Tätigkeit durch eine maschinelle, wobei nach einem Kommando der bisher durch Menschen erledigte Prozess selbsttätig abläuft. Die Steuereingriffe, die die Teilprozesse zum richtigen Zeitpunkt in Gang setzen, kommen bei mechanisierten Systemen allerdings weiterhin vom Menschen.

Darauf aufbauend sorgt die Automatisierung für die *selbsttätige Steuerung* von Anlagen, die die mechanisierten Teilprozesse aktiviert und überwacht. Damit wird die zur Steuerung notwendige *geistige* Tätigkeit einem technischen Gerät übertragen. Aus dieser Sicht besteht die Automatisierung also in der Schaffung und Anwendung von technischen Mitteln, mit deren Hilfe mechanisierte Operationen nach vorgegebenen Programmen selbsttätig gesteuert werden.

### Beispiel 1.1 Waschautomat

Die Schritte der Mechanisierung und der Automatisierung werden offensichtlich, wenn man das Wäschewaschen per Hand durch eine Waschmaschine erleichtern will. Dass sich die Trommel mit der Wäsche dreht, Wasser in die Waschmaschine hineingelassen und wieder abgepumpt wird und das Wasser gegebenenfalls auf eine vorgegebene Temperatur erhitzt wird, ist das Ergebnis einer Mechanisierung. Frühe Formen von Waschmaschinen, die damals bereits mit der Bezeichnung „Waschautomat“ verkauft wurden, führten diese mechanisierten Prozesse durch, aber die Prozesse mussten von der Hausfrau an- und abgeschaltet werden.

Dieses An- und Abschalten übernimmt bei modernen Waschautomaten die eingebaute Automatisierungseinrichtung. Sie sorgt dafür, dass ein richtiger „Waschautomat“ die gewünschten Prozesse in der programmierten Reihenfolge und mit der gewünschten Dauer ausführt. Für Wäsche unterschiedlicher Art kann man zwischen verschiedenen Waschprogrammen wählen, wodurch nicht die Mechanisierung des Waschprozesses verändert wird, sondern das Automatisierungsziel und folglich die Folge von Steuereingriffen, mit denen mechanisierte Teilprozesse aktiviert werden.

Die Automatisierung erfordert i. Allg. den Einbau neuer Sensoren, mit denen der Fortschritt der aktivierten Prozesse gemessen wird. Beim Waschautomat muss beispielsweise mit Hilfe eines Temperatursensors festgestellt werden, wann die Waschlauge die richtige Temperatur besitzt, oder mit einer Uhr gemessen werden, wie lange der Waschprozess bereits andauert. □

**Aufgabe 1.1** Automatische Automobile

Die Automatisierung von Kraftfahrzeugen schreitet schnell voran. Überlegen Sie sich für die im Folgenden aufgeführten Prozesse, was gemessen werden muss und auf welche Stellgrößen eine Automatisierungseinrichtung eingreifen muss. Welche Schritte der Mechanisierung sind notwendig, damit die Automatisierungsaufgabe lösbar wird?

1. **Automatisches Anfahren am Berg:** Jeder Fahrschüler weiß, wie schwierig es für Anfänger ist, mit einem Fahrzeug am Berg anzufahren, ohne zurückzurollen. Kupplung, Gaspedal und Handbremse müssen dabei in genau abgestimmter Weise betätigt werden.  
Welche Schritte der Mechanisierung und der Automatisierung muss man ausführen, damit das Fahrzeug automatisch am Berg anfahren kann, so dass der Fahrer wie bei modernen Fahrzeugen nur noch das Gaspedal zu betätigen braucht? Kann man ein Fahrzeug mit Handbremse automatisch anfahren? Welche zusätzlichen Sensoren sind notwendig?
2. **Automatische Kofferraumklappe:** Zur Komfortverbesserung soll die Kofferraumklappe automatisch abgeschlossen werden, wenn die Fahrgäste eingestiegen sind und das Fahrzeug anfährt. Welche Mechanisierungs- und Automatisierungsschritte sind hierfür durchzuführen?
3. **Automatisches Einparken:** Rückwärtseinparken ist Übungssache, aber es gibt bei modernen Fahrzeugen Einparkhilfen, bei denen der Fahrer nur noch Gas geben muss. Welche Schritte der Mechanisierung und der Automatisierung waren erforderlich, bevor die ersten Systeme serienreif waren?
4. **Automatisches Parkhaus:** Es ist heute noch ein Traum, dass der Fahrer sein Fahrzeug am Eingang des Parkhauses abgeben kann und das Fahrzeug selbstständig zu einem freien Platz fährt. Nach einem Anruf vom Handy soll das Fahrzeug allein zur Ausfahrt zurückkommen. Welche Mechanisierungs- und Automatisierungsschritte sind hierfür notwendig? Welche Informationen muss das Fahrzeug vom Parkhaus bekommen, bevor es losfahren kann? ☐

## 1.2 Anwendungsbeispiele

### 1.2.1 Prozessautomatisierung

Die Realisierung verfahrenstechnischer Prozesse beinhaltet vielfältige Automatisierungsaufgaben, so dass für dieses Gebiet ein eigener Begriff geprägt wurde: Prozessautomatisierung. Die Palette der zu lösenden Aufgaben beginnt bei der Informationsgewinnung, -übertragung und -aufbereitung für das Bedienpersonal in der Warte einer großen verfahrenstechnischen Anlage und führt über die Realisierung von Durchfluss-, Temperatur- oder Füllstandsregelungen bis zu Rezeptsteuerungen, mit denen Batchprozesse realisiert werden. Darüber hinaus kann die Automatisierungseinrichtung auch die Produktionsplanung übernehmen.

Die Verfahrenstechnik ist eines der ältesten Anwendungsgebiete der Automatisierungstechnik, denn hier werden automatische Steuerungs- und Überwachungseinrichtungen bereits seit etwa 1960 in breitem Umfang genutzt.

**Kontinuierliche Prozesse.** Viele verfahrenstechnische Prozesse laufen kontinuierlich ab. Drücke, Temperaturen, Konzentrationen, Durchflüsse, pH-Werte usw. müssen kontinuierlich überwacht und konstant gehalten oder in einer vorgeschriebenen Weise verändert werden. Dafür sind an industriellen Anlagen typischerweise hunderte von Regelkreisen installiert.

Bei der Betrachtung dieser Automatisierungsaufgaben sind vier Signaltypen wichtig:

- **Regelgröße:** Die Größe, die auf einem vorgegebenen Wert gehalten werden soll, wird Regelgröße genannt und mit dem Symbol  $y$  bezeichnet. So ist bei einer Temperaturregelung die Temperatur die Regelgröße, bei einer Füllstandsregelung der Füllstand usw.
- **Stellgröße:** Um ihre Aufgabe erfüllen zu können, muss die Regeleinrichtung eine Eingangsgröße des Prozesses so verändern, dass die Regelgröße den vorgegebenen Wert annimmt. Diese Eingangsgröße heißt Stellgröße  $u$ .
- **Führungsgröße:** Der gewünschte Verlauf der Regelgröße wird durch ein Signal  $w$  vorgegeben, das Führungsgröße heißt.  $w$  kann konstant sein oder sich zeitlich ändern. Im ersten Falle spricht man bei  $w$  auch von einem *Sollwert*. In beiden Fällen besteht das Regelungsziel in der Erfüllung der Forderung  $y(t) \stackrel{!}{=} w(t)$ . Um dies zu erfüllen, erhalten Regler die *Regelabweichung*  $e(t) = w(t) - y(t)$  als Eingangsgröße und sollen die Stellgröße so bestimmen, dass die Regelabweichung asymptotisch verschwindet.
- **Störgröße:** Abweichungen zwischen der Regelgröße und der Führungsgröße werden durch nicht beeinflussbare äußere Störungen hervorgerufen, die durch das Signal  $d$  dargestellt werden. Bei der Füllstandsregelung ist beispielsweise die Entnahme von Flüssigkeit aus einem Behälter die Störung.

Diese Signaltypen treten auch bei diskreten Prozessen auf.

### Beispiel 1.2 Temperatur- und Füllstandsregelung eines Reaktors

Als Beispiel zeigt Abb. 1.1 einen Reaktor, dessen Aufgabe darin besteht, Flüssigkeit mit einer vorgegebenen Solltemperatur für die weitere Verarbeitung bereitzustellen. Die Flüssigkeit kann bei Bedarf durch das untere Ventil entnommen werden, wodurch der Reaktorfüllstand verkleinert („gestört“) wird. Der Reaktor ist mit einer Füllstandsregelung (LC – *level control*) ausgerüstet, die das obere Ventil öffnet, damit der Reaktor stets bis zu einer vorgegebenen Höhe gefüllt ist. Die Temperaturregelung (TC – *temperature control*) greift auf das Ventil zu, das den Dampfstrom durch die Heizung dosiert. Ihre Funktion ist es, die Temperatur konstant zu halten.

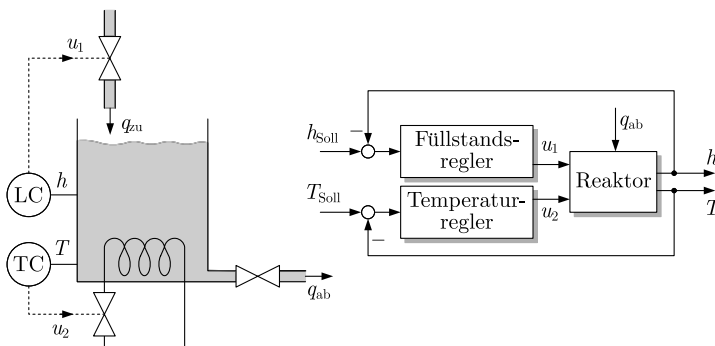


Abb. 1.1: Füllstands- und Temperaturregelung eines Reaktors

Die Funktionsweise der Regelung ist in dem rechts angegebenen Blockschaltbild zu sehen, in dem der Reaktor durch einen Block mit den beiden Ventilstellungen als Stellgrößen  $u_1$  und  $u_2$ , der Füllhöhe  $h$  und der Temperatur  $T$  als Regelgrößen sowie dem Flüssigkeitsstrom  $q_{ab}$  als Störgröße zu sehen ist. Die gemessenen Istwerte für die Füllhöhe und die Temperatur werden mit den vorgegebenen Sollwerten  $h_{\text{Soll}}$  bzw.  $T_{\text{Soll}}$  verglichen. Die erhaltenen Differenzen dienen als Eingangsgrößen für die beiden Regler, die die Ventilstellungen vorgeben.

Kennzeichnend für die Regelung sind die im Blockschaltbild erkennbaren Rückführungen. Der gemessene Temperaturwert  $T$  beeinflusst den Temperaturregler, der die Ventilstellung  $u_2$  vorgibt und damit auf die Temperatur  $T$  zurückwirkt. Derartige Rückführungen sind ein wichtiges Hilfsmittel der Automatisierungstechnik in allen Anwendungsbereichen. Sie treten in vielfältiger Form in diesem Buch auf.

Die Regelungen des Füllstands und der Temperatur können getrennt voneinander aufgebaut werden, weil sich beide Größen zwar gegenseitig beeinflussen, dieser Einfluss die Funktionsweise der getrennten Regelungen jedoch nicht gefährdet. Die Regelung besteht also aus zwei unabhängigen Regelkreisen und wird deshalb als *dezentrale Regelung* bezeichnet.  $\square$

**Batchprozesse.** Eine andere Klasse verfahrenstechnischer Anlagen realisiert Batchprozesse, bei denen Stoffe in bestimmten Mengen in Apparate eingefüllt werden, dort eine bestimmte Zeit verbleiben, dabei chemisch reagieren oder mechanisch bearbeitet werden und schließlich in nachfolgende Apparate weitergeleitet werden. Die Automatisierung derartiger Prozesse erfolgt durch diskrete Steuerungen, die im Unterschied zu kontinuierlichen Steuerungen mit diskreten, häufig binären Mess- und Stellgrößen arbeiten und beispielsweise Ventile (vollständig) öffnen und schließen oder Heizungen und Mühlen an- und abschalten.

Auch bezüglich des Zeitverhaltens gibt es grundlegende Unterschiede zwischen diskreten und kontinuierlichen Steuerungen. Während im kontinuierlichen Regelkreis der Istwert fortwährend mit dem Sollwert der Regelgröße verglichen und die Stellgröße dementsprechend verändert wird, arbeitet eine diskrete Steuerung nur zu den Zeitpunkten, an denen eine Messgröße das Erreichen eines vorgegebenen Grenzwertes signalisiert. Zwischen diesen Zeitpunkten sind die Stellgrößen konstant. Nichtsdestotrotz wirkt auch eine diskrete Steuerung im geschlossenen Wirkungskreis. Wenn dieser Aspekt hervorgehoben werden soll, wird auch bei diskreten Systemen von einer Regelung gesprochen.

### Beispiel 1.3 Steuerung eines Batchprozesses

Abbildung 1.2 zeigt das R&I-Fließbild (Rohrleitungs- & Instrumentenfließbild) für einen Batchprozess, der aus drei Behältern besteht, die durch zwei Rohrleitungen verbunden sind.  $V_1$ ,  $V_2$  und  $V_3$  bezeichnen drei Ventile und  $M$  den Motor, der einen Rührer antreibt. Ein verfahrenstechnisches Rezept sieht folgende Schritte vor:

1. Es wird Flüssigkeit aus dem Behälter  $B_1$  in den leeren Behälter  $B_3$  bis zu der vom Sensor  $L_1$  überwachten Füllhöhe gefüllt.
2. Es wird Flüssigkeit aus dem Behälter  $B_2$  in den Behälter  $B_3$  bis zur Füllhöhe  $L_2$  zugeführt.
3. Der Rührer wird angeschaltet.
4. Das Gemisch wird auf eine vorgegebene Temperatur  $T_{\text{Soll}}$  erhitzt.
5. Der Rührer wird abgeschaltet.
6. Das Produkt wird durch Öffnen des Ventils  $V_3$  abgezogen.

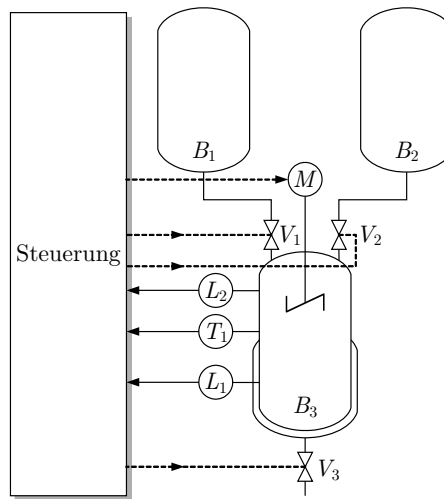


Abb. 1.2: Batchprozess

Es ist offensichtlich, dass zur Durchführung dieses Rezepts nur diskrete Grenzwerte für den Füllstand bzw. die Temperatur überwacht werden müssen und dass nur diskrete Stellgrößen (Ventil auf/zu; Heizung an/aus; Rührermotor an/aus) durch die Steuereinrichtung vorzugeben sind. Insofern unterscheidet sich die hier betrachtete Regelung von den im Beispiel 1.2 behandelten Regelungen, bei denen alle Größen wertkontinuierlich waren.

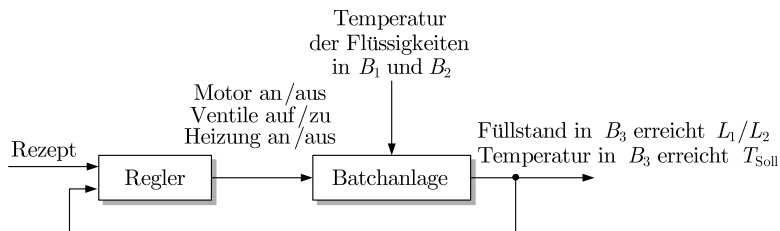
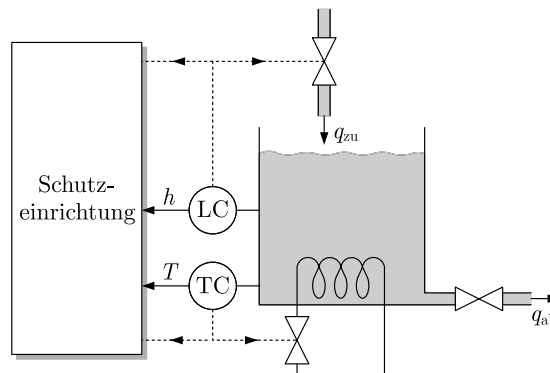


Abb. 1.3: Blockschaltbild des gesteuerten Batchprozesses

Strukturell sind diskrete Steuerungsaufgaben jedoch den vorher betrachteten Regelungsaufgaben sehr ähnlich (Abb. 1.3). Auch hier werden die gemessenen Größen durch einen Regler verarbeitet, um die Eingangsgrößen des Prozesses geeignet festzulegen. Das Steuerungsziel ist durch das Rezept als eine Folge von Prozessschritten vorgegeben. Der Vorgang wird durch Störgrößen beeinflusst, z. B. durch veränderte Temperaturen der in den Behältern  $B_1$  und  $B_2$  vorgelegten Flüssigkeiten. Im Unterschied zur kontinuierlichen Temperaturregelung reagiert die hier betrachtete Regelung aber nur zu den Zeitpunkten, zu denen die Messgrößen das Erreichen vorgegebener Grenzwerte für die Temperatur und den Füllstand anzeigen. □

Kontinuierliche Prozesse und Batchprozesse gehören zu den beiden sich in ihrem Verhalten grundlegend unterscheidenden Klassen der wertkontinuierlichen bzw. der ereignisdiskreten Systeme. Diesen beiden Klassen ist jeweils etwa die Hälfte dieses Buches gewidmet.

**Schutzeinrichtungen.** Die bisher behandelten Steuerungen und Regelungen werden zusammen mit Komponenten für die Informationsbereitstellung in der Warte zur MSR-Betriebseinrichtung zusammengefasst (MSR – Messen, Steuern, Regeln). Getrennt davon verfügen verfahrenstechnische Anlagen über eine MSR-Schutzeinrichtung, die unzulässige Anlagenzustände verhindert und die Anlage im Gefahrenfall abschaltet.



**Abb. 1.4:** Steuerung und Aggregateschutz eines Batchprozesses

Die aus Sicherheitsgründen vorgenommene Trennung von Betriebs- und Schutzeinrichtungen ist in Abb. 1.4 zu sehen, in der die Reaktorregelung aus Abb. 1.1 um eine Schutzeinrichtung erweitert wurde. Die Schutzeinrichtung überwacht den Füllstand  $h$  und die Temperatur  $T$  und verändert im Fehlerfall die von den beiden Regelungen an die Ventile gesendeten Stellinformationen. So wird beispielsweise beim Überschreiten eines oberen Grenzwertes für die Temperatur das Ventil der Heizung durch die Schutzeinrichtung geschlossen.

**Prozessleittechnik.** Neben der Realisierung von Steuerungen und Regelungen hat die Automatisierungseinrichtung auch die Aufgabe, die für die Überwachung des bestimmungsgemäßen Betriebs der Anlage durch das Bedienpersonal notwendigen Informationen in die Warte zu übertragen, auszuwerten und darzustellen. Außerdem müssen die Stellbefehle auf dem entgegengesetzten Weg von der Warte zu den Stellgliedern der Anlage übermittelt werden. Damit stellt die Automatisierungseinrichtung auch die Schnittstelle zwischen der Anlage und dem Bedienpersonal dar und schafft die Voraussetzung dafür, dass der Mensch die Anlage führen kann. Die dafür erforderliche Gerätetechnik wird als Prozessleittechnik oder Prozessleitsystem bezeichnet. Diese Begriffe sind abgeleitet vom Begriff leiten, der die Gesamtheit aller Maßnahmen bezeichnet, die einen erwünschten Ablauf eines Prozesses bewirken.

Die Prozessleittechnik schließt das weite Feld der Mensch-Maschine-Kommunikation ein, das sich mit der zweckmäßigen Darstellung der Informationen auf Bildschirmen sowie mit Geräten für die Eingabe manueller Steuerbefehle befasst.



Die hier genannten Begriffe wurden zunächst für die Automatisierung von Prozessen der Verfahrenstechnik und Energietechnik eingeführt. Sie sind jetzt jedoch auch in vielen anderen Anwendungsbereichen gebräuchlich.

### 1.2.2 Fertigungsautomatisierung

Die Automatisierung ist ein unverzichtbares Element moderner Fertigungssysteme. Sie überwacht und steuert die Bewegung von Werkzeugmaschinen und Robotern.

Als eigenständiges Gebiet hat sich die Fertigungsautomatisierung wesentlich später als die Prozessautomatisierung herausgebildet. Der Grund dafür war vor allem die Tatsache, dass die Rechentechnik erst zwischen 1970 und 1980 einen solchen Stand erreichte, dass sie die zahlreichen Mess- und Stellgrößen fertigungstechnischer Anlagen in Echtzeit verarbeiten konnte. Deshalb sind noch heute viele Begriffe der Automatisierungstechnik stärker durch die Prozessautomatisierung geprägt als durch die Fertigungsautomatisierung.

Neben einer Qualitätserhöhung und der Verkürzung der Fertigungszeiten führt die Automatisierung von Fertigungsprozessen auch zu einer Erhöhung der Flexibilität. Unterschiedliche Produkte lassen sich auf denselben Werkzeugmaschinen fertigen, weil lediglich die Steuerungsalgorithmen ausgetauscht werden müssen.

Auch die Einsatzplanung der Werkzeugmaschinen ist eine Automatisierungsaufgabe. Sie betrifft den Werkstückfluss von Maschine zu Maschine, der so geplant wird, dass die Maschinen gut ausgelastet sind. Das Ergebnis der Einsatzplanung passt die Steuerung der einzelnen Werkzeugmaschinen an die aktuellen Aufträge an. Da alle Elemente der Fertigung mit einer durchgängigen Datenverwaltung arbeiten, spricht man von einer rechnerintegrierten Fertigung (CIM – *computer-integrated manufacturing*).

Eine Erweiterung der Automatisierungsfunktionen soll zukünftig sogar auf eine sich selbst organisierende Produktion führen, die unmittelbar auf Kundenwünsche reagieren und die Prozessabläufe flexibel den aktuellen Erfordernissen anpassen kann. Die „Industrie 4.0“ (in Anlehnung an die vierte industrielle Revolution) soll in der Lage sein, individualisierte Produkte herzustellen.

**Materialflusssysteme.** Die Fertigungstechnik umfasst vielfältige Systeme zur Lagerung, zum Transport, zur Kommissionierung und zur Verpackung von Gütern. Zu diesen Systemen gehören Hochregallager, fahrerlose Transportsysteme und automatisch arbeitende Förder- und Lagereinrichtungen. Hier kommt es bei der Automatisierung vor allem auf Steuerungsstrategien an, die eine kurze Durchlaufzeit, Termintreue und geringe Lagerbestände sichern. Die Anforderungen an die Steuerung dieser Systeme gehen deshalb aus einer detaillierten Planung hervor, bei der die einzelnen Transportschritte in eine zeitliche Reihenfolge gebracht werden, wobei die beschränkte Kapazität der Transport- und Lagertechnik ebenso zu beachten ist wie der Fertigungsplan.

#### Beispiel 1.4 Transportsysteme in der Schaltkreisfertigung

Bei der Schaltkreisfertigung durchlaufen die Wafer eine Vielzahl von Bearbeitungsschritten, bei denen sie mehrfach hintereinander maskiert, dotiert und schließlich zerschnitten, geprüft und gebondet werden. Den wiederholten gleichartigen Bearbeitungsschritten entsprechend werden die Wafer von einem

Transportsystem zwischen den Maschinen hin- und herbewegt. Eine wichtige Automatisierungsaufgabe besteht in der Steuerung und Überwachung dieses Transportsystems.

Diese Automatisierungsaufgabe kann mit diskreten Modellvorstellungen gelöst werden. Die Position und der Inhalt eines Transportbehälters werden zur Beschreibung des aktuellen Zustands des Transportsystems herangezogen. Eine Zustandsänderung beschreibt die Bewegung der Behälter zwischen den Abzweigen der Transportbahn und den Maschinen. Beim Entwurf des Transportsystems und bei der Überwachung und Steuerung sind u. a. die folgenden Aufgaben zu lösen:

- Analyse der Quelle-Ziel-Beziehungen: Sind alle erforderlichen Transportwege durch eine geeignete Steuerung realisierbar?
- Belastungsanalyse: Gibt es Zweige, die unter normalen Fertigungsbedingungen überlastet sind?
- Steuerung: Wie müssen die Transportbehälter gesteuert werden, damit alle Bereiche der Fertigung gleichmäßig ausgelastet sind?
- Fehlerdiagnose: Ist im Transportsystem ein Fehler aufgetreten, so dass Transportbehälter nicht die gewünschten Ziele erreichen?
- Rekonfiguration: Wie kann das Transportsystem unter Umgehung eines durch Fehler blockierten Bereiches so gesteuert werden, dass die Fertigung weitergeht? □

### 1.2.3 Gebäudeautomatisierung

Der Betrieb von Gebäuden stellt für die Automatisierungstechnik vielfältige Aufgaben. Um das Klima in großen Räumen angenehm zu gestalten, muss die Raumtemperatur auf einem konstanten Wert gehalten, eine angemessene Luftfeuchtigkeit erzeugt und für genügend Frischluft gesorgt werden, ohne dass durch den Luftaustausch eine unangenehme Luftbewegung entsteht. Ferner müssen Fahrstühle gesteuert, die Beleuchtung der Räume und Treppenhäuser überwacht und der Zugang zu Gebäuden kontrolliert werden.

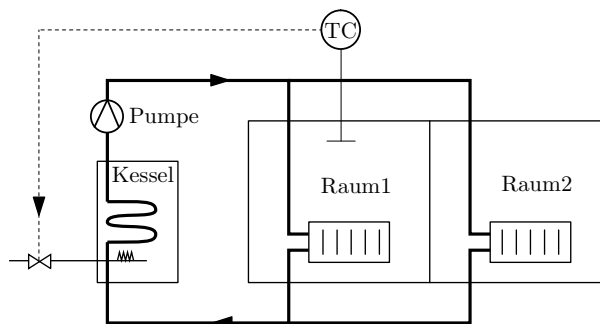


Abb. 1.5: Raumtemperaturregelung

#### Beispiel 1.5 Raumtemperaturregelung

Die Aufgaben der Gebäudeautomatisierung beziehen sich nicht nur auf große Gebäude, sondern sie treten auch im Haushalt auf. Abbildung 1.5 zeigt das technologische Schema des Temperaturregelkreises eines Raumes. Die Raumtemperatur wird durch den mit TC (*temperature control*) bezeichneten Regler

gemessen und mit dem vorgegebenen Sollwert verglichen. In Abhängigkeit von der Differenz zwischen der Soll- und der Isttemperatur wird der Brenner für eine längere oder kürzere Zeitdauer angeschaltet.

Man braucht eine solche Regelung, weil sich die Raumtemperatur auf Grund von Sonneneinstrahlung, Frischluftzufuhr oder der Anwesenheit von Personen verändert, was zusammen als Störung bezeichnet wird. Durch die Regelung kann die Temperatur annähernd konstant auf dem vorgegebenen Sollwert gehalten werden.

Für die spätere Behandlung kontinuierlicher und diskreter Systeme ist es interessant zu sehen, dass bei einer Raumtemperaturregelung beide Signalarten gleichzeitig auftreten. Die Raumtemperatur verändert sich kontinuierlich, aber der Brenner kann nur diskret geschaltet werden. Man kann jedoch eine einheitlich kontinuierliche Betrachtung dadurch einführen, dass man die Brennerleistung über einen bestimmten Zeitraum mittelt, wodurch sie einen kontinuierlichen Wert erhält.

Dieser Temperaturregelung übergeordnet ist die Festlegung des Sollwertes der Raumtemperatur. Aus Gründen der Energieeinsparung arbeitet man nicht mit konstanten Sollwerten, sondern legt in Abhängigkeit von der Nutzungsdauer der Räume einen Tagesgang fest, demzufolge die Solltemperatur in den Nachtstunden tiefer als am Tag liegt.

Bei großen Gebäuden werden die wichtigsten Daten in eine Warte übertragen, so dass das Gebäude dort manuell oder automatisch überwacht werden kann.  $\square$

## 1.2.4 Überwachung und Steuerung von Energiesystemen

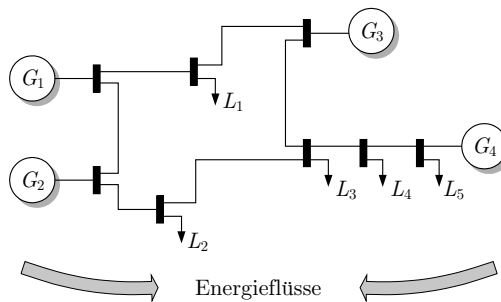
Energiesysteme versorgen eine große Anzahl von Abnehmern mit elektrischem Strom, Gas oder Wärme. Sie sind vernetzte Systeme, in denen mehrere Automatisierungsaufgaben gleichzeitig durch eine Vielzahl dezentraler und zentraler Automatisierungseinrichtungen zu lösen sind. Elektrische Spannungen und die Netzfrequenz, Gasdrücke und Durchflüsse, Vorlauftemperaturen von Wärmeversorgungssystemen und viele weitere Größen sind durch Regelungen von der Energieentnahme und von Störungen unabhängig konstant zu halten. Der Einsatz von Energieerzeugern ist zu planen und die Netze sind mit einer der voraussichtlichen Belastung angepassten Struktur zu betreiben. Für diese Steuerungsentscheidungen braucht das Bedienungspersonal umfangreiche Informationen.

Energieversorgungssysteme sind ein anschauliches Beispiel für kooperierende Systeme. Eine sichere Stromversorgung kann nur durch das Zusammenwirken vieler Erzeuger erreicht werden, so dass bei unvorhersehbaren Verbrauchsänderungen bzw. Erzeugungsausfällen das kurzzeitige Energiedefizit auf viele Teilsysteme verteilt und durch diese ausgeglichen werden kann.

### Beispiel 1.6 Überwachung und Steuerung von Elektroenergienetzen

Das in Abb. 1.6 dargestellte Elektroenergienetz zeigt mit den in der Energietechnik üblichen Symbolen ein Netz mit vier Generatoren  $G_1, \dots, G_4$ , die in unterschiedlichen Knoten Energie in das Verteilungsnetz einspeisen. Diese Energie wird durch das Netz über die dargestellten Leitungen zu fünf Verbrauchern  $L_1, \dots, L_5$  transportiert. Auf Grund der geografischen Lage der Energieerzeuger und der Verbraucher erfolgt in dem gezeigten Netz vorrangig ein Energiefluss von links nach rechts.

Energienetze müssen eine sichere und ausreichende Versorgung der Abnehmer mit Elektroenergie gewährleisten. Trotz wechselnder Belastung muss die Energie für den Verbraucher mit konstanter Spannung (230 V) und konstanter Frequenz (50 Hz) zur Verfügung stehen. Diese Zielstellung kann nur dann erreicht werden, wenn eine Regelung dafür sorgt, dass die erzeugte Leistung ständig an die verbrauchte Leistung angepasst wird.



**Abb. 1.6:** Energieflüsse in einem Elektroenergieverteilungsnetz

Diese Regelung realisiert jedoch nur die unterste Ebene einer hierarchisch gegliederten Automatisierungseinrichtung. Durch die höheren Ebenen muss erreicht werden, dass die Energie in der richtigen Menge am richtigen Ort erzeugt wird, um über das Verteilungsnetz zu den Verbrauchern geleitet zu werden. Die Planungskomponente liefert dafür eine Lastvorhersage (Tagesgang), die den zeitlichen Verlauf des Leistungsbedarfs für einen oder mehrere Tage im Voraus beschreibt. Dies ist die Grundlage für die Einsatzplanung, die den Bedarf für die Energieerzeugung und den Verlauf der Einspeisespannungen so festlegt, dass keine Übertragungsleitung überlastet wird und dass das Gesamtsystem in einem wirtschaftlich günstigen Arbeitspunkt arbeitet. Mit der Liberalisierung des Energiemarktes muss außerdem erreicht werden, dass die Energieflüsse im Netz die vertraglich gebundenen und sich zeitlich stark ändernden Energietransporte realisieren. Als Ergebnis entstehen die Sollwerte für die unterlagerten Spannungsregelkreise. □

### 1.2.5 Automatisierungstechnik in Fahrzeugen

Moderne Fahrzeuge sind mit einer Vielzahl von Steuergeräten ausgestattet, deren wichtigste Funktionen die Überwachung und Steuerung der Fahrzeugkomponenten betreffen. Mess- und Stellinformationen werden über umfangreiche Kabelbäume bzw. durch Bussysteme zwischen den Fahrzeugkomponenten und den Steuergeräten übertragen, wobei elektrische Signale auch frühere mechanische Kopplungen ersetzen, so dass man von „Brake-by-wire“- oder „Steer-by-wire“-Systemen spricht.

#### **Beispiel 1.7** Aufgaben der Motorsteuerung

Welche große Bedeutung die Automatisierungstechnik für den technologischen Wandel besitzt, soll am Beispiel der Motorsteuerung erläutert werden. Moderne Motoren sind mit einer Vielzahl von Sensoren und Aktoren ausgerüstet. Der vom Fahrer vorgegebene Gaspedalwinkel gibt das gewünschte Drehmoment an, das im Steuergerät des Motors unter Berücksichtigung von Emission, Komfort und Verbrauch in die erforderliche Einspritzmenge des Kraftstoffes umgerechnet wird (Aufg. 2.2 auf S. 62).

Der Fortschritt, der durch eine Direkteinspritzung von Kraftstoff in Verbindung mit einer automatischen Steuerung gegenüber der früher eingesetzten mechanischen Steuerung erreicht werden kann, wird aus der folgenden Überlegung offensichtlich. Bei alten Fahrzeugen wurde das Ansaugventil des Zylinders über Hebel mechanisch durch die Kurbelwelle geöffnet und geschlossen. Dadurch öffneten und schlossen sich die Ventile bei jeder Drehzahl an demselben Kurbelwellenwinkel. Da sich bei

Dieselmotoren das Luft-Kraftstoffgemisch selbstständig mit einem näherungsweise konstanten Zündverzögerung entzündet, die Kurbelwelle aber in Abhängigkeit von der Drehzahl in dieser Zeitspanne eine unterschiedliche Winkeländerung ausführt, beginnt die Verbrennung bei der mechanischen Steuerung je nach Drehzahl bei unterschiedlichen Kurbelwellenwinkeln und man kann keinen für alle Drehzahlen geltenden optimalen Zündzeitpunkt einstellen. Im Unterschied dazu kann mit einer elektronischen Steuerung der Beginn und das Ende des Einspritzens von Kraftstoff der Drehzahl, dem gewünschten Drehmoment sowie der Motortemperatur angepasst werden, wodurch ein wesentlich höherer Wirkungsgrad und eine geringere Schadstoffemission resultieren.

Weitere Komponenten der Motorsteuerung legen den Ladedruck, die Abgasrückführung und die Lehlaufrdrehzahl fest. Der Zündzeitpunkt wird bei modernen Steuerungen sogar der Kraftstoffqualität angepasst. □

**Sicherheitsfunktionen.** Wichtige Sicherheitseinrichtungen in modernen Fahrzeugen sind nur durch eine automatische Steuerung möglich. Zu den bekanntesten Komponenten gehören das Antiblockiersystem (ABS) (siehe Beispiel 7.7 und Aufgabe 11.23), die Antriebsschlupfregelung, die Fahrdynamikregelung, das elektronische Stabilisierungsprogramm (ESP) sowie die aktive Federung. Moderne Autos verfügen darüber hinaus über Geschwindigkeitsregler und überwachen den Abstand zum vorausfahrenden Fahrzeug.

#### **Beispiel 1.8** *Bremsassistent und Antiblockiersystem*

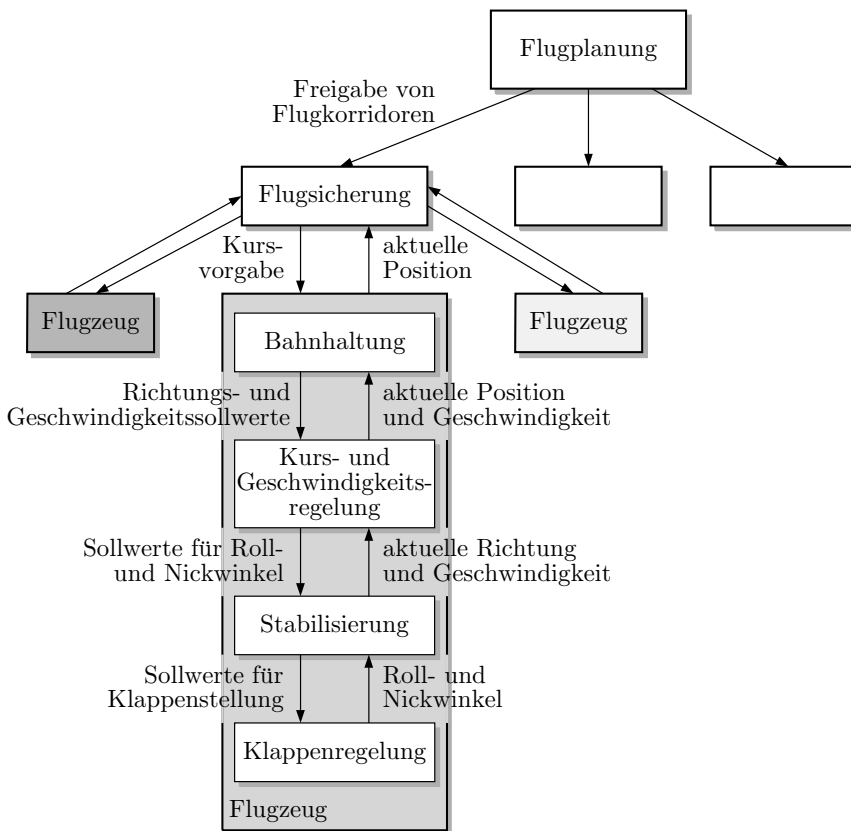
Wie umfangreich die Informationsverarbeitung bei sicherheitskritischen Funktionen sein kann, soll am Beispiel des Bremssystems erläutert werden. Elektronisch gesteuerte Bremssysteme sollen einerseits die Wirksamkeit der Bremse verbessern, um in Notsituationen eine größere Sicherheitsreserve zu schaffen, andererseits ein Blockieren der Räder verhindern. Während bei konventionellen Bremsen die Fußkraft auf dem Bremspedal über einen Bremskraftverstärker an die Bremsen übertragen wird, baut ein elektronischer Bremsassistent schon am Beginn des Bremsvorgangs in Sekundenbruchteilen die maximale Bremskraft auf und verkürzt dadurch den Anhalteweg.

Um diese Funktion zu erfüllen, muss das System den Fahrerwunsch zu bremsen erkennen. Aus der Geschwindigkeit, mit der das Bremspedal heruntergedrückt wird, berechnet es die Stärke, mit der der Fahrer bremsen möchte. Wenn diese Stärke einen bestimmten Grenzwert überschreitet, handelt es sich um eine Notbremsung und die Bremse wird unabhängig von der Fußkraft auf dem Bremspedal mit maximaler Kraft betätigt. Der Bremsassistent unterstützt den Fahrer also dadurch, dass er einen höheren Druck in der Bremsanlage erzeugt, als es der Pedalkraft des Fahrers entspricht.

Die maximal mögliche Bremswirkung ist durch den Kontakt der Reifen und der Fahrbahn begrenzt. Deshalb werden Antiblockiersysteme verwendet, die ein Blockieren der Räder dadurch verhindern, dass sie die vom Fahrer ausgeübte und gegebenenfalls durch einen Bremsassistenten verstärkte Bremskraft auf einen den aktuellen Straßenbedingungen angepassten Maximalwert beschränken. Das schwierigste dabei zu lösende Problem entsteht aus der Tatsache, dass sich die Straßenbedingungen nicht messtechnisch erfassen lassen, denn es gibt weder für das aktuelle Wetter noch für die Beschaffenheit der Straßenoberfläche einen geeigneten Sensor. Deshalb muss von der Reaktion des Rades auf die Fahrbahneigenschaften geschlossen werden. Trotz ABS kann die Bremskraft die maximal mögliche überschreiten und das Rad blockieren, weil das ABS erst nach einem Blockieren des Rades die aktuellen Straßenverhältnisse erkennt und dann angemessen reagieren kann. □

### 1.2.6 Überwachung und Steuerung des Flugverkehrs

Bei der Überwachung und Steuerung des Flugverkehrs wirken viele unterschiedliche Komponenten zusammen. Abbildung 1.7 zeigt die hierarchische Struktur, in der die Flugplanung und -sicherung die Bewegung der einzelnen Flugzeuge koordiniert. Durch die Flugplanung werden für einen geografisch großen Bereich Flugkorridore für bestimmte Flüge freigegeben. Die Flugsicherung überwacht die Nutzung dieser Flugkorridore durch die einzelnen Maschinen und koordiniert die Bewegung von Flugzeugen, die sich in demselben Gebiet bewegen. Den Flugzeugen wird in Abhängigkeit von der Position und dem Ziel aller Flugzeuge in diesem Gebiet der Kurs vorgegeben.



**Abb. 1.7:** Hierarchischer Aufbau der Überwachung und Steuerung des Flugverkehrs

Der Kurs gibt die Flughöhe und die Flugrichtung vor, die mit Hilfe geeigneter Steuereingriffe des Piloten und des Autopiloten einzuhalten sind. Dafür enthält die Abbildung Steuerungen auf vier Hierarchieebenen, wobei die in der Hierarchie jeweils höher angeordnete Steuerung der

untergeordneten die Sollwerte vorgibt. Es entstehen Wirkungskreise, die in der Abbildung aus jeweils zwei Blöcken bestehen und die mit den anderen Wirkungskreisen verkoppelt sind. So wirkt die Regelung, die das Flugzeug in der Luft stabilisiert, auf die Klappenregelung ein, indem sie dieser Regelung Sollwerte für die aktuelle Klappenposition vorgibt. In Gegenrichtung beeinflusst die Klappenregelung durch die mit der Klappenstellung verbundene Bewegung des Flugzeugs im Raum die Regelung für die Stabilisierung.

Die in der Abbildung gezeigten Hierarchieebenen unterscheiden sich bezüglich ihrer Aufgaben, wobei eine auch für andere Anwendungen der Automatisierungstechnik charakteristische Tendenz zu beobachten ist: Wenn man die Hierarchieebene von unten nach oben durchläuft, werden die Überwachungs- und Steuerungsziele globaler und die Zeithorizonte, in denen die Steuerungen reagieren müssen, länger. So muss die Klappenregelung im Millisekunden-takt arbeiten, während sich die Flugsicherung mit der Position der Flugzeuge im Minutentakt beschäftigt.

Menschen sind an mehreren Stellen in die Automatisierung des Flugverkehrs eingebunden. Sie haben vor allem Aufgaben zu übernehmen, deren Ziele sich zeitlich stark ändern bzw. von der aktuellen Situation des Flugverkehrs abhängen und sich deshalb nicht ohne Weiteres automatisieren lassen. Beispielsweise fliegen Flugzeuge heute vollautomatisch, nachdem sie ihre endgültige Flughöhe erreicht haben. Der Pilot beschäftigt sich dann nur noch mit der Überwachung des Fluges. Allerdings übernimmt er bei der Landung die Steuerung des Flugzeugs selbst, auch wenn heute große Flugzeuge vollautomatisch landen könnten.

### 1.2.7 Automatisierungsaufgaben in der Informations- und Kommunikationstechnik

Die Übertragung von Informationen unterschiedlichen Typs wie Bilder, Sprache und Texte über Kommunikationsnetze mit vielen Sendern und Empfängern und ständig wechselnden Sender-Empfänger-Zuordnungen umfasst viele steuerungstechnische Fragestellungen. Durch eine geeignete Steuerung soll erreicht werden, dass die Netzkapazität möglichst gut ausgenutzt wird. Dafür müssen die Warteschlangen bis in die Nähe ihrer physikalischen Begrenzung genutzt werden, aber es muss gleichzeitig verhindert werden, dass sie überlaufen und Datenpakete verloren gehen. Um dies zu erreichen, muss eine Steuerung die aktuellen Warteschlangenlängen auswerten und die Datenströme an den Quellen gegebenenfalls begrenzen.

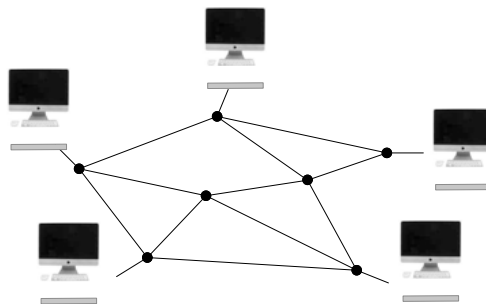


Abb. 1.8: Datenverbindungen über ein Rechnernetz

Die Steuerung muss den Datenverkehr im Netz im Sinne vorgegebener Qualitätskenngrößen (*quality of service (QoS)*) für den maximalen Datenverlust oder die maximale Verzögerungszeit bei der Übertragung optimieren. Ein weiteres wichtiges Kriterium für die Steuerung ist die Fairness, derzufolge alle Sender in angemessener Weise an der Nutzung der verfügbaren Übertragungskapazität beteiligt werden sollen.

Die Steuerung nutzt die Tatsache, dass die Datenverbindungen stets bidirektional aufgebaut werden, obwohl die Datenpakete nur in der Richtung vom Sender zum Empfänger zu transportieren sind. Knoten, deren Warteschlange die zulässige Größe überschreitet, berechnen maximale Raten, mit denen diejenigen Sender senden dürfen, deren Datenpakete zu diesen Knoten gelangen.

### 1.2.8 Zusammenfassung: Notwendigkeit der Automatisierung technischer Systeme

Die Beispiele haben gezeigt, dass die Automatisierungstechnik vielfältige Ziele erfüllen soll, diese jedoch im Wesentlichen den folgenden zwei Gruppen zugeordnet werden können:

- *Der Mensch soll durch eine Automatisierungseinrichtung ersetzt werden.*

Dies soll nicht nur die Sehnsucht der Menschen erfüllen, dass sich Aufgaben „von selbst“ erledigen, sondern den Menschen auch in gefährlichen Situationen, bei schwierigen oder ermüdenden Aufgaben, in nicht erreichbaren oder weit entfernten Orten ersetzen oder Anlagen automatisch steuern, weil der Mensch dazu gar nicht in der Lage wäre. In der zuletzt genannten Situation ist die Automatisierungseinrichtung also *funktionsnotwendig*, denn ein Mensch würde zu langsam oder zu ungenau reagieren. Beispiele sind die Positionierung des Schreibkopfes auf der Festplatte, die Steuerung des Laserstrahls bei der Herstellung integrierter Schaltkreise, die genaue Einstellung des Luftspaltes zwischen dem Fahrzeug und dem Fahrweg beim Transrapid oder die Realisierung mehrerer verkoppelter Regelkreise in verfahrenstechnischen Anlagen mit schnell ablaufenden chemischen Reaktionen, die der Mensch in der zur Verfügung stehenden Zeit nicht überschauen kann.

Zu dieser Gruppe von Aufgaben gehören viele Regelkreise, durch die wichtige Prozessgrößen auf konstanten Werten gehalten werden, durch die also beispielsweise die Kühltbertemperatur oder die Zimmertemperatur gegenüber äußeren Störungen unabhängig gemacht wird, der Fahrstuhl ohne Fahrstuhlführer fährt oder Flugzeuge auf Langstreckenflügen selbsttätig auf dem vorgegebenen Kurs gehalten werden. Diese Prozesse könnte ein Mensch steuern, aber dies wäre für den Betreffenden langweilig und ermüdend und würde darüber hinaus den Betrieb der Anlage verteuern. Zu dieser Gruppe gehören Automatisierungsaufgaben, die aus der Forderung abgeleitet sind, dass Geräte oder Anlagen so autonom arbeiten sollen, dass sie auch von Unkundigen benutzt werden können. Die Waschmaschine oder der heimische CD-Spieler sind Beispielsysteme, die weitgehend autonom arbeiten, nachdem ihrer Steuerung durch wenige Handgriffe die Zielvorgaben mitgeteilt wurden.



- *Automatisierungseinrichtungen sollen die Zuverlässigkeit, Sicherheit und Präzision einer Anlage erhöhen und die Produktionskosten senken.*

Die Anlage soll unabhängig von menschlichen Fehlern arbeiten. Diese Automatisierungseinrichtungen sind *güteverbessernd*, auch im Sinne von Sicherheits- oder Umweltvorschriften.

Bei dieser Aufgabengruppe soll die Automatisierungseinrichtung zusätzliche Kriterien erfüllen, die in der heutigen Zeit vor allem aus Sicherheits- oder Umweltschutzvorschriften resultieren oder die die Genauigkeit gegenüber einer manuell geführten Produktion erhöhen und die Kosten senken. Hierzu gehören viele Sicherheitsvorkehrungen. Ein Auto kann ohne Antiblockiersystem gefahren werden, aber mit einem solchen System ist es sicherer. Die Verbesserung der Anlage durch eine Automatisierungseinrichtung kann auch bedeuten, dass die Flexibilität bezüglich einer Produktumstellung erhöht wird. Eine Umstellung auf neue Aufgaben bedeutet dann nicht einen Umbau der Anlage, sondern im Wesentlichen nur eine Änderung der Steuerungsalgorithmen.

Die große Bedeutung, die die automatische Überwachung und Steuerung für die Funktion technischer Anlagen hat, wird oft übersehen. Wer denkt schon beim Besteigen eines Flugzeugs als erstes an die Servoregelung der Klappen und nicht an die mechanische Konstruktion? Auch beim Autokauf spielen Aussehen und Motorleistung eine wichtigere Rolle als die Leistungsfähigkeit des Antiblockiersystems, zumal man dessen Wirkung selten spürt. Die Automatisierung wirkt also oft im Verborgenen, was sehr treffend mit dem englischen Begriff *hidden technology* beschrieben wird. Aufgrund dieser Tatsache wird häufig verkannt, dass entscheidende Fortschritte in allen technologischen Bereichen ohne die Automatisierungstechnik unmöglich wären und der Automatisierungsgrad technischer Systeme ständig steigt.

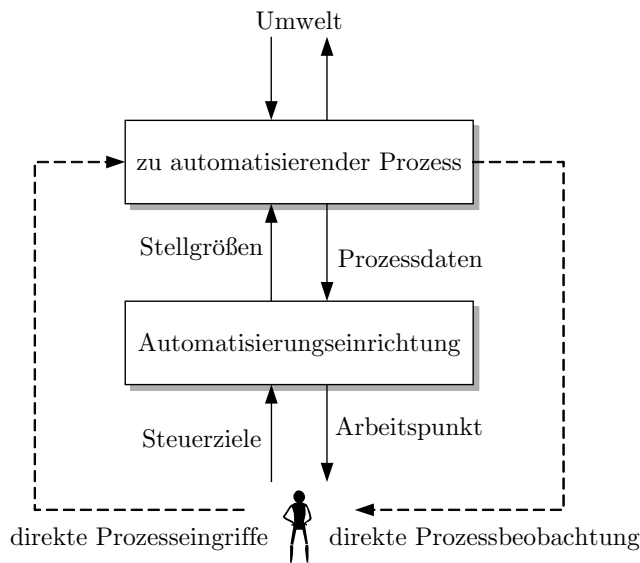
**Wirtschaftliche Ziele der Automatisierung.** Die genannten Automatisierungsziele wurden aus technischer Sicht formuliert. Es ist offensichtlich, dass diese Ziele unmittelbare betriebswirtschaftliche Wirkungen haben, u. a. die Einsparung von Personalkosten, Material, Energie und Herstellungszeit, die Steigerung der Quantität und Qualität der Produkte sowie die Erhöhung der Zuverlässigkeit, Verfügbarkeit und Lebensdauer der Anlage. Damit trägt die Automatisierung entscheidend zur Erfüllung globaler Ziele wie der Erhöhung der Wettbewerbsfähigkeit eines Unternehmens, der Steigerung der Energieeffizienz oder der Verbesserung des Umweltschutzes bei. In der industriellen Produktion sind es häufig diese Argumente, die die Automatisierung der Anlagen vorantreiben.

## 1.3 Grundstruktur automatisierter Systeme

### 1.3.1 Beziehungen zwischen der Automatisierungseinrichtung und dem zu automatisierenden Prozess

Die im Abschn. 1.2 aufgeführten Beispiele zeigen, dass bei der Lösung von Automatisierungsaufgaben stets mit derselben Grundstruktur gearbeitet wird. Entsprechend Abb. 1.9 besteht ein automatisiertes System aus zwei Komponenten: dem zu automatisierenden Prozess und der Automatisierungseinrichtung. Das zu automatisierende System wird häufig einfach als Prozess

oder auch als System bezeichnet oder, wenn es sich um Regelungs- oder Steuerungsaufgaben handelt, als Regelstrecke bzw. Steuerstrecke. Die Automatisierungseinrichtung heißt auch Automatisierungssystem, bei Regelungs- oder Steuerungsaufgaben auch Regeleinrichtung oder Regler bzw. Steuereinrichtung, Steuerung oder, wenn die technische Realisierung betont werden soll, Steuergerät. Andererseits gebraucht man die Begriffe der Regelung und Steuerung nicht nur für den Teil der Automatisierungseinrichtung, der aus den Messgrößen die Steuerbefehle erzeugt, sondern auch für die Vorgänge des Regelns und des Steuerns, bei denen sowohl die Regelstrecke bzw. Steuerstrecke und der Regler bzw. die Steuereinrichtung beteiligt sind. Die zusammengehörigen Begriffe werden im Folgenden synonym gebraucht, wobei aber bei Beispielen die in den betrachteten Anwendungsgebieten geläufigen Begriffe bevorzugt verwendet werden.



**Abb. 1.9:** Grundstruktur automatisierter Systeme

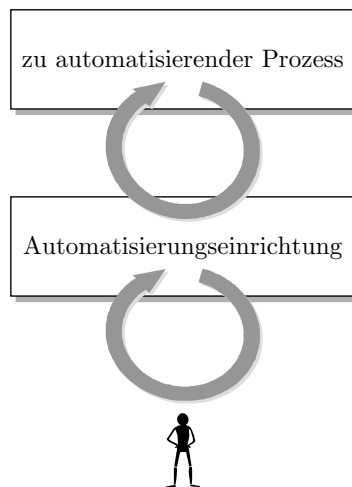
In dem zu automatisierenden Prozess wird Energie oder Materie umgewandelt, was man sich leicht an einem Kraftwerksblock oder einer Werkzeugmaschine vorstellen kann. Für die Automatisierungseinrichtung sind nicht diese Prozesse im Einzelnen, sondern nur die Informationen interessant, die diese Prozesse beschreiben. Einerseits werden Prozessdaten, also insbesondere *Messwerte* von wichtigen Prozessgrößen, an die Automatisierungseinrichtung übertragen. In anderer Richtung wirkt die Automatisierungseinrichtung auf den Prozess durch *Stellgrößen* ein, die beispielsweise eine Ventilstellung verändern oder einen Motor schalten.

Eine wichtige Charakteristik der Automatisierungstechnik wird aus dieser Grundstruktur offensichtlich:

|| Mit den Methoden der Automatisierungstechnik werden nicht nur gegebene technische Systeme analysiert, sondern es wird mit der Automatisierungseinrichtung eine neue Komponente geschaffen, die die Systeme überwacht und steuert.

### 1.3.2 Das Rückkopplungsprinzip

Das wichtigste Grundprinzip der Automatisierungstechnik besteht in der Schaffung von Rückkopplungen (Abb. 1.10). In der Grundstruktur automatisierter Systeme gibt es zwei Rückkopplungen, die das zu automatisierende System mit der Automatisierungseinrichtung bzw. die Automatisierungseinrichtung mit dem menschlichen Operator verbindet.



**Abb. 1.10:** Zwei Arten von Rückkopplungen in automatisierten Systemen

Durch den oberen Kreis werden Informationen, die im Prozess durch Sensoren erfasst werden, über die Automatisierungseinrichtung an den Prozess zurückgekoppelt. Die Stellgrößen beeinflussen das Prozessverhalten und damit auch die Prozessdaten, mit deren Hilfe die nächsten Stellgrößen berechnet werden. Diese Rückkopplung läuft vollautomatisch ab. Sie ist im kontinuierlichen Regelkreis fortwährend wirksam, während sie im diskreten Regelkreis nur zu den Zeitpunkten aktiv ist, an denen die Prozessdaten eine Änderung der diskreten Ausgangsgrößen anzeigen.

Der untere Kreis entsteht durch die Eingriffe des Menschen, der das Prozessverhalten bewertet und daraus Steuerungsentscheidungen ableitet. Diese Rückkopplung wirkt nur zu den unregelmäßigen Zeitpunkten, an denen der Mensch in den Prozess eingreift.

Beide Rückkopplungen unterscheiden sich vor allem bezüglich der Informationen, auf die sie sich beziehen, und bezüglich der Geschwindigkeit, mit der beide Kreise wirksam sind. Der automatisch wirkende Kreis arbeitet vor allem mit den numerischen oder binären Mess- und Stellgrößen im Millisekunden- bis Minutentakt, während die Rückkopplung über das Bedienpersonal auf einer höheren Abstraktionsebene und in größeren Zeitabständen erfolgt. Dort spielen vor allem eine globale Beschreibung des aktuellen Betriebszustands und globale Steuerungsziele eine wichtige Rolle.

Viele Methoden der Automatisierungstechnik beruhen auf dem Verständnis von Rückkopplungsstrukturen. Derartige Strukturen werden in vielfältiger Form in den folgenden Kapiteln behandelt. Der Informationskreislauf bringt neue Systemeigenschaften hervor, ohne dass die im gesteuerten System ablaufenden Wirkungsmechanismen verändert werden. Das Automatisierungsobjekt wird also nicht „umkonstruiert“, sondern von außen so beeinflusst, dass es ein vorgegebenes Verhalten zeigt. So kann durch einen Regler ein instabiles System stabilisiert werden und es können die dynamischen Eigenschaften einer Anlage wesentlich verändert werden. Diese Tatsache weist auch darauf hin, dass man das Verhalten eines automatisierten Systems nicht allein aus den Eigenschaften der gesteuerten Anlage oder der Automatisierungseinrichtung erklären kann, sondern sich die Analyse stets auf das Gesamtsystem beziehen muss.

### **Beispiel 1.9**   *Aktive Federung*

Ein Beispiel aus der Fahrzeugtechnik, das wichtige Eigenschaften automatisierter Systeme illustriert, ist die aktive Schwingungsdämpfung der Räder. Während man bei einer mechanischen Federung die Fahreigenschaften des Fahrzeugs, die maßgeblich durch die Dämpfung der Radaufhängung bestimmt werden, nur durch eine konstruktive Veränderung des Fahrwerks beeinflussen kann, kann eine aktive Federung durch eine Parameteränderung der Steuerung an die aktuelle Fahrsituation angepasst werden.

Bei der aktiven Schwingungsdämpfung wird der Öldruck, der die Federwirkung beeinflusst, über ein Ventil gesteuert. Durch Messung des Abstandes des Fahrzeugaufbaus von der Straßenoberfläche, der aktuellen Fahrgeschwindigkeit, der Federwege und der Beschleunigung kennt man das aktuelle Fahrverhalten und kann durch geeignete Verstellung des Ventils die Fahrzeugbewegung innerhalb der durch die Konstruktion vorgegebenen Grenzen stark verändern. Moderne System reagieren dabei sogar auf den durch eine Kamera erfassten zukünftigen Fahrweg. Statt das Fahrzeug umzukonstruieren, wird das Fahrverhalten durch eine Rückführsteuerung beeinflusst. Fahrzeuge, bei denen man während der Fahrt zwischen einem sportlichen und einem komfortablen Fahrwerk wählen kann, sind ein anschauliches Beispiel für die Flexibilität automatisierter Systeme. □

### **1.3.3 Die Rolle des Menschen in automatisierten Systemen**

Nur in seltenen Fällen übernimmt die Automatisierungseinrichtung die Steuerung einer Anlage vollständig und der Mensch wird für den Betrieb der Anlage nicht mehr gebraucht. Viel häufiger läuft die Automatisierung auf eine Arbeitsteilung hinaus, bei der wichtige Teilprozesse automatisch ablaufen und der Mensch den Gesamtprozess nach übergeordneten Gesichtspunkten wie Effektivität oder Sicherheit führt. Beispielsweise gibt es in der verfahrenstechnischen

Industrie hunderte von Regelkreisen, aber die Gesamtanlage wird von menschlichen Operateuren gesteuert, die in einer Warte mit allen notwendigen Informationen versorgt werden und die die Anlage überwachen und entsprechend der aktuellen Produktionsziele beeinflussen.

Um diesen Unterschied zwischen der Tätigkeit des Menschen und der Automatisierungseinrichtung hervorzuheben, macht man (vor allem in der Prozessautomatisierung) auch begriffliche Unterschiede:

- Tätigkeiten der Automatisierungseinrichtung werden als überwachen, alarmieren, melden, steuern, regeln oder optimieren bezeichnet,
- Tätigkeiten des menschlichen Operators als leiten, führen oder bedienen.

Außer den bisher betrachteten Informationskopplungen kann es in einzelnen Anlagen direkte Verbindungen zwischen der Anlage und dem Operator geben, beispielsweise, wenn sich der Anlagenfahrer vor Ort ein Bild von der Anlage verschafft und dabei über die durch Prozessdaten vermittelten Informationen hinausgehende Informationen „sieht, riecht oder hört“. Auch kann er vor Ort an Stellhebeln oder Ventilen direkt auf den Prozess einwirken. Diese Möglichkeiten sind zwar stets vorhanden; sie werden jedoch i. Allg. nur in Havariesituationen genutzt und sollen im Weiteren nicht betrachtet werden.

### 1.3.4 Spezielle Klassen automatisierter Systeme

**Reaktive und eingebettete Systeme.** Die in Abb. 1.9 angegebene Grundstruktur automatisierter Systeme findet man auch in angrenzenden Fachgebieten, in denen jedoch meist mit anderen Begriffen gearbeitet wird. Der in der Informatik gebräuchliche Begriff des *reaktiven Systems* (rückwirkendes System) beschreibt eine Automatisierungseinrichtung, denn man versteht darunter einen Rechner, der auf die von einem physikalischen Prozess erzeugten Ereignisse reagiert und auf diesen Prozess zurückwirkt. An Stelle von Prozessdaten spricht man bei reaktiven Systemen von Ereignissen, an Stelle von Stellgrößen von Aktionen.

Eng damit verwandt ist der Begriff der *eingebetteten Systeme*, mit dem man Rechner bezeichnet, die auf ihre Umgebung reagieren, also die durch die Umgebung vorgegebenen Aufgaben lösen und die Ergebnisse an die Umgebung übermitteln. Automatisierungseinrichtungen sind eingebettete Systeme, wie man aus den entsprechenden Blockschaltbildern erkennt. Sie müssen unter Echtzeitbedingungen arbeiten und in der Lage sein, auf Störungen zu reagieren.

Für die Vernetzung physikalischer Prozesse mit Automatisierungseinrichtungen, durch die beispielsweise Sensordaten oder Stellgrößen übertragen werden, werden seit langer Zeit Bussysteme eingesetzt. Die technologische Entwicklung erlaubt gegenwärtig auch die Verwendung von Funknetzen, mit Hilfe derer neue Automatisierungsaufgaben realisiert werden können. So kann die Überwachung des Autobahnverkehrs mit Hilfe von Einzelfahrzeugdaten, die von Fahrzeugen über Funkverbindungen zur Verkehrsleitzentrale übertragen werden, zu wesentlich besseren Ergebnissen führen als die alleinige Verwendung der Daten ortsfester Sensoren. Das Gebiet der *vernetzten Regelungssysteme* beschäftigt sich mit Fragen, die die Echtzeitfähigkeit dieser Informationsverbindungen bzw. die aufgabenabhängige Nutzung von Kommunikationsnetzen für Automatisierungsaufgaben betreffen.

Zukünftig wird die digitale Vernetzung weiter zunehmen und zu sogenannten *Cyber-physical systems* führen. Der Cyber-Teil besteht aus den eingebetteten Systemen, die durch

globale Netze verbunden sind, der physikalische Teil aus den zu steuernden und zu überwachenden Prozessen.

**Mechatronische Systeme.** Das Gebiet der *Mechatronik* hat eng mit der hier beschriebenen Grundstruktur automatisierter Systeme zu tun, denn in diesem Gebiet werden mechanische Systeme mit elektronischen Systemen so weit verknüpft, dass sie nicht nur gemeinsam entworfen, sondern auch gemeinsam technisch realisiert werden. Die elektronischen Systeme übernehmen dabei auch die Steuerungsfunktionen.

Viele Beispiele findet man in der Fahrzeugtechnik, wo Regelkreise durch direkte Verbindungen von Sensoren und Aktoren „vor Ort“ (und nicht in externen Steuergeräten) realisiert werden. Bei intelligenten Einzelantrieben müssen Drehmomente bzw. Kräfte für Einzelbewegungen unter Nutzung elektromagnetischer Prinzipien aufgebaut werden, wofür die Steuerung bestimmte Ströme bzw. Spannungen vorschreibt und diese Größen beim Einwirken von Störungen so verändert, dass die gewünschten Drehmoment- oder Positionswerte eingehalten werden.

Mechatronische Systeme bestehen deshalb aus mechanischen Komponenten, die mit elektronischen Einheiten funktionell und konstruktiv integriert sind. Der Begriff der Mechatronik fasst die Verknüpfung der drei beteiligten Disziplinen zusammen: die Mechanik für den konstruktiven Aufbau des Systems, die Automatisierungstechnik für die Festlegung der zu realisierenden Steuerungsmethode und die Elektronik für die Realisierung von Steuerungen.

#### **Aufgabe 1.2\***    *Automatisierungstechnik im täglichen Leben*

Überlegen Sie sich, welche Funktionen die Steuergeräte in den folgenden Geräten oder Anlagen haben und welche Funktionen durch einen Bediener ausgeführt werden müssten, wenn es diese Steuergeräte nicht gäbe. Wäre ein Betrieb des Gerätes bzw. der Anlage ohne automatische Steuerung überhaupt möglich?

1. Verkaufsautomat für Fahrkarten, Bankautomat
2. Videorekorder, Digitalkamera
3. Waschmaschine
4. Heizungsanlage
5. Kopiergerät
6. Spielautomat.

Nennen Sie Prozesse, die heute noch nicht automatisiert sind. Welche Voraussetzungen fehlen heute noch, um diese Prozesse zu automatisieren? ☐

#### **Aufgabe 1.3\***    *Automatiktoaster und andere automatische Geräte*

Inwiefern arbeitet ein Automatiktoaster „automatisch“? Was haben eine automatische Schranke der Eisenbahn, der Kaffeeautomat, das Automatikgetriebe im Fahrzeug und eine automatische Klimaanlage mit Automatisierungstechnik zu tun? ☐

**Aufgabe 1.4** *Rezeptsteuerung in der Bäckerei*

Schreiben Sie in Form eines Programmablaufplanes auf, welche Prozesse der Steuerungsalgorithmus eines Backautomaten einleiten muss, um Kuchen herzustellen. Welche Informationen (Messwerte) muss die Steuerung auswerten, um den Zeitpunkt erkennen zu können, an dem der nächste Prozessschritt eingeleitet werden muss? □

## 1.4 Automatisierungsaufgaben

Die Vielfalt der Anwendungsgebiete der Automatisierungstechnik lässt zunächst vermuten, dass für jedes dieser Gebiete spezielle Steuerungs- und Überwachungsmethoden entwickelt worden sind und ein Automatisierungstechniker alle diese Methoden erlernen muss. Diese Vermutung ist (glücklicherweise) falsch. Da es möglich ist, die grundlegenden automatisierungstechnischen Fragestellungen weitgehend losgelöst von der konkreten technischen Anwendung aus einer systemtheoretischen Sicht zu formulieren, kann man für sie Methoden entwickeln, die nicht nur in einem, sondern in mehreren Anwendungsgebieten gleichzeitig anwendbar sind. NORBERT WIENER<sup>1</sup> hat deshalb für die Steuerungstheorie den Begriff *Kybernetik* geprägt, der im Griechischen soviel wie die „Kunst des Steuerns“ heißt. WIENER hat auch darauf hingewiesen, dass diese Methoden nicht nur für technische, sondern auch für biologische Systeme gelten und dass man Steuerungsprinzipien der Biologie in die Technik übertragen kann. Für die Automatisierung technischer Prozesse gebraucht man deshalb auch den Begriff *Technische Kybernetik*.

Grundlage für diese allgemeine Theorie ist die systemtheoretische Betrachtung, die in den nachfolgenden Kapiteln eingeführt und erläutert wird. Bei dieser Betrachtung wird von den Stoff- und Energieflüssen abstrahiert und der Prozess einheitlich durch die das Prozessgeschehen beschreibenden Informationsflüsse dargestellt. Die Prozesskomponenten werden deshalb als Blöcke und die Signale durch Pfeile symbolisiert. Dabei entstehen die in der Automatisierungstechnik weit verbreiteten Blockschaltbilder (Kap. 2).

In den nachfolgenden Abschnitten werden die Grundaufgaben der Automatisierungstechnik anhand einfacher Blockschaltbilder erläutert, in denen wie in Abb. 1.9 das zu steuernde System stets oben und die zum Automatisierungssystem gehörenden Komponenten unten angeordnet sind. Die einzelnen Aufgaben unterscheiden sich in der Art der verwendeten Informationen, in der Richtung des Informationsflusses sowie in der Art und Weise, wie Modelle des zu steuernden Prozesses genutzt werden.

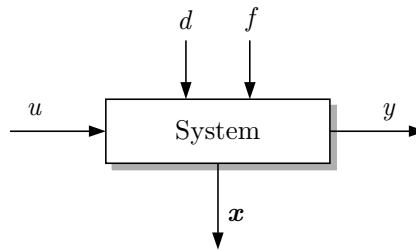
Wenn man die Blockschaltbilder vom Standpunkt der Implementierung der Automatisierungsfunktionen betrachtet, erkennt man die Architektur der verwendeten Steuerungen. Blockschaltbilder der Automatisierungstechnik sind also eng verwandt mit den Strukturbildern der Informatik, die die Gliederung von Programmen veranschaulichen. Beide Darstellungsformen lassen erkennen, welche Komponenten welche Informationen benötigen und woher sie diese Informationen erhalten. Die nachfolgenden Bilder werden insbesondere zeigen, dass für die Lösung von Überwachungsaufgaben der Informationsfluss von dem zu automatisierenden Prozess zur Automatisierungseinrichtung verläuft, während er für Steuerungen und Regelung von

<sup>1</sup> NORBERT WIENER (1894 – 1964), amerikanischer Mathematiker

der Automatisierungseinrichtung zum Prozess gerichtet ist bzw. eine Informationsrückkopplung realisiert.

### 1.4.1 Modellbildung dynamischer Systeme

Grundlage für die Lösung von Automatisierungsaufgaben ist ein dynamisches Modell des zu automatisierenden Prozesses. Dieses Modell soll das Prozessverhalten unter der Wirkung aller Einflussgrößen beschreiben, wobei man die folgenden Signaltypen unterscheidet (Abb. 1.11):



**Abb. 1.11:** Definition der an einem Prozess wirkenden Signale

- **Stellgrößen**  $u(t)$  sind diejenigen Eingangssignale, deren Verlauf durch das Automatisierungssystem oder den menschlichen Operator festgelegt werden kann, um das Prozessverhalten zielgerichtet zu beeinflussen.
- **Störgrößen**  $d(t)$  sind Eingangssignale, über die die Umwelt den Prozess beeinflusst. Sie können möglicherweise gemessen, aber nicht vom Automatisierungssystem beeinflusst werden.
- **Fehler**  $f(t)$  sind innere Veränderungen, die als Eingangssignale interpretiert werden. Sie bewirken unzulässige Abweichungen des Prozessverhaltens vom Nominalverhalten. Im Unterschied zu den gleichfalls nicht beeinflussbaren Störsignalen haben Fehler typischerweise größere, bleibende Wirkungen, die nicht durch eine Regelung ausgeglichen werden können.
- **Zustandsgrößen**  $x_i(t)$  kennzeichnen diejenigen inneren Prozesssignale, die für eine eindeutige Vorhersage des zukünftigen Verhaltens bekannt sein müssen. Der Systemzustand ist  $i$ . Allg. nicht messbar, obwohl mehrere oder alle Zustandsgrößen die messbare Ausgangsgröße  $y$  beeinflussen. Der Zustand eines dynamischen Systems ist ein Vektor  $x$ , der alle Zustandsgrößen  $x_i$  enthält.
- **Ausgangsgrößen**  $y(t)$  sind die (häufig messbaren) Signale, über die der Prozess von außen beobachtet werden kann und über die der Prozess seine Umgebung beeinflusst.

Obwohl bei den meisten Automatisierungsaufgaben jeweils mehrere Signale desselben Typs auftreten, können die grundlegenden Fragestellungen und Methoden anhand von Systemen erläutert werden, bei denen nur wenige Signale vorhanden sind. Es werden häufig Eingrößensysteme betrachtet, die lediglich eine Stellgröße  $u(t)$  und eine Ausgangsgröße  $y(t)$  besitzen. Bei



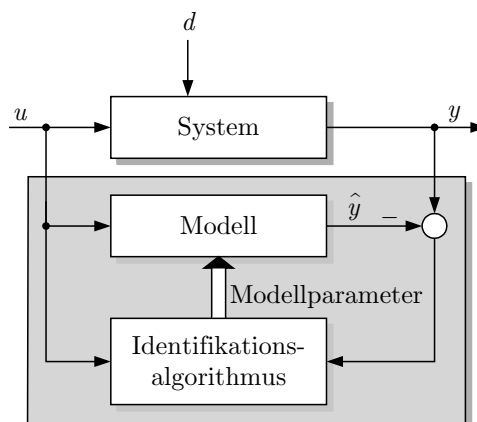
größeren Systemen können alle Stellsignale zu einem Vektor  $u(t)$  und alle Ausgangssignale zum Vektor  $y(t)$  zusammengefasst werden. Der Zustand  $x(t)$  ist auch bei „kleinen“ Systemen i. Allg. ein Vektor.

Die hier eingeführten Symbole sind diejenigen, die später für wertkontinuierliche Signale gebraucht werden. Für wertdiskrete Systeme werden andere Symbole verwendet, nämlich für den Eingang  $v$ , den Ausgang  $w$ , den Zustand  $z$ , den Fehler  $e$  und die Störung  $p$ . Die nachfolgenden Betrachtungen gelten jedoch für beide Systemklassen gleichermaßen, auch wenn der Übersichtlichkeit wegen nur die für kontinuierliche Systeme eingeführten Symbole benutzt werden.

**Modellbildung.** Als Modellbildung (oder Modellierung) bezeichnet man die Schritte, durch die man für ein technisches System ein dynamisches Modell aufstellt. Für die meisten Automatisierungsaufgaben ist dies der erste Lösungsschritt, weshalb das Thema Modellbildung im Gebiet der Automatisierungstechnik eine sehr große Bedeutung hat. Prinzipiell unterscheidet man zwei Wege: die theoretische Prozessanalyse und die experimentelle Prozessanalyse (oder Identifikation).

Bei der *theoretischen Prozessanalyse* wird das Modell durch das Aufschreiben der physikalischen Beziehungen erhalten, die in dem betrachteten System gelten. Bei kontinuierlichen Systemen entsteht dabei eine Menge von algebraischen Gleichungen und Differentialgleichungen, die man zu einer Differentialgleichung mit den Größen  $u(t)$  und  $y(t)$  oder zu einem Zustandsraummodell zusammenfassen kann. Die dafür notwendigen Modellbildungsschritte werden ausführlich im Kap. 3 behandelt.

Bei diskreten Systemen führt die Modellbildung auf Automaten oder Petrinetze. Dabei geben die in dem System wirkenden physikalischen Gesetze ebenfalls Hinweise darauf, aus welchen Komponenten das Modell aufgebaut ist, aber es gibt für die meisten Systeme keinen direkten Zusammenhang zwischen den physikalischen Gesetzen und den diskreten Zustandsübergängen des Automaten bzw. des Petrinetzes. Die Modellbildung diskreter Systeme wird im Kap. 11 erläutert.



**Abb. 1.12:** Identifikation dynamischer Systeme

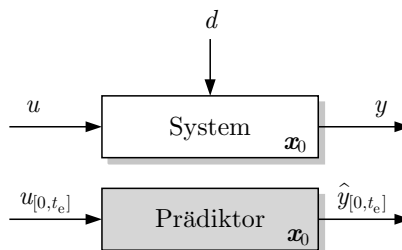
Bei der *Identifikation* macht man für das Modell einen Ansatz, bei dem man beispielsweise annimmt, dass das betrachtete System durch eine Differentialgleichung mit einer bestimmten Ordnung beschrieben wird. Die Koeffizienten der Differentialgleichung werden dann aus Messreihen für  $u(t)$  und  $y(t)$  ermittelt. Voraussetzung dafür ist, dass man mit dem System Experimente durchführen kann. Das prinzipielle Vorgehen bei der Identifikation kann man sich so vorstellen, dass ein Identifikationsalgorithmus die Modellparameter solange verändert, bis die Differenz  $y(t) - \hat{y}(t)$  zwischen dem gemessenen Systemausgang  $y$  und dem Modellausgang  $\hat{y}$  näherungsweise verschwindet (Abb. 1.12).

Die Identifikation ist für kontinuierliche Systeme ein sehr gut ausgearbeitetes Gebiet der Automatisierungstechnik, auf das hier nicht eingegangen werden kann. Es wird lediglich im Abschn. 4.4 auf eine Methode der Kennwertermittlung hingewiesen, mit der aus Experimentdaten die Parameter eines sehr einfachen Modells ermittelt werden.

Bei beiden Arten der Modellbildung kommt es darauf an, die physikalischen Vorgänge in dem zu automatisierenden System systemtheoretisch zu beschreiben. Das heißt, dass von den physikalischen Vorgängen abstrahiert und der Informationsfluss durch das System mathematisch erfasst wird. Dabei ist eine theoretische Prozessanalyse bereits bei der Planung einer Anlage möglich, die experimentelle Prozessanalyse jedoch erst nach der Inbetriebnahme, wobei sie vor allem für die Verbesserung der Modellgenauigkeit bzw. für die Anpassung der Modelle an veränderte Anlagenparameter eine besondere Bedeutung hat.

### 1.4.2 Vorhersage des Systemverhaltens

Grundlage vieler Automatisierungsfunktionen ist die Vorhersage (Prädiktion) des zukünftigen Systemverhaltens. Für den über einem Zeitintervall  $[0, t_e]$  gegebenen Verlauf der Eingangsgröße  $u$ , der mit  $u_{[0, t_e]}$  bezeichnet wird, soll der Verlauf  $y_{[0, t_e]}$  der Ausgangsgröße über demselben Zeitintervall bestimmt werden. Wie Abb. 1.13 zeigt, ist dafür auch die Kenntnis des Systemzustands  $x_0$  zu Beginn des Vorhersageintervalls ( $t = 0$ ) erforderlich. Es ist außerdem aus dem Blockschaltbild erkennbar, dass die Vorhersage nur dann zutreffen kann, wenn die Störung  $d$  hinreichend klein ist, denn diese nicht messbare Einflussgröße des Systems kann bei der Vorhersage nicht berücksichtigt werden.



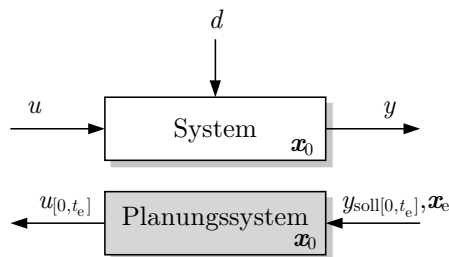
**Abb. 1.13:** Vorhersage des Systemverhaltens

Vorhersageaufgaben werden einerseits beim Entwurf einer Steuerung gelöst, um die Wirkung des ausgewählten Steuerungsalgorithmus auf den Prozess zu bewerten. Dabei wird die Vorhersageaufgabe vor der Inbetriebnahme der Steuerung (offline) für einen Zeithorizont  $[0, t_e]$  betrachtet. Es besteht keine direkte Informationskopplung zwischen dem Prozess und dem Prädiktor. Auf diese Situation der Vorhersage konzentrieren sich die Kap. 4 und 12.

Andererseits spielt die Vorhersage des Systemverhaltens bei operativen Steuerungsaufgaben eine große Rolle, bei denen das Bedienpersonal während des Betriebs einer Anlage die Steuerungsentscheidung in Abhängigkeit von der aktuellen Arbeitsweise fällen muss und vorher mit einem dynamischen Modell testen will, ob der zur Zeit  $t$  für das Zeitintervall  $[t, t + t_e]$  ausgewählte Steuereingriff  $u_{[t, t+t_e]}$  das zukünftige Systemverhalten in der beabsichtigten Weise verändert. Diese Vorhersageaufgabe muss während des Betriebs der Anlage (online) gelöst werden, wobei eine Informationskopplung den aktuellen Systemzustand  $x(t)$  an den Prädiktor übermittelt.

### 1.4.3 Planung von Steuereingriffen

In vielen technischen Anwendungen will man ein System so steuern, dass es vorgegebene Güteanforderungen in bestmöglicher Weise erfüllt. So soll eine Fertigung in möglichst kurzer Zeit beendet sein, ein Raumflugkörper mit möglichst wenig Energie sein Ziel erreichen, ein Fahrstuhl in möglichst kurzer Zeit alle Fahrtwünsche erfüllen, ein Kraftwerksblock in möglichst kurzer Zeit angefahren werden usw. Vorgegeben sind bei diesen Aufgaben ein gewünschter Verlauf  $y_{[0, t_e]}$  der Ausgangsgröße oder ein gewünschter Endzustand  $x(t_e) = x_e$  sowie Güteforderungen, die durch eine geeignet gewählte Steuerung bestmöglich erfüllt werden sollen. Diese Aufgabe beinhaltet, im Prinzip, eine Invertierung des Systems, denn der Ausgang  $y_{[0, t_e]}$  ist vorgegeben und die Steuerung  $u_{[0, t_e]}$  ist gesucht (Abb. 1.14).



**Abb. 1.14:** Planung eines Steuereingriffs

Planungsaufgaben werden häufig in Optimierungsprobleme überführt, wobei das Gütefunktional den Abstand des Verlaufs der Ausgangsgröße  $y$  vom gewünschten Verlauf  $y_{soll}$ , die Differenz des erreichten vom vorgegebenen Endzustand sowie den Verlauf der Stellgröße  $u_{[0, t_e]}$  bewertet. Die erhaltene Lösung erzeugt jedoch nur dann den gewünschten Ausgang bzw. Endzustand, wenn die Störung  $d_{[0, t_e]}$ , die bei der Planung nicht berücksichtigt werden kann, einen hinreichend kleinen Einfluss auf das Systemverhalten hat.

Bei der Planungsaufgabe besteht keine Informationskopplung zwischen dem Prozess und der Automatisierungseinrichtung, denn die Planungsaufgabe wird vor der Durchführung des betreffenden Prozesses gelöst. Es muss allerdings der Anfangszustand  $x_0$  bekannt sein.

#### 1.4.4 Zustandsbeobachtung

Alle Automatisierungsaufgaben sind umso besser zu lösen, je mehr man über den inneren Zustand  $x$  des Prozesses weiß. Der Zustand  $x(t)$  kann jedoch nicht aus dem zu einem Zeitpunkt  $t$  erhaltenen Messwert  $y(t)$  berechnet werden, was allein aus der Tatsache folgt, dass der Zustand  $x$  stets ein Vektor mit mehreren Signalwerten, die Messgröße bei den hier untersuchten Beispielen jedoch nur ein Skalar ist. Es ist jedoch möglich, aus dem Verlauf  $u_{[0,t_e]}$  der Eingangsgröße und dem Verlauf  $y_{[0,t_e]}$  der Ausgangsgröße unter Verwendung eines Prozessmodells den inneren Zustand zu rekonstruieren. Dieses Vorgehen nennt man Zustandsbeobachtung.

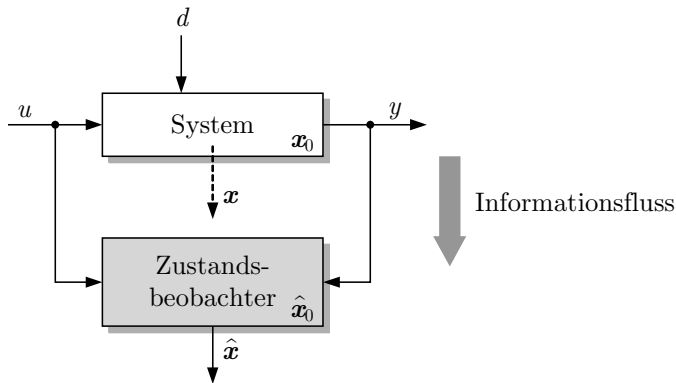


Abb. 1.15: Zustandsbeobachtung

Wie Abb. 1.15 zeigt, verwendet der Zustandsbeobachter einen Schätzwert  $\hat{x}_0$  für den unbekannten Anfangszustand  $x_0$  des Prozesses und verbessert diesen Schätzwert durch Auswertung des Verlaufs der Eingangs- und der Ausgangsgröße, so dass der Fehler  $\|x(t) - \hat{x}(t)\|$  immer kleiner wird und für  $t \rightarrow \infty$  verschwindet. Voraussetzung dafür ist jedoch, dass die Störung  $d$  klein genug ist, denn die Störung ist eine für den Zustandsbeobachter unbekannte Größe.

Bei der hier behandelten Beobachtungsaufgabe wird davon ausgegangen, dass die Störung  $d$  deterministisch ist. Wenn  $d$  ein stochastisches Signal ist, spricht man von einer *Zustandsschätzung*. Ein bekannter Zustandsschätzer ist das Kalmanfilter, das hier nur erwähnt wird, um auf die Ähnlichkeit der Aufgabenstellungen bei deterministischer und stochastischer Störung hinzuweisen.

Die Zustandsbeobachtung ist eine wichtige Teilaufgabe der Prozessüberwachung. Mit dem berechneten Zustand kann beispielsweise überprüft werden, ob einzelne Zustandsvariablen vorgegebene Grenzwerte unter- bzw. überschreiten. Der Informationsfluss führt vom Prozess zum

Zustandsbeobachter, aber es gibt keine Rückwirkung vom Beobachter auf den Prozess. Methoden der Zustandsbeobachtung werden für kontinuierliche Systeme im Kap. 9 und für ereignisdiskrete Systeme im Kap. 15 behandelt.

### 1.4.5 Prozessdiagnose

Durch eine Fehlerdiagnose wird ermittelt, ob ein gegebener technischer Prozess fehlerfrei arbeitet bzw. welche Fehler aufgetreten sind, um gegebenenfalls Instandhaltungsmaßnahmen einzuleiten. Da Fehler zu einer Veränderung des Eingangs-Ausgangsverhaltens (E/A-Verhaltens) des betreffenden Systems führen, kann man eine Diagnoseaufgabe dadurch lösen, dass man das aktuelle E/A-Verhalten, das sich in den Messwerten des Eingangs  $u$  und des Ausgangs  $y$  äußert, mit dem E/A-Verhalten vergleicht, das ein Modell der fehlerfreien technischen Anlage besitzt. Stimmt das aktuelle Verhalten nicht mit dem Modellverhalten überein, so wird darauf geschlossen, dass ein Fehler das Prozessverhalten verändert hat (Abb. 1.16). Wenn man diese Vorgehensweise mit einem Modell wiederholt, das das Prozessverhalten unter der Wirkung eines Fehlers  $f$  beschreibt, so kann man erkennen, ob möglicherweise der Fehler  $f$  eingetreten ist.

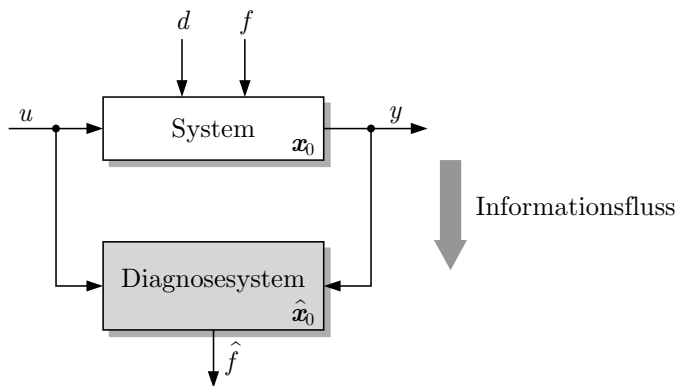


Abb. 1.16: Prozessdiagnose

Aus der Abbildung erkennt man, dass dieses Vorgehen zwei Schwierigkeiten aufweist. Erstens braucht man für die Diagnose den Anfangszustand  $x_0$ , der i. Allg. nicht messbar und daher unbekannt ist. Jede Diagnoseaufgabe enthält deshalb implizit eine Zustandsbeobachtungsaufgabe. Dieser Umstand schlägt sich in der Tatsache nieder, dass in den Kap. 10 und 16 die Fehlerdiagnoseaufgabe für kontinuierliche bzw. ereignisdiskrete Systeme mit Methoden der Zustandsbeobachtung gelöst wird.

Zweitens sieht man, dass die auf den Prozess wirkende Störung  $d$  die Ursache für die Abweichung des gemessenen vom berechneten Ausgang sein kann. Bei der Bewertung der festgestellten Abweichung im Hinblick auf die Existenz eines Fehlers muss man diese Störung in

Betracht ziehen und deshalb möglicherweise eine bestimmte Toleranz akzeptieren, bevor man einen Fehler signalisiert.

### 1.4.6 Regelung und Steuerung

Da bei den bisher betrachteten Automatisierungsaufgaben das Verhalten einer Anlage vorhergesagt bzw. überwacht werden sollte, war der Informationsfluss vom Prozess zur Automatisierungseinrichtung gerichtet. Bei den in diesem Abschnitt behandelten Steuerungen ist dies anders: Der Informationsfluss führt von der Automatisierungseinrichtung zum technischen Prozess bzw. verkoppelt die Automatisierungseinrichtung mit dem Prozess zu einem Kreis. Den Erläuterungen der dabei verwendeten Strukturen muss eine allgemeine Bemerkung zu den Begriffen Regelung und Steuerung vorangestellt werden.

**„Regeln“ und „Steuern“ in der deutschen Fachsprache.** Wie bereits hervorgehoben wurde, wird der Begriff Steuern in diesem Buch in der sehr allgemeinen Bedeutung verwendet, ein dynamisches System zielgerichtet zu beeinflussen. Die Stellgröße kann dabei sowohl reellwertig als auch wertdiskret sein. Dies sei hier noch einmal betont, weil traditionell nach anderen Kriterien zwischen Regeln und Steuern unterschieden wurde.

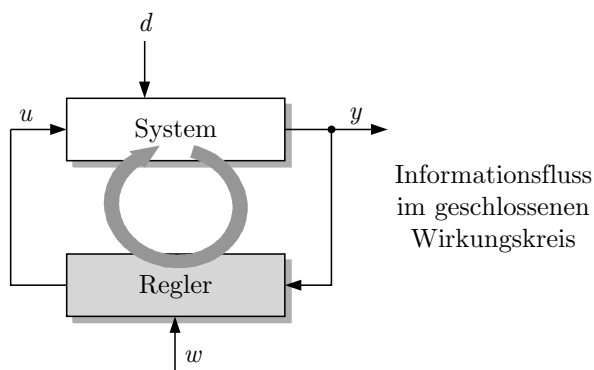
Bei kontinuierlichen Systemen setzt man beide Begriffe in Abhängigkeit von der Struktur ein, in der der Prozess und die Automatisierungseinrichtung verkoppelt ist, und spricht von einer Regelung, wenn die Steuerung im geschlossenen Wirkungskreis erfolgt, und von einer Steuerung (oder Vorsteuerung), wenn es sich um eine offene Wirkungskette handelt. Bei diskreten Systemen verwendete man in der Vergangenheit ausschließlich den Begriff der Steuerung, weil man von einem geschlossenen Wirkungskreis forderte, dass die Informationen kontinuierlich rückgekoppelt werden und dabei „die zu regelnde Größe ... fortlaufend erfaßt, mit ... der Führungsgröße verglichen und im Sinne einer Angleichung beeinflusst wird“ (DIN 19226 in [23]). Weil die kontinuierliche Arbeitsweise fehlt, betrachtete man diskrete Steuerungen als Steuerungen in der offenen Kette. Entsprechend dieser Definition sagt man auch, dass die Automatisierungstechnik mit der Regelungstechnik und der Steuerungstechnik zwei bisher weitgehend getrennt voneinander entwickelte Bereiche umfasst, wobei der Begriff Regelung den kontinuierlichen und der Begriff Steuerung den diskreten Systemen zugeordnet wird.

Diese Unterteilung ist nicht mehr zeitgemäß, weil auch die meisten diskreten Steuerungen im geschlossenen Wirkungskreis arbeiten, wie Kap. 13 noch ausführlich erläutern wird. Der einzige Unterschied zwischen einem kontinuierlichen und einem diskreten Regelkreis besteht in der Art der verwendeten Signale, die im ersten Fall reell und im zweiten Fall wertdiskret sind. Auch wenn bei diskreten Systemen der Informationskreis nur zu den Zeitpunkten geschlossen wird, an denen ein neues Ereignis auftritt, so hat die Rückkopplungsstruktur bei diskreten Systemen dieselben Charakteristika wie Regelkreise mit kontinuierlich arbeitenden Komponenten und es ist zweckmäßig, aufgrund der strukturellen Ähnlichkeiten für beide Systemklassen dieselben Begriffe zu verwenden. Dementsprechend werden in der neueren Literatur für diskrete Steuerungen im geschlossenen Kreis auch die Begriffe ereignisbasierte Regelung oder diskrete Regelung verwendet.

Dieses Lehrbuch schließt sich diesem Trend an und verwendet „Steuerung“ als übergeordneten Begriff, der kontinuierlich und diskret wirkende Stelleingriffe in der offenen Wir-

kungskette und im geschlossenen Wirkungskreis gleichermaßen bezeichnet. Untergeordnet ist der Begriff *Regelung*, der sich auf Steuerungen im geschlossenen Wirkungskreis bezieht und zwar unabhängig davon, ob die Signale kontinuierlich oder diskret sind. Der geschlossene Wirkungskreis wird in beiden Fällen als Regelkreis und die im geschlossenen Kreis arbeitende Steuereinrichtung als Regler bezeichnet. Wenn betont werden soll, dass die betrachtete Steuerung mit diskreten Stellgrößen arbeitet, wird von einer *diskreten Steuerung* oder einer diskreten Regelung gesprochen.

Diese Begriffsbestimmung entspricht übrigens auch den Gepflogenheiten in der englischsprachigen Literatur, in der man nur den Begriff *control* für beide Steuerungsarten kennt. Eine Steuerung im geschlossenen Wirkungskreis heißt dort *feedback control* und eine in der offenen Wirkungskette *feedforward control*.

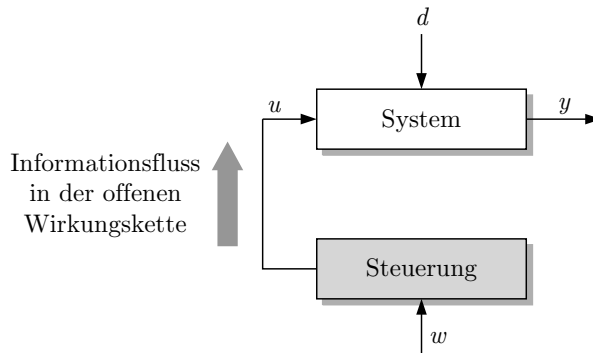


**Abb. 1.17:** Steuerung im geschlossenen Wirkungskreis (Regelung)

**Steuerung im geschlossenen Wirkungskreis (Regelung).** Bei der in Abb. 1.17 dargestellten Regelung bildet die Automatisierungseinrichtung mit dem zu steuernden System einen geschlossenen Kreis. Der zum Zeitpunkt  $t$  auf den Prozess einwirkende Stelleingriff  $u(t)$  wird in Abhängigkeit von der zur selben Zeit gemessenen Ausgangsgröße  $y(t)$  gewählt. Die Art und Weise, nach der die Stellgröße  $u(t)$  aus der Messgröße  $y(t)$  und der Sollgröße  $w(t)$  bestimmt wird, wird *Reglergesetz* genannt. Verfahren zur Festlegung des Reglergesetzes werden für kontinuierliche Systeme in den Kap. 7 und 8 und für diskrete Systeme in den Kap. 13 und 14 behandelt.

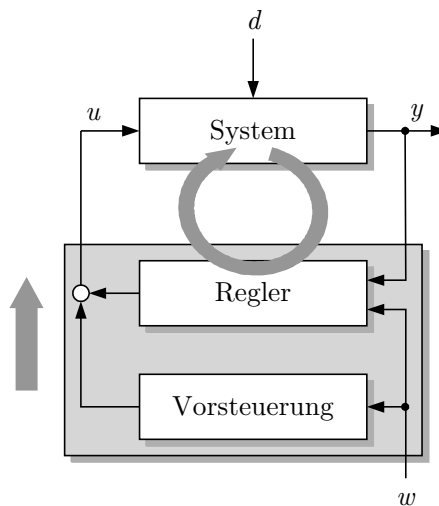
Die Rückführstruktur hat wesentliche Konsequenzen in Bezug auf die Wirkung, die die Automatisierungseinrichtung auf den Prozess haben kann (Abschn. 2.5). Insbesondere kann ein Regler den Einfluss einer Störung  $d$  aus der gemessenen Ausgangsgröße  $y$  erkennen und auf diese Störung reagieren, was in der offenen Kette nicht möglich ist.

**Steuerung in der offenen Wirkungskette.** Die in der Abb. 1.18 dargestellte Steuerung muss genauer als eine Steuerung in der offenen Wirkungskette bezeichnet werden. Bei ihr ergibt sich kein geschlossener Wirkungskreis, so dass sie nicht auf Störungen reagieren kann. Die Steuer-



**Abb. 1.18:** Steuerung in der offenen Wirkungskette

einrichtung kann lediglich aus der gegebenen Führungsgröße  $w$  mit Hilfe eines Prozessmodells vorausberechnen, welchen Verlauf  $u(t)$  die Eingangsgröße haben muss, damit die Regelgröße  $y(t)$  den vorgegebenen Verlauf  $w(t)$  bekommt. Die Art und Weise, wie  $u(t)$  aus  $w(t)$  ermittelt wird, wird *Steuerungsgesetz* genannt. Störungen machen sich stets in Abweichungen der Ausgangsgröße  $y$  von der Führungsgröße  $w$  bemerkbar.



**Abb. 1.19:** Kombination beider Steuerungsarten

**Kombination von Steuerungen in der offenen Wirkungskette und im Regelkreis.** Die beiden Steuerungsarten treten häufig in Kombination auf (Abb. 1.19), wobei die Steuerung in der offenen Kette eine schnelle Reaktion des Systems auf Änderungen des Steuerungsziels  $w$



erreichen soll, während der Regler Störungen  $d$  und Modellunbestimmtheiten kompensieren kann. Bei kontinuierlichen Systemen wird dabei der in der offenen Wirkungskette arbeitende Anteil der Steuerung als Vorsteuerung bezeichnet. Bei diskreten Systemen beinhaltet die in der offenen Kette wirkende Steuerungskomponente insbesondere Zeitplansteuerungen, bei denen die Steuereinrichtung den nächsten Stelleingriff nicht in Abhängigkeit vom Prozessausgang  $y$ , sondern nach dem Ablauf einer bestimmten Zeitspanne erzeugt (siehe Beispiel 2.11 auf S. 75). Abschnitt 2.5 enthält einen detaillierten Vergleich beider Steuerungsarten für beide Systemklassen.

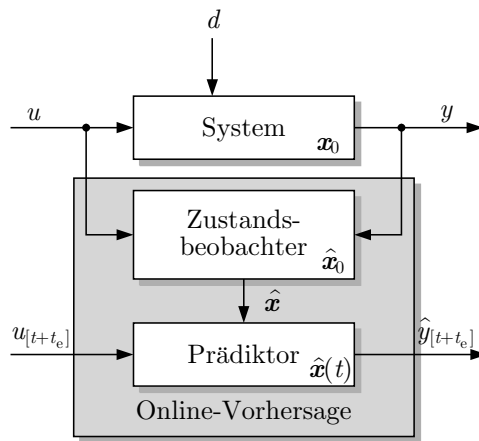


Abb. 1.20: Online-Vorhersage des Systemverhaltens

### 1.4.7 Kombination von Automatisierungsfunktionen

Die bisher betrachteten Automatisierungsfunktionen können kombiniert werden, worauf hier anhand von zwei Beispielen hingewiesen wird. Abbildung 1.20 zeigt, wie die Vorhersage des zukünftigen Verhaltens eines Prozesses verbessert werden kann, wenn der in Abb. 1.13 gezeigte Prädiktor mit dem in Abb. 1.15 dargestellten Zustandsbeobachter kombiniert wird.

Bei dieser Kombination liefert die Zustandsbeobachtung den aktuellen Prozesszustand  $\hat{x}(t)$ , in dem die Prädiktion beginnt. Dadurch werden Prädiktionsfehler vermieden, die durch einen falsch angenommenen Anfangszustand  $\hat{x}_0 \neq x_0$  entstehen. Außerdem reagiert die Prädiktion auf Störungen, weil sich diese in der Ausgangsgröße bemerkbar machen und damit über die Zustandsbeobachtung in die Vorhersage einbezogen werden.

Das zweite Beispiel ist die Kombination von Zustandsbeobachtung und Regelung. Wie in Abb. 1.21 zu sehen ist, kennt der Regler jetzt nicht nur die aktuelle Ausgangsgröße  $y(t)$ , sondern den gesamten Zustandsvektor  $x(t)$ . Er hat damit mehr Informationen zur Verfügung, was sich in einer verbesserten Regelgüte äußert.

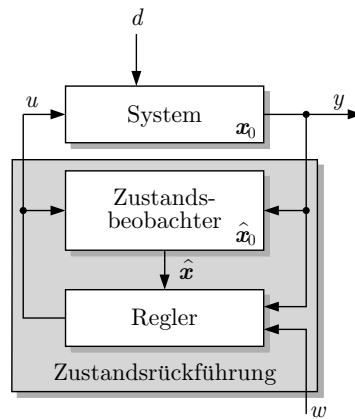


Abb. 1.21: Zustandsrückführung

### 1.4.8 Automatisierungshierarchie

Die genannten Automatisierungsfunktionen kommen auf unterschiedlichen Hierarchieebenen vor. Wie die Beispiele im Abschn. 1.2 gezeigt haben, übernimmt ein Automatisierungssystem Aufgaben, die sich bezüglich des Zeithorizonts der zu treffenden Entscheidungen, der verwendeten Informationen sowie der Zielstellungen unterscheiden. Für ein Elektroenergieversorgungsnetz wurden im Beispiel 1.6 mit der Lastvorhersage, der Einsatzplanung für die Kraftwerke und der Regelung der Kraftwerksblöcke drei Ebenen der Automatisierung beschrieben. Eine ähnliche hierarchische Gliederung wurde auch bei der Flugüberwachung und -regelung behandelt (Abb. 1.7 auf S. 16).

Abbildung 1.22 zeigt eine geläufige Gliederung der Automatisierungsaufgaben in einer *Automatisierungspyramide*. Automatisch gelöste Aufgaben findet man vor allem auf den unteren beiden Ebenen, der Feldebene und der Prozessleitebene. Zu den wichtigsten Aufgaben dieser Ebenen gehören die folgenden:

- **Feldebene:** Messen und Datenerfassung, Anzeigen (vor Ort), Stellen, Realisierung von Bedienhandlungen vor Ort.
- **Prozessleitebene:** Realisierung von Regelungen und Steuerungen, Datenaufzeichnung und Protokollierung wichtiger Signale, Datenaufbereitung und -archivierung, Optimierung des Prozessablaufs, Realisierung von Rezeptursteuerungen bei Chargenprozessen, Mensch-Maschine-Kommunikation.

Mit der Weiterentwicklung der Automatisierungsgeräte werden Regelungs- und Steuerungsfunktionen zunehmend von der Prozessleitebene in die Feldebene verlagert. Der Datenaustausch zwischen beiden Ebenen betrifft vor allem die Übertragung von Mess- und Stellinformationen.

Darüber sind die Leitungsebenen eines Unternehmens angeordnet, die sich mit der Führung von Betrieben bzw. des gesamten Unternehmens befassen. Zu den wichtigsten Aufgaben dieser Ebene gehören die folgenden:

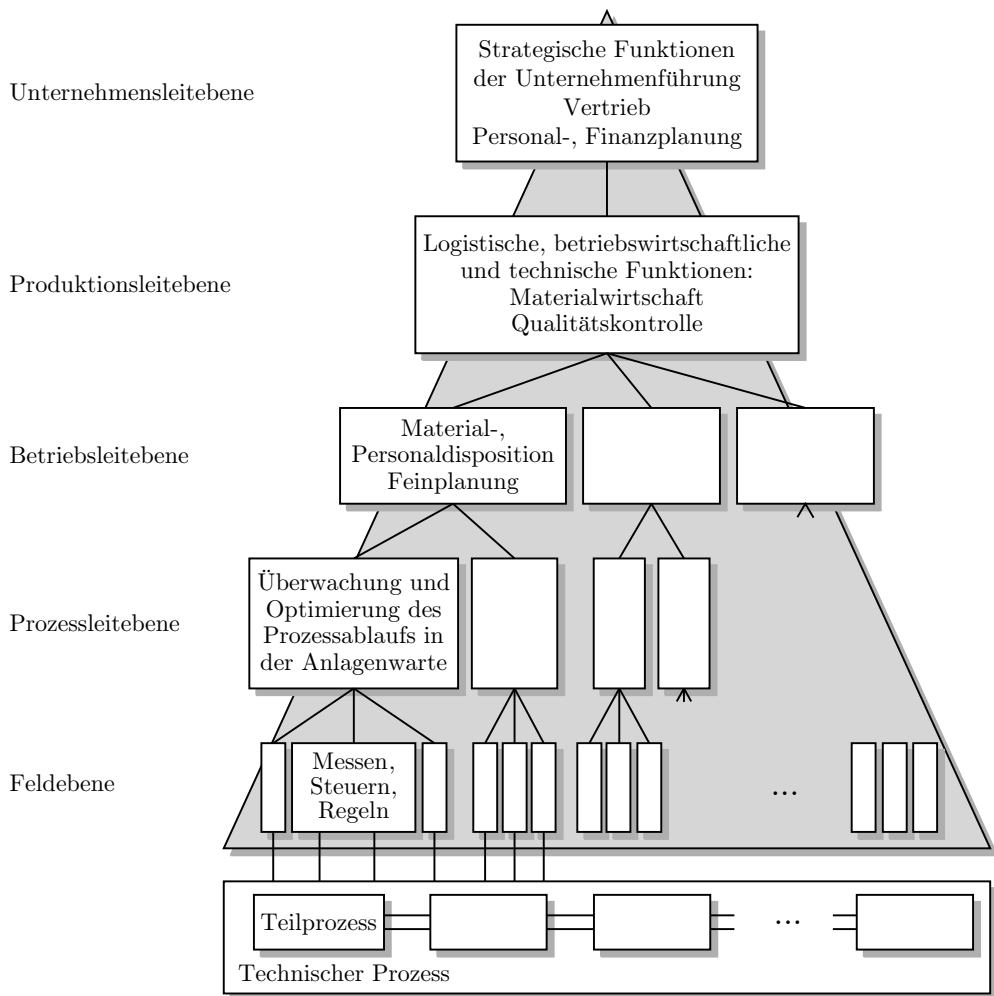


Abb. 1.22: Hierarchische Gliederung eines Automatisierungssystems

- **Betriebsleitebene:** Kurzfristige Produktionsplanung einschließlich der Einsatzplanung von Maschinen und Anlagen, Bilanzierungsrechnungen, Qualitätskontrolle, Langzeitarchivierung von Daten, Rezepturverwaltung, Planung des Personaleinsatzes.
- **Produktionsleitebene:** Verwaltung und Bearbeitung von Lieferaufträgen, Produktionsplanung und Terminüberwachung, Kostenanalyse.
- **Unternehmensleitebene:** Unternehmensführung, Marktanalyse, strategische Produktionsplanung, Investitionsplanung, Personalplanung und Personaleinsatz.

Diese Managementaufgaben beziehen sich nicht nur auf technische Daten wie die beiden darunter liegenden Ebenen, sondern auch auf eine Vielzahl organisatorischer Daten. Sie werden

heute durch umfangreiche Informationssysteme unterstützt, aber die meisten dieser Aufgaben sind nicht vollständig automatisiert und werden unter wesentlicher Mitwirkung des Personals der entsprechenden Ebene gelöst. Die Automatisierung dieser *Geschäftsprozesse* ist ein Gebiet, das auf Grund der Vielfalt und Menge der zu verarbeitenden Daten ein eigenes Gebiet der Informatik geworden ist. Automatisierungstechnische Methoden werden beispielsweise bei der Einsatzplanung von Maschinen angewendet.

In der gezeigten hierarchischen Gliederung findet man auf den unteren Ebenen typischerweise Aufgaben, die eine kontinuierliche Betrachtung des zu automatisierenden Prozesses notwendig machen, während auf übergeordneten Ebenen diskrete Erscheinungen und Ereignisse das Verhalten des betrachteten Systems und die zu treffenden Entscheidungen charakterisieren. Die Mischung aus beiden Verhaltensformen ist ein wichtiger Grund dafür, dass dieses Buch die Automatisierung beider Systemklassen behandelt. Dabei wird sich zeigen, dass die grundlegenden Herangehensweisen für beide Systemklassen sehr ähnlich sind, während sich die mathematische Behandlung unterscheidet.

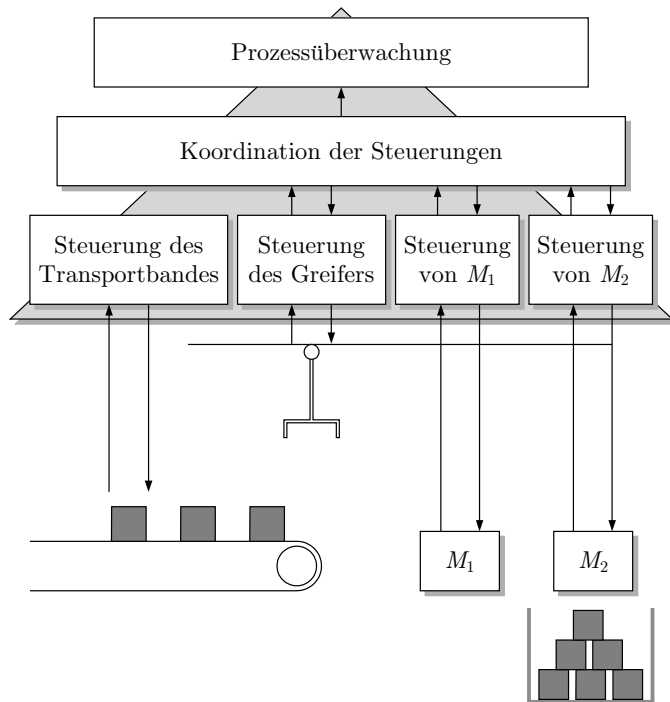
#### **Beispiel 1.10** Hierarchische Steuerung einer Fertigungszelle

Bei der in Abb. 1.23 dargestellten Fertigungszelle werden Werkstücke nacheinander durch die Maschinen  $M_1$  und  $M_2$  bearbeitet. Ein Greifer befördert ein Werkstück vom Band zur Werkzeugmaschine  $M_1$  und nach der dortigen Bearbeitung zur Werkzeugmaschine  $M_2$ . Die Maschine  $M_2$  kann das fertige Werkstück ohne Hilfe des Greifers selbst in die Ablage schieben.

Die Steuerung ist hierarchisch gegliedert. Die Komponenten „Transportband“, „Werkzeugmaschine  $M_1$ “, „Werkzeugmaschine  $M_2$ “ und „Greifer“ besitzen Steuereinrichtungen, die z. B. bewirken, dass das Transportband nach der Entnahme eines Werkstücks das nächste Werkstück in den Arbeitsbereich des Greifers fährt, dass der Greifer die Werkstücke vom Transportband bzw. der linken Werkzeugmaschine abholt und zu einer Maschine bringt und dass die beiden Werkzeugmaschinen die vorgesehenen Bearbeitungsschritte ausführen. Da diese Vorgänge an jeder Komponente weitgehend unabhängig von den Vorgängen an anderen Komponenten ablaufen, werden sie durch lokale Steuereinrichtungen realisiert. Allerdings ist zu ausgewählten Zeitpunkten eine Koordination der Steuerungen notwendig, um beispielsweise nach der Bearbeitung eines Werkstücks durch die Maschine  $M_1$  den Greifer zu „rufen“ und den Weitertransport des Werkstücks zur Maschine  $M_2$  zu veranlassen. Diese Koordination erfolgt durch die Komponente „Koordination der Steuerungen“ auf der nächsthöheren Automatisierungsebene. Ob diese Komponente bereits der zentral installierten Prozessleitebene oder noch der direkt an den Maschinen realisierten Feldebene zuzuordnen ist, hängt von der Implementierung ab.

In der Abbildung ist zu erkennen, dass sich die Koordination nicht auf das Transportband erstreckt. Wenn man annehmen kann, dass stets genügend viele Werkstücke zur Bearbeitung bereit liegen und das Transportband das nächste Werkstück weit genug vorfährt, ist keine Koordination mit dem Greifer und den Werkzeugmaschinen notwendig. Generell ist man daran interessiert, eine Koordination nur dort vorzusehen, wo sie notwendig ist, und möglichst viele Komponenten der Automatisierungseinrichtung dezentral arbeiten zu lassen.

Die in der Automatisierungshierarchie darüber angeordnete Komponente der Prozessüberwachung gehört zur Prozessleitebene und ist typischerweise auf einem zentralen Rechner realisiert. Sie erhält ausgewählte Informationen von den dezentralen Steuereinrichtungen und überprüft die Fehlerfreiheit der Arbeit aller Komponenten. Bei der Fehlerdiagnose tritt nur ein Informationsfluss von den dezentralen Steuerungen zur Überwachungskomponente auf. □



**Abb. 1.23:** Schematische Darstellung der Fertigungszelle

**Aufgabe 1.5\*** *Autofahren als Problem der Mehrebenensteuerung*

Beim Autofahren werden Steuerungsaufgaben auf unterschiedlichen Abstraktionsebenen gelöst, beginnend auf der unteren Ebene bei einfachen Temperaturregelkreisen im Motor bis zur Festlegung der Fahrtroute auf der obersten Ebene.

1. Ordnen Sie die Ihnen bekannten Regelungs- und Steuerungsaufgaben in einer hierarchischen Struktur an. Beachten Sie, dass Steuerungsaufgaben der höheren Ebene Zielvorgaben für die Steuerungsaufgaben der unteren Ebene machen.
2. Welche Steuerungsaufgaben werden heute automatisch, welche durch den Fahrer gelöst? Begründen Sie, warum die von Ihnen als automatisch gekennzeichneten Steuerungen automatisiert wurden.
3. Beim Einsatz von Krankenwagen und der Feuerwehr wird die Fahrtroute für mehrere Fahrzeuge durch eine Leitzentrale vorgegeben. Wo findet sich diese Steuerungsaufgabe in Ihrer hierarchischen Gliederung wieder? ☐

## 1.5 Realisierung von Automatisierungseinrichtungen

### 1.5.1 Methoden und Geräte

Bei der Lösung von Automatisierungsaufgaben bedient man sich einerseits der Methoden der Automatisierungstechnik, um die Überwachungs- und Steuerungsalgorithmen zu entwickeln und zu testen, und andererseits der Automatisierungsgeräte, um die Algorithmen zu implementieren und mit dem technischen System zu verknüpfen. Die Automatisierungstechnik beruht deshalb auf zwei Säulen:

- **Methoden der Automatisierungstechnik:** Es gibt eine gut entwickelte Theorie, mit der dynamische Systeme beschrieben, analysiert, gesteuert und überwacht werden können. Obwohl sich diese Methoden nicht nur auf die Steuerung beziehen, sondern die ganze Palette der im Abschn. 1.4 genannten Automatisierungsaufgaben betreffen, fasst man sie zur *Theorie der automatischen Steuerung oder Steuerungstheorie* und – wenn es sich um kontinuierliche Systeme handelt – zur *Regelungstheorie* zusammen.
- **Automatisierungsgeräte:** Für die Realisierung von Automatisierungssystemen steht eine umfassende Gerätetechnik zur Verfügung, die auf die physikalischen Gegebenheiten der einzelnen Anwendungsgebiete zugeschnitten ist. Dazu gehören programmierbare Geräte wie beispielsweise speicherprogrammierbare Steuerungen (SPS), Prozessleitsysteme oder Industrie-PCs. Auf diesen Geräten werden die Automatisierungsfunktionen durch Anwenderprogramme realisiert, die meist mitgeliefert werden und nur entsprechend der konkreten Situation zu konfigurieren sind. Außerdem gehören Sensoren und Aktoren zu den Automatisierungsgeräten. Sie stellen die Schnittstelle zwischen dem Automatisierungssystem und dem zu steuernden System her. Automatisierungsgeräte werden heute mit rechnergestützten Werkzeugen für die Projektierung, Inbetriebnahme und Instandhaltung geliefert (CAD – *computer-aided design*; CAE – *computer-aided engineering*).

Beide Gebiete hängen eng zusammen, denn man wird eine Methode zur Überwachung oder zur Steuerung eines Systems nur dann auch praktisch einsetzen, wenn die für die Realisierung dieser Methode notwendigen Automatisierungsgeräte zur Verfügung stehen.

Die folgenden Kapitel konzentrieren sich auf die Methoden der Automatisierungstechnik und gehen nur in dem für das Verständnis der Anwendungsbeispiele notwendigen Maße auf gerätetechnische Aspekte ein. Sie betrachten also die Automatisierungstechnik als eine Methodenwissenschaft, deren wichtigstes Ziel es ist, systematische Ansätze für die Lösung der genannten Automatisierungsaufgaben in unterschiedlichen Anwendungsgebieten und weitgehend unabhängig von der verwendeten Gerätetechnik zu entwickeln und für die Anwendung bereitzustellen.

In dieser die Methoden in den Mittelpunkt stellenden Betrachtungsweise wird der Begriff Automatisierungstechnik synonym mit den Begriffen Systemtechnik und Technische Kybernetik gebraucht. Im Unterschied dazu verwendet man Begriffe wie Industrielle Informationstechnik oder Steuerungstechnik mit stärkerem Bezug zur gerätetechnischen Realisierung von (häufig diskreten) Steuerungen.

### 1.5.2 Lösungsweg für Automatisierungsaufgaben

Um eine Automatisierungsaufgabe zu lösen, geht man i. Allg. in folgenden fünf Schritten vor:

---

**Algorithmus 1.1** *Lösung von Automatisierungsaufgaben*

---

1. **Prozessanalyse:** Stellen Sie die für die Automatisierung wichtigen Informationen über das Verhalten des zu automatisierenden Systems durch ein Modell dar.
  2. **Analyse des Systemverhaltens:** Ermitteln Sie die für die Automatisierung wichtigen Eigenschaften des zu automatisierenden Systems mit Hilfe des Modells.
  3. **Entwurf:** Entwickeln Sie für die zu lösende Automatisierungsaufgabe Lösungsalgorithmen. Das Ergebnis ist beispielsweise ein Regler, ein Zustandsschätzer, ein Diagnosealgorithmus oder eine Kombination derartiger Elemente. Diese Algorithmen beschreiben die Funktion der gesuchten Automatisierungseinrichtung.
  4. **Erprobung am Modell:** Erproben Sie Ihre Algorithmen am Modell des zu automatisierenden Systems. Dies geschieht i. Allg. durch Simulationsuntersuchungen, bei denen getestet wird, ob sich das Gesamtsystem in entscheidenden Situationen wie vorgeschrieben verhält.
  5. **Implementierung und Inbetriebnahme:** Realisieren Sie die Automatisierungseinrichtung gerätetechnisch, schließen Sie sie an den gegebenen Prozess an und nehmen Sie sie in Betrieb.
- 

Im ersten Schritt muss ein Modell des zu steuernden Systems auf einem für die Automatisierung geeigneten Abstraktionsniveau und mit einer geeigneten Genauigkeit aufgestellt werden. Mit diesem Modell wird die Verbindung zwischen der technischen Funktion des betrachteten Systems und den Methoden der Automatisierungstechnik hergestellt. Die Schritte 2 bis 4 nutzen Methoden, die in den folgenden Kapiteln behandelt werden. Wenn sich im vierten Schritt herausstellt, dass die gefundene Lösung nicht vollständig oder nicht zufriedenstellend ist, muss im dritten Schritt ein neuer Entwurf durchgeführt werden. Dieses iterative Vorgehen ist keine Seltenheit, sondern charakteristisch für automatisierungstechnische Entwurfsaufgaben. Im fünften Schritt werden die entwickelten Algorithmen mit Hilfe von Automatisierungsgeräten realisiert.

Diesen Schritten schließen sich die Inbetriebnahme und während des Betriebs gegebenenfalls die Wartung und Verbesserung der Überwachungs- und Steuerungskomponenten an, wobei sich der letzte Schritt sowohl auf die verwendeten Methoden als auch auf die eingesetzte Hardware und Software beziehen kann.

### 1.5.3 Beziehungen zwischen der Automatisierungstechnik und angrenzenden Fachdisziplinen

**Interdisziplinärer Charakter der Automatisierungstechnik.** Für die Lösung einer Automatisierungsaufgabe reicht es nicht aus, die Methoden und die Geräte der Automatisierungstechnik zu kennen. Jede Automatisierungsaufgabe beschäftigt sich mit einem konkreten technischen Prozess, dessen wichtigste Eigenschaften herausgearbeitet und bei der Lösung der Automatisierungsaufgabe ausgenutzt werden müssen. Dafür ist eine interdisziplinäre Zusammenarbeit mit den Fachleuten, die die Wirkungsweise des zu automatisierenden Objekts und die Ziele der Automatisierung kennen notwendig, also mit Anlagentechnikern bei der Automatisierung von Fertigungs- und Materialflussprozessen, mit Verfahrenstechnikern für die Lösung von Aufgaben der Prozessautomatisierung oder mit Konstrukteuren für die Fahrzeugautomatisierung. Eine solche Zusammenarbeit ist für den Erfolg jeder Automatisierungsaufgabe von großer Bedeutung.

Das erste Ziel dieser Zusammenarbeit besteht in der systemtheoretischen Darstellung der Automatisierungsaufgabe. Dabei liegt der schwierigste Schritt in der Abstraktion von den physikalischen Gegebenheiten, so dass im Wesentlichen nur noch mit den für die betrachtete Aufgabenstellung wichtigen Informationen gerechnet wird, also insbesondere Material- und Energieflüsse durch Informationsflüsse ersetzt sind. Auf diesen Schritt wird ausführlicher in den Kap. 2, 3 und 11 eingegangen. Wenn dieser Schritt vollzogen und ein dynamisches Modell des Automatisierungsobjekts aufgestellt ist, kann man mit den Methoden der Automatisierungstechnik, die in diesem Lehrbuch behandelt werden, die betreffende Aufgabe lösen.

**Automatisierungstechnik und Informatik.** Es gibt zwei Gebiete, mit denen sich sowohl die Automatisierungstechnik als auch die Informatik beschäftigen: Methoden für die Modellierung und Analyse diskreter Systeme sowie Hardware- und Softwareprobleme der Echtzeitdatenverarbeitung.

Das erste Gebiet betrifft die im dritten Teil des Buches behandelten ereignisdiskreten Systeme. Wichtige Modellformen wie Automaten oder Petrinetze werden in vielen Ingenieurwissenschaften und in der Informatik gleichermaßen eingesetzt, wobei vor allem Analysemethoden aus der Informatik in die Automatisierungstechnik übernommen wurden. Ein Schwerpunkt ist dabei die Beherrschung bzw. Vermeidung der kombinatorischen Komplexität von Modellierungs- und Analyseaufgaben. In Erweiterung dessen wurden in der Automatisierungstechnik Methoden zum Steuerungsentwurf für diese Systemklasse entwickelt.

Die Zusammenarbeit beider Fachgebiete für die Entwicklung der Automatisierungsgeräte und die Realisierung von Automatisierungssystemen kann man folgendermaßen charakterisieren: Aufgabe der Automatisierungstechnik ist es festzulegen, *was* informationsverarbeitende Systeme zur Lösung einer Automatisierungsaufgabe zu tun haben, während die Informatik Antworten auf die Frage nach dem *Wie* der technischen Realisierung dieser Informationsverarbeitung geben soll. Zu den wichtigsten gerätetechnischen Problemen, bei denen die Automatisierungstechnik Methoden der Informatik einsetzt, gehören die folgenden:

- **Echtzeitprogrammierung:** Wie können Algorithmen strukturiert und implementiert werden, so dass sie unter den in der Automatisierungstechnik vorherrschenden Echtzeitbedingungen ablaufen?



- **Datenbanken:** Wie können die umfangreichen Mengen an Prozessdaten gespeichert, aufbereitet, strukturiert und ausgewertet werden?
- **Vernetzte Rechnersysteme:** Welche Kommunikationsformen zwischen den zu einem Automatisierungssystem gehörenden Rechnern sind zweckmäßig und notwendig?
- **Fernautomatisierung:** Wie können Automatisierungsfunktionen über Datennetze von weit entfernt gelegenen Automatisierungseinrichtungen aus realisiert werden?

Diese Aufzählung zeigt die sehr enge Verflechtung von Automatisierungstechnik und Informatik, auf Grund derer im Zusammenhang mit der Realisierung von Steuerungs- und Überwachungsfunktionen durch Rechner auch der Begriff der *Prozessinformatik* gebraucht wird.

**Automatisierungstechnik und Nachrichtentechnik.** Die Automatisierungstechnik beruht auf der Verarbeitung von Informationen. Insofern enthalten Automatisierungsaufgaben stets auch eine Reihe von Aufgaben der Informationsübertragung und der Signalverarbeitung.

Grundlage für die Zusammenarbeit von Automatisierungstechnik und Nachrichtentechnik ist die gleichartige systemtheoretische Betrachtungsweise von Signalen und Systemen. Viele grundlegende Methoden werden in beiden Fachgebieten in gleicher oder ähnlicher Weise eingesetzt.

Typische Aufgaben der Signalverarbeitung wie das Herausfiltern des Nutzsignals aus einer verrauschten Messwertfolge oder die Übertragung von Signalen sind mit allen in diesem Buch behandelten automatisierungstechnischen Fragestellungen verbunden, ohne dass darauf im Einzelnen eingegangen wird. Da die Nachrichtentechnik das Handwerkszeug für die Lösung dieser Aufgaben schafft, kann im Folgenden in idealisierender Weise angenommen werden, dass Messgrößen wenig gestört und dass Signale sicher und schnell übertragen und unter Echtzeitbedingungen fehlerfrei verarbeitet werden können.

## Literaturhinweise

Für Themen, die eng mit den hier behandelten Methoden der Automatisierungstechnik zusammenhängen, aber nicht im Einzelnen erläutert werden, wird auf die folgende Spezialliteratur verwiesen. Die *Prozessoptimierung*, die sich mit der Wahl zweckmäßiger Arbeitspunkte beschäftigt, basiert auf Methoden für die statische und dynamische Optimierung. Eine leicht verständlich Einführung gibt [27], eine breitere Übersicht [82].

Die *Prozessleittechnik* für verfahrenstechnische Anlagen wird in [33] in Übersichten behandelt. Die Methoden der *Prozessführung* sind in [95] zusammengefasst. Eine umfangreiche Darstellung der Prozessinformatik, also der mit dem Einsatz von Rechnern zur Steuerung technischer Systeme verbundenen Probleme und Lösungswege, findet sich in [39]. In [74] werden die Unterschiede und Gemeinsamkeiten von Leitsystemen in kontinuierlichen und diskreten Produktionsprozessen behandelt.

Einführungen in die experimentelle Prozessanalyse werden beispielsweise in [43, 107] gegeben.

Als Literatur für spezielle Automatisierungsaufgaben sei auf [49] für die Steuerung von Materialflusssystemen und auf [44] für Regelungsaufgaben in großen Kommunikationsnetzen verwiesen. In [52] werden die automatisierungstechnischen Aufgaben und Lösungen im Kraftfahrzeug ausführlich dargestellt. Dieses Buch enthält viele im Einzelnen nachvollziehbare Modelle und beschreibt Regelungs- und Steuerungsmethoden in ihrem praktischen Einsatz, z. B. die Steuerung eines Antiblockiersystems oder eine Motorsteuerung. Der Band [96] beschreibt die Aufgaben und Lösungswege der Prozessregelung mit

vielen verfahrenstechnischen Beispielen, wobei einfache, gut nachvollziehbare dynamische Modelle und einfache Regelungsstrukturen behandelt werden. Der Beitrag [38] weist auf neue Anwendungsgebiete der Automatisierungstechnik im Gesundheitsbereich, in der Mikrosystemtechnik und Nanotechnik sowie im Bereich der mobilen Systeme hin. Die Monografien [59, 106] zeigen, dass medizintechnische Geräte ohne Automatisierungstechnik undenkbar wären.

Das Thema „Industrie 4.0“ wird derzeit in vielen Publikationen abgehandelt. Eine regelungstechnische Sicht beschreibt der Aufsatz [63].

Automatisierungsgeräte (Sensoren, Aktoren sowie Geräte für die Realisierung unterschiedlicher Automatisierungsfunktionen) werden ausführlich in dem dreibändigen Werk [34] behandelt.

Der im Beispiel 1.8 beschriebene Bremsassistent ist [53] entnommen. Für weitere Beispiele mechatronischer Systeme wird auf [94] verwiesen.

Schon früh in der Entwicklung der Automatisierungstechnik wurde versucht, die beiden Säulen der Steuerung kontinuierlicher Systeme und ereignisdiskreter Systeme zusammenzuführen. Zwei Arbeiten, die sich mit den Gemeinsamkeiten der Regelungstheorie und der Automatentheorie befassen, sind [5, 6].

# 2

## Grundlegende Eigenschaften dynamischer Systeme

*Bei der Lösung von Automatisierungsaufgaben werden grundlegende Entscheidungen anhand der Struktur des zu steuernden Systems getroffen. In diesem Kapitel werden mit dem Blockschaltbild und dem Signalflussgraphen zwei strukturelle Beschreibungsformen eingeführt und für die Kopplungsanalyse eingesetzt. Anschließend werden die Eigenschaften von Steuerungen in der offenen Wirkungskette und von Steuerungen im geschlossenen Wirkungskreis miteinander verglichen.*

### 2.1 Grundbegriffe der Systemtheorie

#### 2.1.1 Signal, Prozess, System

Die Automatisierungstechnik beschäftigt sich mit dynamischen Vorgängen in technischen oder nichttechnischen Systemen. Zur Beschreibung dieser Vorgänge wurden im Kap. 1 die Begriffe Signal, Prozess und System gebraucht, allerdings ohne sie genau zu erläutern. Diese Definitionen werden hier nachgeholt.

Unter einem dynamischen System versteht man eine Einrichtung, die unter dem Einfluss von einem oder von mehreren Eingangssignalen steht und dessen zeitliches Verhalten in einem oder mehreren Ausgangssignalen sichtbar wird (Abb. 2.1). Um diesen abstrakten Begriff zu verstehen, muss zunächst erklärt werden, was Signale sind, die am Eingang und Ausgang des Systems und darüber hinaus im Inneren eines Systems auftreten.

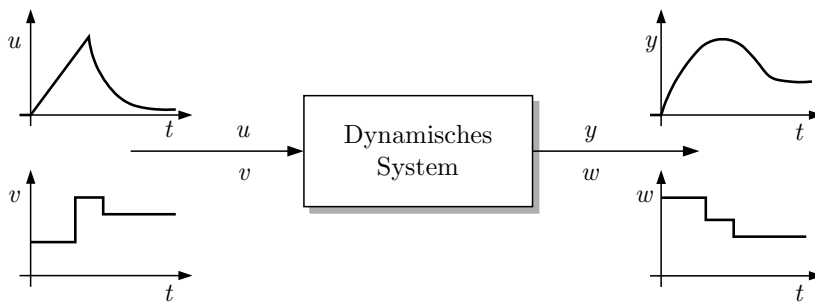


Abb. 2.1: Dynamisches System

Ein *Signal* beschreibt eine sich zeitlich ändernde Größe, durch die eine Information ausgedrückt wird.

Ein Signal  $s$  wird als eine Funktion der Zeit dargestellt

$$s : \mathcal{T} \rightarrow \mathcal{S},$$

wobei  $\mathcal{T}$  eine Menge von Zeitpunkten und  $\mathcal{S}$  die Menge der möglichen Signalwerte ist. Beispiele sind die Spannung an einem ohmschen Widerstand, der Füllstand in einem Behälter, die Temperatur eines Werkstücks oder die Konzentration eines Stoffes in einer Flüssigkeit. Diese Signale beschreiben den zeitlichen Verlauf physikalischer Größen. Signale können aber auch abstrakte Größen bezeichnen wie den Wert eines aus mehreren Messgrößen berechneten Qualitätsparameters (Durchsatz, Wirkungsgrad) in der Fertigung.

Unter einem *Prozess* versteht man einen Vorgang, bei dem sich der Wert einer oder mehrerer Kenngrößen zeitlich verändert.

Die sich zeitlich ändernden Größen werden durch Signale beschrieben.

Die in einer zu automatisierenden Anlage ablaufenden Prozesse sind durch die Umformung, die Speicherung oder den Transport von Energie, von Stoffen oder von Informationen gekennzeichnet. Der Begriff des Prozesses ist allerdings nicht an bestimmte physikalische Vorgänge gebunden. Beispielsweise sind das Füllen eines Behälters, das Erwärmen eines Werkstücks, das Bohren eines Loches und das Vermengen von Flüssigkeiten Prozesse mit unterschiedlichem physikalischen Hintergrund. Bei jedem dieser Vorgänge kann man eine oder mehrere Kenngrößen definieren, die sich zeitlich verändern. Diese Kenngrößen werden in Signalen ausgedrückt und zur Darstellung des Prozesses verwendet.

Unter einem *dynamischen System* versteht man einen Teil der Welt, in dem sich Prozesse abspielen.

Das Verhalten von Systemen ist durch ein oder mehrere Signale beschrieben, die durch das System in einen Zusammenhang zueinander gebracht werden.

Die gegebenen Definitionen zeigen, dass der Prozess den physikalischen Vorgang und das System die abstrakte technische Vorrichtung beschreibt, in der der Vorgang abläuft. Da man bei Automatisierungsaufgaben i. Allg. nicht zwischen der technischen Einrichtung und dem Prozessablauf unterscheiden muss, werden beide Begriffe im Folgenden synonym gebraucht.

In der Automatisierungstechnik spielen Systeme eine dominierende Rolle, deren Verhalten über Eingangsgrößen beeinflusst werden kann und sich in Ausgangsgrößen äußert. Entsprechend Abb. 2.1 werden die Eingangssignale mit  $u$  oder  $v$  und die Ausgangssignale mit  $y$  oder  $w$  bezeichnet, je nachdem, ob es sich um kontinuierliche oder diskrete Signale handelt.

Eingangssignale beschreiben Einflussgrößen der Umwelt auf das System. Sie sind die Ursache dafür, dass in dem System ein Prozess in Gang gesetzt wird, und sie nehmen Einfluss darauf, wie der Prozess abläuft. Ausgangssignale beschreiben die durch den Prozess hervorgerufenen und außerhalb des Systems beobachtbaren Veränderungen. Das System kann weitere innere Signale haben, die zur Beschreibung des Prozesses wichtig sind, aber nicht gemessen werden können. Derartige Signale werden später zum Zustandsvektor  $x(t)$  zusammengefasst.

Ein System kann man deshalb durch einen mathematischen Operator  $S$  darstellen, der die Eingangsgrößen in die Ausgangsgrößen transformiert. Ein System mit einer Eingangsgröße  $u$  und einer Ausgangsgröße  $y$  wird dann durch die Gleichung

$$y = S(u)$$

repräsentiert, bei der  $S$  als Systemfunktion, Systemoperator oder Eingangs-Ausgangsrelation bezeichnet wird. Diese Darstellung zeigt, dass man bei einer systemtheoretischen Betrachtung von der physikalischen Natur der Prozesse abstrahiert und das System anhand seiner durch den Operator  $S$  gegebenen mathematischen Beschreibung analysiert, überwacht und steuert. Der Systemoperator  $S$  wird später durch eine Differentialgleichung, ein Zustandsraummodell, einen Automaten oder ein Petrinetz repräsentiert. Wichtig ist jetzt zunächst nur, dass ein System eine technische Einheit darstellt, die eine Transformation der Eingangsgröße  $u$  in eine Ausgangsgröße  $y$  realisiert. Dabei sind  $u$  und  $y$  nicht einzelne Werte, sondern Zeitfunktionen. Der Systemoperator  $S$  bildet also die Menge aller möglichen Zeitfunktionen  $u(t)$  in die Menge aller möglichen Zeitfunktionen  $y(t)$  ab.

## 2.1.2 Kontinuierliche und diskrete Signale und Systeme

**Kontinuierliche und diskrete Signale.** Eine wichtige Einteilung von Signalen und Systemen bezieht sich auf die Wertebereiche  $\mathcal{T}$  und  $\mathcal{S}$  der Signale. Wertkontinuierliche Signale sind Zeitfunktionen, bei denen sowohl die Zeit  $t$  als auch die Signalwerte  $s$  reellwertig sind, die also als Funktionen der Form

$$s : \mathbb{R} \rightarrow \mathbb{R}$$

dargestellt werden können, wobei  $\mathbb{R}$  die Menge der reellen Zahlen bezeichnet. Viele kontinuierliche Signale können durch analytische Ausdrücke beschrieben werden, z. B. durch

$$s(t) = 5t \quad \text{oder} \quad s(t) = \sin 7t.$$

Sie repräsentieren Prozesse mit sich kontinuierlich verändernden Kenngrößen, beispielsweise einen Druck- oder einen Temperaturverlauf.

Alternativ dazu können die Zeit und die Signalwerte auf diskrete Mengen beschränkt sein. Wenn man die Zeitpunkte, für die man den Signalwert betrachtet, und die Elemente des Wertebereichs durchnummeriert, so kann man diskrete Signale als Funktionen der Form

$$s : \mathbb{N} \rightarrow \mathbb{N}$$

notieren, bei denen  $\mathbb{N}$  die Menge der natürlichen Zahlen bezeichnet. Für diskrete Signale gibt es häufig keine geschlossenen analytischen Beschreibungen, sondern man stellt den Verlauf des Signals  $s$  durch die Folge der Signalwerte dar. Die Gleichung

$$s = (1, 3, 2, 2, \dots)$$

besagt, dass das Signal  $s$  zum Zeitpunkt 0 den Wert 1 hat, zum Zeitpunkt 1 den Wert 3, zum Zeitpunkt 2 den Wert 2 usw.:

$$s(0) = 1, \quad s(1) = 3, \quad s(2) = 2, \dots$$

Beispielsweise beschreiben diskrete Signale den zeitlichen Verlauf der Anzahl der Pakete in einem Lager oder der Werkstücke in einem Transportbehälter, die Folge von Montageoperationen oder die Liste von nacheinander auftretenden Kommandos einer Steuerung. Bei diesen Beispielen nehmen die Signale ganzzahlige oder symbolische Werte an. Andererseits entstehen diskrete Signale, wenn man kontinuierliche Signale quantisiert, indem man beispielsweise einen reellwertigen Füllstand qualitativ als „voll“ oder „leer“ beschreibt oder den Systemzustand als „fehlerfrei“ oder „fehlerbehaftet“ klassifiziert. Viele Stellsignale sind diskret, z. B. wenn sie einen Brenner oder eine Beleuchtung an- oder ausschalten. Die Werte aller dieser Signale ändern sich nur zu diskreten Zeitpunkten, die man durchnummeriert.

In Abb. 2.1 sind das Eingangssignal  $u$  und das Ausgangssignal  $y$  kontinuierlich. Diese Signale können sich stetig ändern, aber auch zwischen verschiedenen Werten springen. Die Signale  $v$  und  $w$  sind diskrete Signale. Sie können nur diskrete Werte annehmen. Ihr Verlauf ist durch das Springen zwischen den diskreten Werten gekennzeichnet. Die zeitlichen Abstände der Sprünge müssen nicht gleich groß sein.

Kontinuierliche Eingangsgrößen werden später durchgängig mit  $u$  und diskrete mit  $v$  bezeichnet, während für die entsprechenden Ausgangsgrößen die Symbole  $y$  bzw.  $w$  verwendet werden. Soweit sich die folgenden Erläuterungen auf beide Signalklassen beziehen, werden für die Eingangsgröße das Symbol  $u$  und die Ausgangsgröße das Symbol  $y$  verwendet, ohne dass damit der kontinuierliche Charakter der Signale hervorgehoben werden soll.

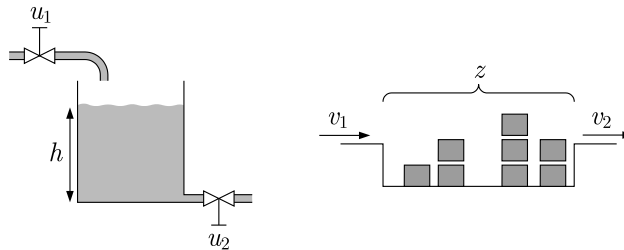
Neben diesen hier verwendeten Signalen kann es in der Automatisierungstechnik weitere Signale geben, beispielsweise solche, bei denen die Zeit diskrete Werte, aber das Signal reelle Werte annimmt. Diese Signale spielen im Folgenden jedoch keine Rolle.

**Kontinuierliche und diskrete Systeme.** Entsprechend der Klassifikation der auftretenden Signale ist ein wertkontinuierliches System ein dynamisches System, dessen Signale kontinuierlich sind und ein ereignisdiskretes System ein dynamisches System mit diskreten Signalen. Die Attribute „kontinuierlich“ und „diskret“ für das System entsprechen also den Attributen der im System auftretenden Signale und beziehen sich stets sowohl auf den Wertebereich als auch auf die Zeit. An Stelle von „wertkontinuierlich“ bzw. „ereignisdiskret“ wird im Folgenden kürzer von „kontinuierlich“ und „diskret“ gesprochen.

In der Automatisierungstechnik spielen beide Systemklassen eine große Rolle, so dass ihnen je etwa die Hälfte dieses Buches gewidmet ist. Dabei wird sich zeigen, dass die mathematischen Methoden zur Beschreibung und Behandlung beider Systemklassen sehr unterschiedlich sind, sich die zu lösenden Automatisierungsaufgaben und die grundlegenden Methoden jedoch ähneln.

### Beispiel 2.1 Batchreaktor und Paketlager

Als Beispiel zeigt Abb. 2.2 zwei Behälter, die gefüllt und entleert werden können. Der linke Behälter enthält eine Flüssigkeit der Füllhöhe  $h$ , die durch den über die Eingangsgrößen  $u_1$  und  $u_2$  beeinflussbaren Zu- bzw. Abfluss kontinuierlich erhöht bzw. verkleinert werden kann. Auf der rechten Seite ist ein Lager dargestellt, in dem Pakete gestapelt sind, deren Anzahl  $z$  durch die Anlieferung  $v_1$  und die Auslieferung  $v_2$  diskret beeinflusst wird. Während sich Zufluss, Abfluss und Füllhöhe kontinuierlich ändern, verändern sich die Anzahl der gelieferten, gestapelten und abgeholt Pakete diskret.



**Abb. 2.2:** Beispiele für einen kontinuierlichen und einen diskreten Prozess mit ähnlichen systemdynamischen Eigenschaften

Spätere Beispiele werden auch zeigen, dass die Entscheidung, ob ein System als kontinuierliches oder als diskretes System aufgefasst wird, nicht nur von den Eigenschaften des Systems abhängt, sondern in entscheidendem Maße auch vom Automatisierungsziel. Wenn man den Füllstand im linken Behälter in Abb. 2.2 mit Hilfe einer Regelung auf einem konstanten Wert halten will, so wird man das System als ein kontinuierliches behandeln. Wenn der Behälter jedoch Teil eines Batchprozesses ist, bei dem es nur darauf ankommt, ob er gerade bis zu einer vorgegebenen Füllhöhe gefüllt oder entleert ist, so kann man den Füllstand durch die beiden diskreten Werte 0 und 1 beschreiben und den Behälter wie ein diskretes System behandeln. In ähnlicher Weise kann man das im rechten Teil der Abbildung gezeigte Paketlager als diskretes System mit der ganzzahligen Anzahl von Paketen beschreiben oder im Falle eines großen Lagers, dessen Kapazitätsauslastung durch die mittlere Paketanzahl beschrieben wird, als kontinuierliches System auffassen.

Der kontinuierliche bzw. diskrete Charakter der Signale äußert sich auch in der verwendeten Zeitachse. Der Füllstand  $h$  ändert sich typischerweise zu jedem reellen Zeitpunkt  $t \in \mathbb{R}$ , so dass er mit  $h(t)$  bezeichnet wird. Bei dem Paketlager verändert sich der Lagerbestand  $z$  nur zu diskreten Zeitpunkten, die mit dem Zähler  $k$  durchnummeriert sind. Die Signale des diskreten Prozesses sind deshalb  $v_1(k)$ ,  $v_2(k)$  und  $z(k)$ .  $\square$

**Zustandsraummodell.** Die Analogie der beiden in Abb. 2.2 dargestellten Systeme geht noch weiter. Diese Systeme haben mit der Füllhöhe  $h$  bzw. der Anzahl  $z$  der gestapelten Pakete Kenngrößen, die den aktuellen Zustand der beiden Prozesse in direkter Analogie zueinander beschreiben. Dass diese Größen auch den Zustand der beiden Systeme im systemtheoretischen Sinne repräsentieren, wird sich in den Kap. 3 bzw. 11 zeigen. Hier ist zunächst wichtig zu erkennen, dass unabhängig von der kontinuierlichen bzw. diskreten Natur der Systeme Ähnlichkeiten in der Behandlung zu erwarten sind. Die später eingeführten Modelle beschreiben das Systemverhalten in Bezug zu diesen Systemzuständen, die bei kontinuierlichen Systemen durch einen Vektor  $\mathbf{x}(t)$  zu allen reellen Zeitpunkten  $t \in \mathbb{R}$  und bei diskreten Systemen durch einen skalaren Wert  $z(k)$  zu allen diskreten Zeitpunkten  $k \in \mathbb{N}$  erfasst werden. Der Einfluss einer kontinuierlichen Eingangsgröße  $u(t)$  auf den kontinuierlichen Zustand  $\mathbf{x}(t)$  wird durch die Differentialgleichung

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{g}(\mathbf{x}(t), u(t)), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.1)$$

beschrieben, deren Lösung für ein Eingangssignal  $u(t)$  und einen Anfangszustand  $\mathbf{x}_0$  den zeitlichen Verlauf  $\mathbf{x}(t)$  des Systemzustands liefert. Daraus kann die Ausgangsgröße  $y(t)$  mit Hilfe einer Funktion  $h$  berechnet werden:

$$y(t) = h(\mathbf{x}(t), u(t)). \quad (2.2)$$

Bei diskreten Systemen ändert sich der Zustand nur zu den diskreten Zeitpunkten  $k$ , so dass hier anstelle einer Differentialgleichung (2.1) die Gleichung

$$z(k+1) = G(z(k), v(k)), \quad z(0) = z_0 \quad (2.3)$$

steht, mit der für eine gegebene Folge der Eingangsgröße  $v(k)$ , ( $k = 0, 1, \dots$ ) und einen Anfangszustand  $z_0$  die Folge der durchlaufenen Zustände ermittelt werden kann. Daraus kann die diskrete Ausgangsgröße  $w(k)$  berechnet werden:

$$w(k) = H(z(k), w(k)). \quad (2.4)$$

Die Kap. 3 bzw. 11 zur Modellierung kontinuierlicher und diskreter Systeme werden zeigen, wie man den Zustand  $\mathbf{x}$  bzw.  $z$  für konkrete Anwendungen wählt und wie man die in den hier angegebenen Zustandsraummodellen vorkommenden Funktionen  $\mathbf{g}$ ,  $h$  und  $G$ ,  $H$  ermittelt.

### 2.1.3 Statische und dynamische Systeme

Die wichtige Unterteilung der Systeme in statische und dynamische kann ebenfalls ohne Bezug auf eine bestimmte Modellform eingeführt werden. Bei statischen Systemen wird der Wert der Ausgangsgröße  $y$  zum Zeitpunkt  $t$  nur durch den zum selben Zeitpunkt  $t$  auftretenden Wert der Eingangsgröße  $u$  bestimmt, so dass die Beziehung

$$y(t) = S(u(t))$$

gilt, in der  $S : \mathbb{R} \rightarrow \mathbb{R}$  bzw.  $S : \mathbb{N} \rightarrow \mathbb{N}$  eine Funktion ist, die den Wertebereich der Eingangsgröße in den Wertebereich der Ausgangsgröße abbildet. Ein ohmscher Widerstand ist ein



statisches System, weil der durch den Widerstand fließende Strom nur durch die zur selben Zeit am Widerstand abfallende Spannung bestimmt wird. Die Beschreibung des Zusammenhangs von  $u$  und  $y$  bzw. Strom und Spannung des Widerstands erfolgt deshalb durch eine algebraische Gleichung. Als Beispiel für ein statisches diskretes System kann ein Regensensor dienen, dessen binäres Ausgangssignal genau für diejenigen Zeitpunkte gleich eins ist, für die seine Oberfläche mit Wasser bedeckt ist.

Bei dynamischen Systemen hängt der Wert der Ausgangsgröße  $y$  zur Zeit  $t$  nicht nur von dem zum selben Zeitpunkt auftretenden Wert der Eingangsgröße  $u$  ab, sondern auch von deren bisherigem Verlauf. Im Extremfall muss man zur Berechnung von  $y(t)$  den Verlauf der Eingangsgröße für alle vergangenen Zeitpunkte kennen, also für das Intervall  $(-\infty, t]$ . Um zu kennzeichnen, dass nicht nur der aktuelle Wert, sondern der Verlauf der Eingangsgröße gemeint ist, wird das Symbol  $u_{(-\infty, t]}$  verwendet. Das System  $S$  bildet diesen Verlauf der Eingangsgröße in die Menge der Signalwerte der Ausgangsgröße ab:

$$y(t) = S(u_{(-\infty, t]}). \quad (2.5)$$

Der Systemoperator  $S$  wird beispielsweise durch eine Differentialgleichung beschrieben. Beim Zustandsraummodell (2.1), (2.2) wird die Wirkung der Eingangsgröße  $u_{(-\infty, 0]}$  im Anfangszustand  $x_0$  erfasst, so dass zur Berechnung von  $y(t)$  außer  $x_0$  nur die Eingangsgröße  $u_{[0, t]}$  im verkürzten Zeitintervall  $[0, t]$  bekannt sein muss und das System in der Form

$$y(t) = \tilde{S}(x_0, u_{[0, t]}) \quad (2.6)$$

dargestellt wird. Dabei bleibt die für dynamische Systeme charakteristische Eigenschaft erhalten, dass  $y(t)$  vom Verlauf der Eingangsgröße abhängt.

### Beispiel 2.2 Behandlung eines Behälters als dynamisches System

Der in Abb. 2.2 links gezeigte Behälter ist ein dynamisches System, denn man kann den aktuellen Füllstand nicht aus dem aktuellen Zufluss bestimmen. Für den Füllstand gilt die Differentialgleichung

$$\frac{dh(t)}{dt} = \frac{1}{A}(u_1(t) - u_2(t)),$$

in der  $A$  den Behälterquerschnitt,  $u_1(t)$  und  $u_2(t)$  die Massenströme durch die beiden Ventile zum Zeitpunkt  $t$  und  $h(t)$  den Füllstand beschreiben. Um den aktuellen Füllstand zu berechnen, integriert man diese Gleichung und erhält

$$h(t) = \frac{1}{A} \int_{-\infty}^t (u_1(\tau) - u_2(\tau)) d\tau. \quad (2.7)$$

Um  $h(t)$  berechnen zu können, muss man also den Verlauf des Zuflusses  $u_1$  und des Abflusses  $u_2$  im Zeitintervall  $(-\infty, t]$  kennen. Die angegebene Gleichung für den Füllstand hat die Form (2.5).

Beim Zustandsraummodell (2.1) speichert man die Informationen über den Verlauf der Eingangsgröße im Zeitintervall  $g \leq 0$  im Systemzustand  $x_0$ , so dass man sich nicht die gesamte Vergangenheit des Systems merken muss. Für den Behälter ist der Füllstand  $h$  zum Zeitpunkt  $t = 0$  der Anfangszustand  $x_0$  und man berechnet das Systemverhalten aus  $h(0)$  und dem Verlauf von  $u_1(\tau)$  und  $u_2(\tau)$  für das kürzere Zeitintervall  $0 \leq \tau \leq t$ :

$$h(t) = h(0) + \frac{1}{A} \int_0^t (u_1(\tau) - u_2(\tau)) d\tau. \quad (2.8)$$

Dies ist eine Darstellung des Systems der Form (2.6) mit  $x_0 = h(0)$ . Der Füllstand  $h(0)$  enthält also diejenigen Informationen über das Systemverhalten bis zur Zeit  $t = 0$ , die für die Berechnung des Verhaltens für  $t > 0$  notwendig sind. Kennt man diesen Zustand, so braucht man sich den Verlauf der Eingangsgröße für das Intervall  $(-\infty, 0]$  nicht mehr zu merken.

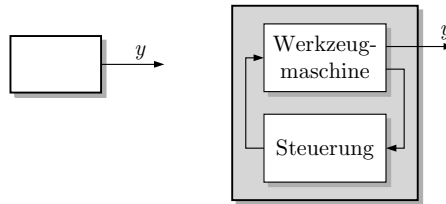
Auf ähnliche Gleichungen kommt man bei dem Paketlager, das in Abb. 2.2 rechts zu sehen ist. Dabei bezeichnen  $v_1(k)$  und  $v_2(k)$  die Anzahl der zum  $k$ -ten Zeitpunkt eintreffenden bzw. ausgelieferten Pakete. Für den Zustand  $z(k)$ , der die Anzahl der Pakete im Lager repräsentiert, erhält man die Beziehung

$$z(k) = z(0) + \sum_{j=0}^{k-1} (v_1(j) - v_2(j)),$$

die Gl. (2.8) sehr ähnlich ist. Auch hier muss man für die Bestimmung des aktuellen Zustands  $z(k)$  den Anfangszustand  $z(0)$  sowie die Folgen  $v_1(j)$  und  $v_2(j)$  der ankommenden und abgehenden Pakete für das diskrete Zeitintervall  $j \in [0, k - 1]$  kennen.  $\square$

### 2.1.4 Autonome und gesteuerte Systeme

Die meisten in der Automatisierungstechnik betrachteten Systeme werden wie das in Abb. 2.1 gezeigte System durch eine Eingangsgröße beeinflusst. Wenn man betonen will, dass das System eine oder mehrere derartige Eingangsgrößen besitzt, spricht man von einem *gesteuerten System*.



**Abb. 2.3:** Autonome Systeme

Andererseits heißt ein System *autonom*, wenn es keine Eingangsgrößen hat, sich also unabhängig von seiner Umgebung bewegt (Abb. 2.3). Die Bewegung erfolgt dann von einem Anfangszustand  $x_0$  aus. Entsprechend Gl. (2.6) gilt

$$y(t) = \tilde{S}(x_0, 0).$$

Das autonome Verhalten eines Systems kann man sich sehr gut an einem Pendel vorstellen, dem man eine Anfangsauslenkung gegeben hat und das sich auf Grund dieser Auslenkung ohne weitere äußere Beeinflussung bewegt.

Autonome Systeme ergeben sich auch dann, wenn man ein gesteuertes System mit seiner Steuereinrichtung verknüpft. Im rechten Teil der Abbildung ist eine gesteuerte Werkzeugmaschine zu sehen. Wenn man die Werkzeugmaschine einschaltet, führt sie beispielsweise zyklisch einen Stanzvorgang aus, wobei die Ausgangsgröße  $y$  anzeigt, in welchem Schritt dieses Zyklus sich die Maschine gerade befindet. Das System arbeitet autonom.

### 2.1.5 Gemeinsamkeiten und Unterschiede kontinuierlicher und diskreter Systeme

Die im Folgenden betrachteten Systeme haben wichtige gemeinsame Eigenschaften, die unabhängig von der kontinuierlichen bzw. diskreten Natur der Signale sind und die folgendermaßen zusammengefasst werden können:

- **Dynamik:** Es werden dynamische Systeme betrachtet, deren Ausgangsgrößen vom Verlauf der Eingangsgrößen (und nicht nur von deren aktuellen Wert) bestimmt werden.
- **Kausalität:** Die betrachteten Systeme sind kausal, d. h., der aktuelle Wert der Eingangsgrößen hat nur einen Einfluss auf die aktuellen und die zukünftigen Werte der Ausgangsgrößen, nicht jedoch auf die Werte der Ausgangsgrößen zu vergangenen Zeitpunkten.
- **Dimensionalität:** Die Systeme haben einen endlichdimensionalen Zustandsraum. Bei kontinuierlichen Systemen wird später mit dem  $n$ -dimensionalen Zustandsraum  $\mathbb{R}^n$ , bei diskreten Systemen mit einer endlichen Zustandsmenge  $\mathcal{Z}$  gerechnet.
- **Zeitinvarianz:** Die Systemeigenschaften ändern sich nicht. Dies äußert sich bei kontinuierlichen Systemen in konstanten Koeffizienten der Differentialgleichung und des Zustandsraummodells, bei diskreten Systemen in zeitlich unveränderlichen Zustandsübergangsrelationen.

Die Zeitinvarianz bezieht sich auf die Übertragungseigenschaften des Systems, nicht auf die Signale. Die Signalwerte ändern sich natürlich in Abhängigkeit von der Zeit. Was sich nicht ändert ist die Art und Weise, wie aus dem Eingangssignal das Ausgangssignal berechnet wird. Der Systemoperator  $S$  bleibt also derselbe. So verhält sich eine Maschine zeitinvariant, wenn der in ihr ablaufende Prozess unabhängig davon ist, ob die Maschine jetzt oder zu einem späteren Zeitpunkt eingeschaltet wird.

Auch wenn sich in der automatisierungstechnischen Praxis Systeme finden lassen, die eine oder mehrere dieser Eigenschaften nicht besitzen, so sind die hier getroffenen Annahmen für viele Systeme zutreffend und für das grundlegende Verständnis der automatisierungstechnischen Methoden sinnvoll.

Unterschiede bei der Behandlung kontinuierlicher und diskreter Systeme ergeben sich aus der unterschiedlichen Natur der Signale, die deshalb mit unterschiedlichen mathematischen Methoden behandelt werden müssen. Die wichtigsten Unterschiede lassen sich folgendermaßen zusammenfassen:

- **Unterschiedliche Wertebereiche für die Signale:** Während kontinuierliche Signale über die kontinuierliche Zeit  $t$  definiert werden und reelle Werte annehmen, können diskrete Signale nur diskrete Werte annehmen und sich deshalb nur zu bestimmten Zeitpunkten  $k$  sprungförmig verändern. Daraus resultieren vollkommen unterschiedliche Modellformen,

die für kontinuierliche Systeme auf Differentialgleichungen und für diskrete Systeme auf Zustandsübergangsrelationen beruhen.

- **Linearität:** Viele kontinuierliche Systeme können in guter Näherung durch lineare Modelle beschrieben werden, was die Modellbildung und die Lösung der Automatisierungsaufgaben wesentlich vereinfacht. Diese Eigenschaft besitzen diskrete Systeme i. Allg. nicht.
- **Nichtdeterminismus:** Diskrete Systeme können sich nichtdeterministisch verhalten, d. h., für eine gegebene Eingangsgröße kann nicht eindeutig vorhergesagt werden, welche zeitliche Entwicklung die Ausgangsgröße nimmt. Ein derartiges Verhalten ist bei kontinuierlichen Systemen nicht typisch und die Modelle kontinuierlicher Systeme lassen nichtdeterministisches Verhalten auch nicht zu.

## 2.2 Blockschaltbild und Signalflussgraph

### 2.2.1 Struktur dynamischer Systeme

Der Begriff Struktur wird mit Bezug zu dynamischen Systemen sehr unterschiedlich gebraucht. Bei den in diesem Abschnitt behandelten Blockschaltbildern und Signalflussgraphen beschreibt die Struktur, welche Teilsysteme bzw. Signale mit welchen anderen Teilsystemen oder Signalen verkoppelt sind. Bei dem für die Steuerbarkeitsanalyse im Kap. 5 eingesetzten Strukturgraphen wird die Struktur durch die von null verschiedenen Elemente der Systemmatrizen festgelegt. Bei der strukturellen Analyse von ereignisdiskreten Systemen im Kap. 12 reduziert man in ähnlicher Weise das Modell des betrachteten Systems auf einen gerichteten Graphen. Die verwendeten Definitionen haben gemeinsam, dass Eigenschaften als strukturell bezeichnet werden, wenn sie maßgebend vom Aufbau des Systems bestimmt sind. Diese Eigenschaften sind deshalb für eine ganze Klasse von Systemen typisch und von den konkreten Parameterwerten weitgehend unabhängig.

Dass zunächst der strukturelle Aufbau eines dynamischen Systems betrachtet wird, hat mehrere Gründe:

- Die Genauigkeit mathematischer Modelle wird wesentlich dadurch beeinflusst, dass die Modelle das betrachtete System strukturell richtig wiedergeben. Bei jeder Aufgabe muss deshalb zunächst die Systemstruktur analysiert werden. Strukturelle Modelle wie das im Folgenden eingeführte Blockschaltbild können häufig schon aus einer verbalen Beschreibung des Systems abgeleitet werden, ohne dass die im System wirkenden physikalischen Gesetzmäßigkeiten und die Werte der darin vorkommenden Parameter im Einzelnen bekannt sind.
- Aus strukturellen Beschreibungen kann man sich einen Eindruck von den dynamischen Vorgängen innerhalb des betrachteten Systems verschaffen. Derartige Darstellungen tragen deshalb entscheidend zum Verständnis des Systemverhaltens bei. Sie zeigen, wie die Eingangssignale auf die inneren Signale des Systems wirken, wie die inneren Signale untereinander verkoppelt sind und wie aus diesen die Ausgangssignale gebildet werden. Es kann insbesondere abgelesen werden, an welchen Stellen im System Rückkopplungen auftreten

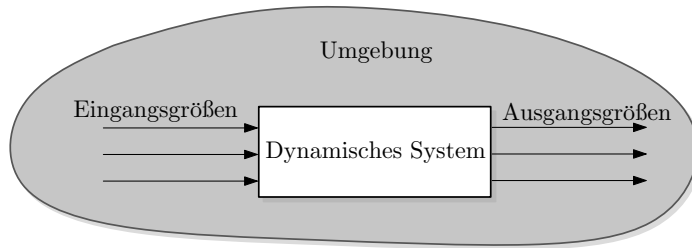
und ob das Gesamtsystem möglicherweise in unabhängige oder nur einseitig gekoppelte Teilsysteme zerlegt werden kann.

- Strukturelle Betrachtungen können auch dann angestellt werden, wenn die genaue Auslegung des Systems entweder nicht bekannt ist oder wenn die Analyseergebnisse weitgehend unabhängig von den Parameterwerten sein sollen. Diese Situation trifft man beispielsweise in der Projektierungsphase technischer Systeme, in der die Parameterwerte noch nicht festgelegt sind, man jedoch bereits wichtige Entscheidungen über die einzuführenden Messgrößen, die notwendigen Stellgrößen und die Struktur des Automatisierungssystems treffen muss. Strukturelle Entscheidungen sind auch dann wichtig, wenn die Analyseergebnisse auf mehrere gleichartige Systeme angewendet werden sollen, obwohl sich die Parameter dieser Systeme auf Grund von Fertigungstoleranzen oder durch die Berücksichtigung von Kundenwünschen unterscheiden.
- Beim Steuerungsentwurf kommt es zunächst auf die Wahl einer geeigneten Steuerungsstruktur an, also auf die Auswahl der verwendeten Mess- und Stellgrößen und die Festlegung der Kopplungen, die durch die Steuerung zwischen diesen Größen hergestellt werden sollen. Diese Strukturentscheidung ist mindestens genauso wichtig wie die später zu treffende Auswahl von geeigneten Parameterwerten oder Steuerungsalgorithmen, denn wenn man erst einmal mit einer falschen Struktur arbeitet, kann man entweder die gestellte Aufgabe gar nicht lösen oder die Lösung ist nicht so gut, wie sie mit einer anderen Steuerungsstruktur sein könnte.

Das im Folgenden eingeführte Blockschaltbild spielt deshalb bei der Lösung automatisierungstechnischer Aufgaben eine herausragende Rolle. Viele Fachleute glauben, dass sie eine Automatisierungsaufgabe erst dann vollkommen verstanden haben, wenn sie das Blockschaltbild vollständig aufzeichnen können. Das Blockschaltbild hat dabei häufig auch eine Vermittlerrolle zwischen den Fachleuten des zu automatisierenden Systems und Automatisierungstechnikern, die die Überwachung und Steuerung des betreffenden Systems entwickeln und in Betrieb nehmen sollen und sich dabei vor allem auf die Informationsflüsse zwischen dem zu steuernden System und der Automatisierungseinrichtung konzentrieren.

### 2.2.2 Blockschaltbild

Durch das Blockschaltbild wird beschrieben, aus welchen Teilsystemen sich ein System zusammensetzt und wie die Teilsysteme durch Signale verkoppelt sind. Diese auch als *Wirkungsschema* oder *Strukturbild* bezeichnete Darstellungsform grenzt ein dynamisches System von seiner Umgebung ab und symbolisiert es durch einen Block. Die Wechselwirkungen zwischen dem System und seiner Umgebung werden durch Signale beschrieben (Abb. 2.4). Dabei sind die Eingangsgrößen diejenigen Signale, durch die die Umwelt auf das System einwirkt. Sie sind deshalb von der Umwelt zum System gerichtet und beschreiben die äußeren Ursachen für alle im System ablaufenden Vorgänge. Die Ausgangsgrößen sind die vom System auf die Umwelt gerichteten Signale, durch die das System seine Umwelt beeinflusst.



**Abb. 2.4:** Blockschaltbild eines Systems

Für das Blockschaltbild gelten folgende Vereinbarungen:

- **Pfeile** stellen Signale dar.
- **Blöcke** kennzeichnen dynamische Systeme.
- Signale haben eine eindeutige **Wirkungsrichtung**, die durch die Pfeilrichtung beschrieben wird. Die Blöcke sind *rückwirkungsfrei*.

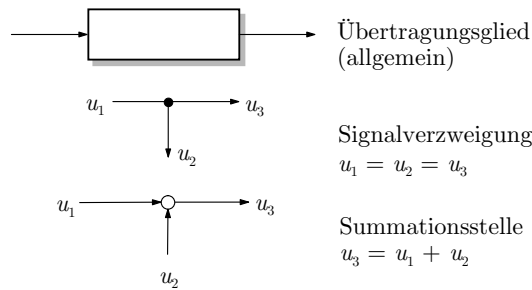
Die Blöcke können mehrere Eingangssignale und mehrere Ausgangssignale besitzen. Vektoriell zusammengefasste Signale werden durch einen Doppelpfeil  $\Rightarrow$  gekennzeichnet, wenn der Unterschied zwischen skalaren und vektoriellen Signalen hervorgehoben werden soll.

Der Pfeilanfang beschreibt, wo das Signal entsteht, und die Pfeilspitze weist auf den Block, der durch das Signal beeinflusst wird. Auf Grund der Rückwirkungsfreiheit der Blöcke haben die Vorgänge innerhalb eines Blocks keinen direkten Einfluss auf die Eingangssignale des Blocks. Wenn Rückkopplungen zu dem Block bestehen, in dem das Eingangssignal entstanden ist, muss dies durch ein weiteres Signal mit entgegengesetzter Wirkungsrichtung dargestellt werden.

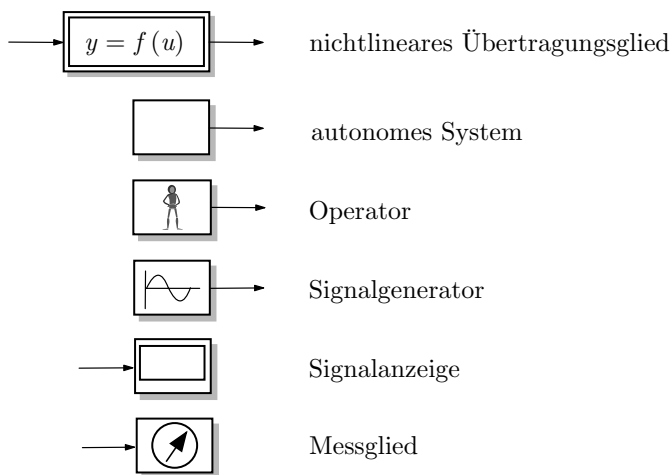
Das Blockschaltbild gilt für wertkontinuierliche und wertdiskrete Systeme gleichermaßen, denn es dient zur Kennzeichnung von Signalverknüpfungen zwischen den Teilsystemen unabhängig vom Charakter der Signale und Teilsysteme. Damit ist es auch die Grundlage für die Kopplungsanalyse (Abschn. 2.4) und die kompositionale Modellierung (Abschn. 3.6 und 11.6).

**Spezielle Symbole.** Für häufig wiederkehrende Abhängigkeiten zwischen Signalen hat man spezielle Symbole eingeführt, von denen drei in Abb. 2.5 zu sehen sind. Eine *Signalverzweigung* wird durch einen Punkt gekennzeichnet. Zwei oder mehreren Pfeile, die von diesem Punkt abgehen, stellen dasselbe Signal dar. Im Unterschied dazu kennzeichnet ein Kreis eine *Summationsstelle*, bei der zwei oder mehr ankommende Signale zu einem abgehenden Signal summiert werden. Setzt man ein Minuszeichen rechts neben die Pfeilspitze, so wird das betreffende Signal von den anderen subtrahiert. Die Summationsstelle kann natürlich nur bei wertkontinuierlichen Signalen verwendet werden.

Bei der Betrachtung kontinuierlicher Systeme werden *nichtlineare Systeme* besonders gekennzeichnet, wenn innerhalb eines Blockschaltbilds sonst nur lineare Teilsysteme vorkommen (Abb. 2.6). Blöcke ohne Eingangsgröße beschreiben autonome Systeme, deren Ausgangssignal eine von außen nicht beeinflusste Systembewegung beschreibt. Ein solcher Block kann beispielsweise einen Anlagenfahrer oder einen Signalgenerator darstellen, wenn diese unabhängig

**Abb. 2.5:** Grundsymbole des Blockschaltbilds

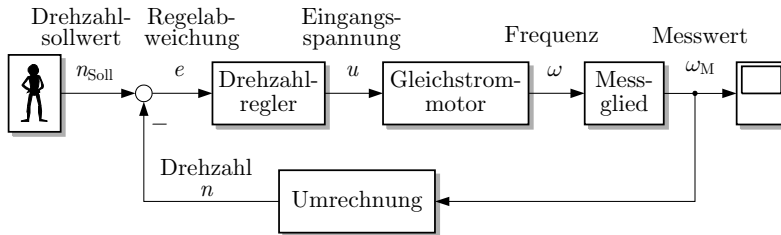
von ihrer Umgebung den Verlauf eines Signals vorgeben. Hängt allerdings ein Steuereingriff, den ein Anlagenfahrer für eine Anlage vorgibt, von Signalen der betreffenden Anlage ab, dann ist der Anlagenfahrer kein „autonomes System“, sondern erscheint als ein Block mit Eingangs- und Ausgangsgröße.

**Abb. 2.6:** Spezielle Symbole des Blockschaltbilds

Blöcke, die nur Eingangsgrößen, aber keine Ausgangsgrößen haben, stellen Messstellen oder Aufzeichnungsgeräte dar. Derartige Blöcke kommen in Blockschaltbildern vor, in denen gemessene oder angezeigte Signale besonders hervorgehoben werden sollen. Dabei muss das bei einer Simulation angezeigte Signal nicht unbedingt ein Signal sein, das an der betreffenden Anlage auch gemessen wird.

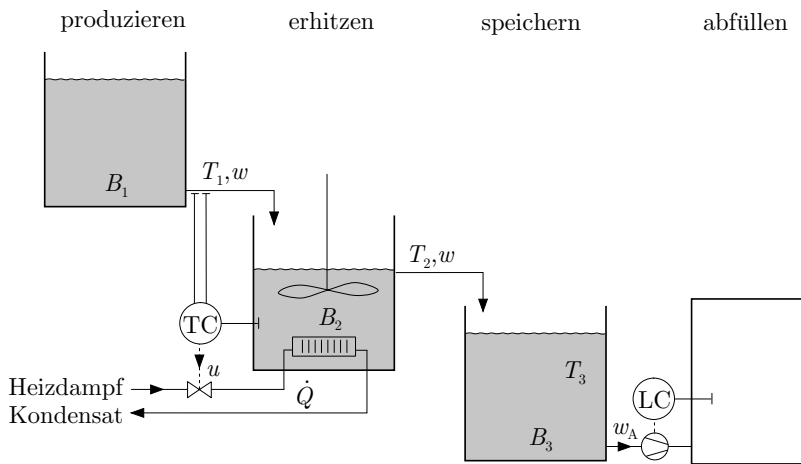
**Beispiel 2.3** Blockschaltbild eines drehzahlgetragenen Gleichstrommotors

Abbildung 2.7 zeigt die Struktur eines geregelten Gleichstrommotors. Die vier mittleren Blöcke stellen den Gleichstrommotor, den Drehzahlregler, das Messglied für die Drehkreisfrequenz und die Umrechnung der Drehfrequenz in die Drehzahl dar. Der Drehzahlsollwert wird von einem Menschen vorgegeben. Der Messwert wird aufgezeichnet.



**Abb. 2.7:** Blockschaltbild des drehzahlgetragenen Gleichstrommotors

Das Beispiel zeigt, dass man sich anhand von Blockschaltbildern vor der Aufstellung mathematischer Modelle einen Überblick über die Struktur des Systems verschaffen kann. Beim drehzahlgetragenen Motor sind – wie bei allen Regelkreisen – alle Blöcke in beiden Richtungen miteinander verkoppelt, so dass der Motor auf den Regler und umgekehrt der Regler auf den Motor einwirkt. Man kann deshalb nicht ohne Weiteres von den Eigenschaften der Blöcke auf die Eigenschaften des Gesamtsystems schließen. Eine quantitative Systemanalyse muss beispielsweise zeigen, ob der Regelkreis stabil ist.  $\square$



**Abb. 2.8:** Abfüllanlage (TC – Temperaturregler, LC – Füllstandsregler)



**Beispiel 2.4** Blockschaltbild einer Abfüllanlage

Im Behälter  $B_1$  der in Abb. 2.8 gezeigten Anlage wird ein flüssiges Produkt hergestellt, das im Behälter  $B_2$  auf die erforderliche Abfülltemperatur erhitzt, im Behälter  $B_3$  zwischengespeichert und anschließend in einer Abfülleinrichtung weiterverarbeitet wird. Im Behälter  $B_2$  wird die Flüssigkeit auf die Temperatur  $T_2$  erhöht, wobei die Heizung einen Wärmestrom  $\dot{Q}$  in die Flüssigkeit erzeugt. Dieser Wärmestrom hängt von der Ventilstellung  $u$  ab, die der Temperaturregler TC in Abhängigkeit von der aktuellen Flüssigkeitstemperatur  $T_2$ , der Temperatur  $T_1$  des Zulaufs und der Masse  $w$  der pro Zeiteinheit in den Behälter  $B_2$  fließenden Flüssigkeit vorgibt. Da die im Behälter  $B_2$  überlaufende Flüssigkeit in den Behälter  $B_3$  fließt, ist der Flüssigkeitsstrom zwischen  $B_1$  und  $B_2$  derselbe wie von  $B_2$  nach  $B_3$  und wird in beiden Fällen mit  $w$  bezeichnet. Der Füllstandsregler LC der Abfülleinrichtung steuert die Pumpe, die pro Zeiteinheit die Flüssigkeitsmenge  $w_A$  dem Behälter  $B_3$  entnimmt.

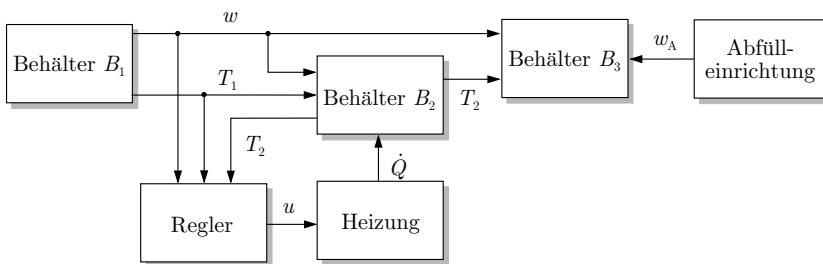
**Abb. 2.9:** Blockschaltbild der Abfüllanlage

Abbildung 2.9 zeigt das Blockschaltbild dieser Anlage, in dem die Behälter, die Abfüllanlage, die Heizung und der Temperaturregler als einzelne Blöcke auftreten. Dieses Bild entsteht direkt aus der zuvor angegebenen Beschreibung des Anlagenaufbaus.

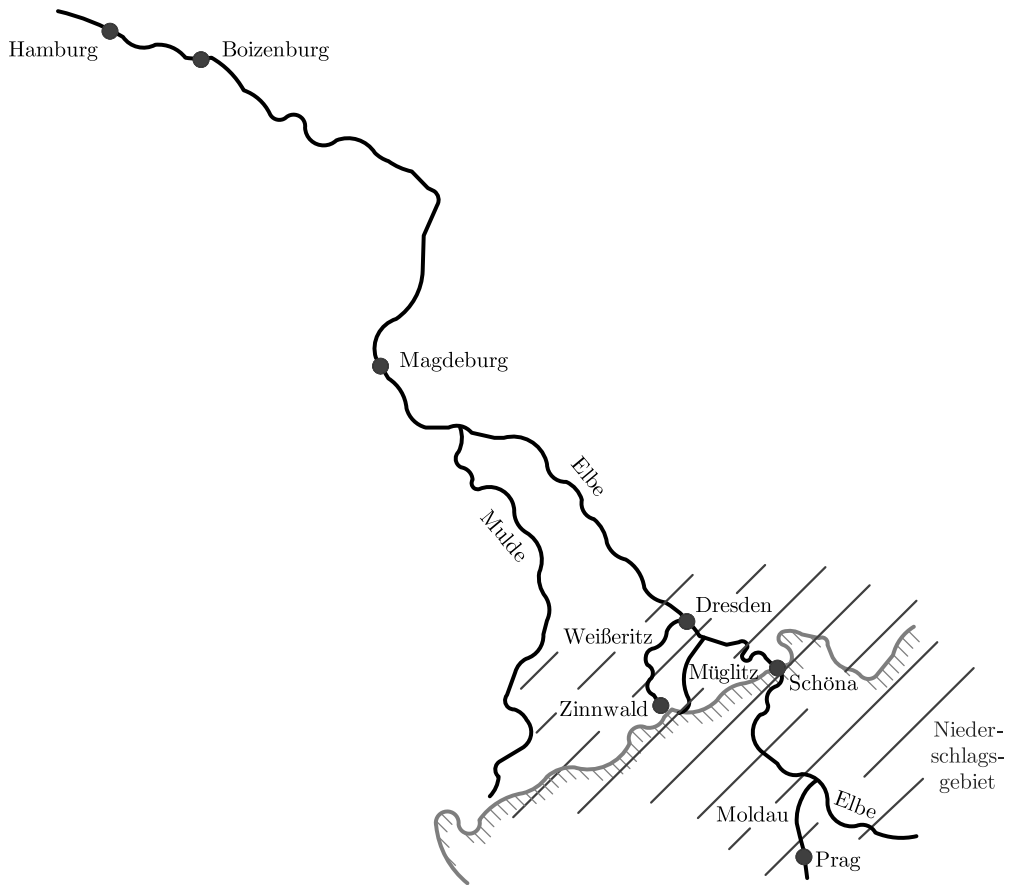
Es ist zu beachten, dass das Signal  $w_A$  als Eingangsgröße des Behälters  $B_3$  wirkt, obwohl die Flüssigkeit in umgekehrter Richtung aus dem Behälter  $B_3$  in die Abfülleinrichtung fließt. Dieses Signal wird in der Abfülleinrichtung in Abhängigkeit vom Füllstand in dem zu füllenden Behälter erzeugt und beeinflusst das Verhalten des Behälters  $B_3$ .  $\square$

**Beispiel 2.5** Analyse des Elbehochwassers anhand eines Blockschaltbilds

Blockschaltbilder eignen sich hervorragend, um die grundsätzlichen dynamischen Phänomene, die sich in Systemen abspielen, verstehen und erklären zu können. Als ein Beispiel dafür wird hier der Verlauf des Elbehochwassers vom August 2002 betrachtet, dessen Ursache ergiebige Niederschläge im Bayerischen Wald und im Erzgebirge waren (Abb. 2.10) und dessen Wirkungen sich entlang der Flüsse Moldau, Elbe und Mulde bis nach Hamburg fortpflanzten.

Durch das in Abb. 2.11 gezeigte Blockschaltbild werden die Zusammenhänge zwischen den Pegelständen in den einzelnen Flussabschnitten dargestellt. Jeder Block repräsentiert ein Teilsystem, das das Eingangssignal (Pegelstand am oberen Ende des betreffenden Flussabschnitts) in ein Ausgangssignal (Pegelstand am unteren Ende des Flussabschnitts) transformiert. Die Blöcke wurden in der Abbildung entsprechend der geografischen Lage der Teilsysteme angeordnet.

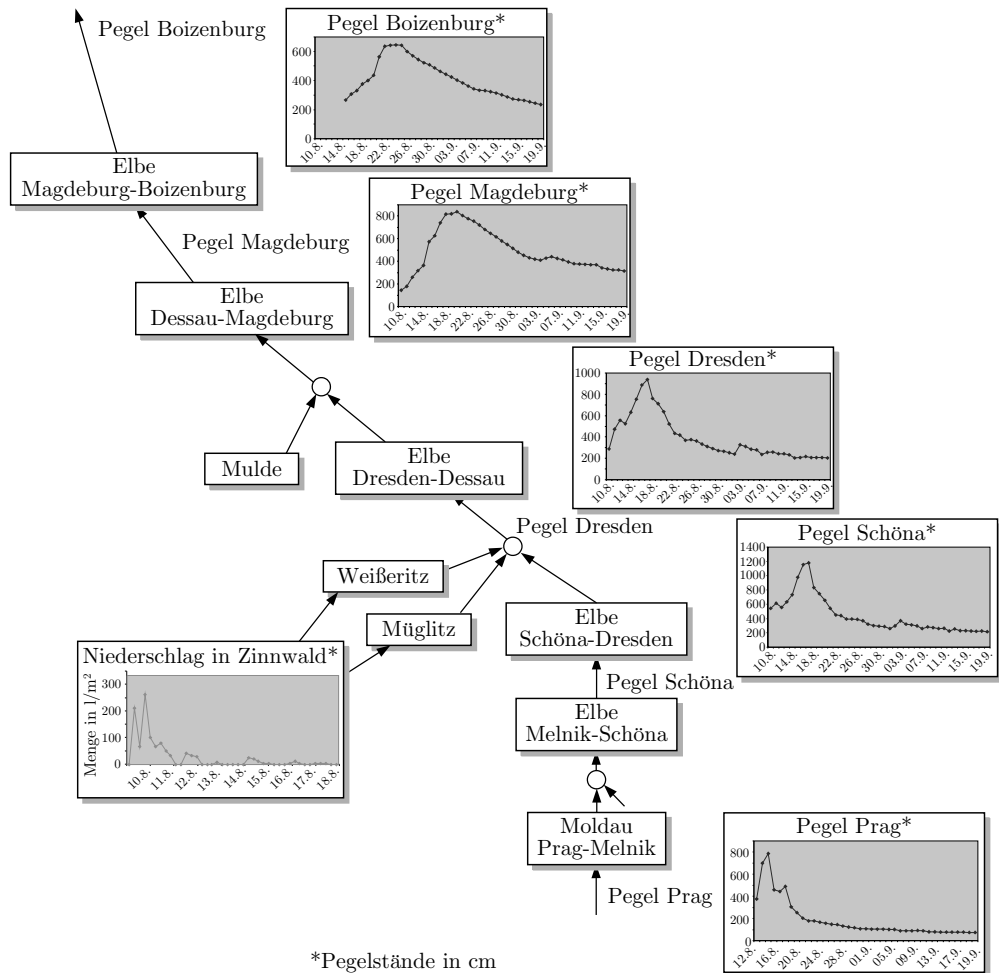
Charakteristisch für dynamische Systeme ist, dass sich Veränderungen ihres Eingangssignals erst mit einer Verzögerung als Veränderungen des Ausgangssignals bemerkbar machen. Bei dem Flussbeispiel ist dies offensichtlich durch die Tatsache begründet, dass das Wasser die Entfernung zwischen dem



**Abb. 2.10:** Landkarte des vom Elbehochwasser betroffenen Gebietes

oberen und unteren Pegel zurücklegen muss. Beim Elbehochwasser trat der höchste Pegelstand in Boizenburg deshalb eine Woche später auf als in Prag. Da sich das Wasser beim Anstieg auch in ufernahe Regionen ausbreitete, veränderte sich auch der Charakter des Hochwassers. Die an das Blockschaltbild angetragenen Kurven zeigen, dass aus einem sehr kurzen, aber hohen Ausschlag des Prager Pegels bei Boizenburg ein nicht so hohes, aber dafür wesentlich länger anhaltendes Maximum wurde.

Das Verfolgen der Pegelstände ist ein Beispiel für eine Prozessüberwachung. Aus den dabei erhaltenen Ergebnissen kann man Steuerungsentscheidungen ableiten. Steuereingriffe sind bei diesem System nur durch Talsperren und Wehre möglich. Eine für das Hochwasser wichtige Talsperre liegt im Flusslauf der Weißeritz zwischen dem Erzgebirge und Dresden bei Malter. Durch ein Schließen der Talsperre hätte man das Wasser zwischenspeichern können, was eine zusätzliche Verzögerung und dadurch eine Verflachung des Pegelverlaufs unterhalb der Talsperre zur Folge gehabt hätte. Diese Stellgröße war im August 2002 nicht wirksam, weil die Talsperre zuvor für die Urlaubszeit gut gefüllt worden war, so dass sie kurz nach den Regenfällen im Erzgebirge überlief. Der Flusslauf war damit nicht mehr steuerbar, was zu Verwüstungen entlang der Weißeritz in Freital und Dresden führte. □



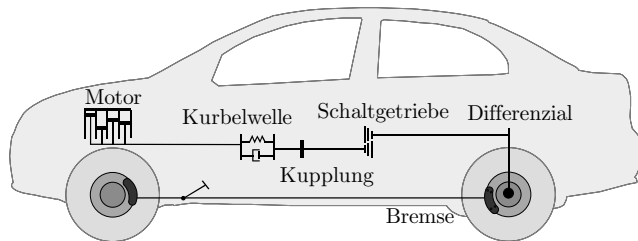
**Abb. 2.11:** Darstellung des strukturellen Zusammenhangs zwischen den Pegelständen durch ein Blockschaltbild

**Aufgabe 2.1\*** Blockschaltbild des Antriebsstrangs eines Kraftfahrzeugs

Abbildung 2.12 zeigt schematisch die in dieser Aufgabe zu betrachtenden Elemente des Antriebsstrangs eines Kraftfahrzeugs.

- Überlegen Sie sich, wie das vom Motor erzeugte Antriebsmoment über Kurbelwelle, Kupplung, Schaltgetriebe und Differential auf die Antriebsräder übertragen wird und zeichnen Sie ein Blockschaltbild, das die Signalübertragung vom Gaspedalwinkel zur Fahrzeuggeschwindigkeit beschreibt. Tragen Sie weitere wichtige Einflussgrößen der Geschwindigkeit in dieses Blockschaltbild ein.
- Welche Signale sind kontinuierlich, welche diskret?

3. Durch welche zusätzlichen Signalverknüpfungen entsteht der Regelkreis für die Geschwindigkeitsregelung („Tempomat“), über die heute viele Fahrzeuge verfügen?
4. Wie verändert sich das Blockschaltbild, wenn das Fahrzeug mit einem Automatikgetriebe ausgerüstet ist? □



**Abb. 2.12:** Antriebsstrang eines Kraftfahrzeugs

### **Aufgabe 2.2\*** *Blockschaltbild einer Motorsteuerung*

Anhand der folgenden Funktionsbeschreibung soll das Blockschaltbild der Motorsteuerung eines Dieselmotors aufgestellt werden.

Der Fahrer fordert das gewünschte Drehmoment des Motors durch Betätigen des Gaspedals, wobei einerseits die Gaspedalstellung über einen Bowdenzug die Drosselklappenstellung bestimmt und andererseits der durch einen Pedalwertgeber gemessene Pedalwinkel in die Berechnung der Einspritzmenge eingeht. Die vom Zylinder angesaugte Luft wird durch die Drosselklappenstellung beeinflusst, die eingespritzte Treibstoffmenge durch die Öffnungszeit des Einspritzventils, das durch einen Injektor betätigt wird. Der Zeitpunkt des Einspritzbeginns, der auf die Kurbelwinkelstellung bezogen und als Winkel der Kurbelwelle zum oberen Totpunkt beschrieben wird, ändert sich mit der Drehzahl. Der Wert wird mit einem nichtlinearen Spritzbeginnfeld berechnet, das den Spritzbeginn in Abhängigkeit von der Drehzahl und der Einspritzmenge darstellt. Der dabei erhaltene Wert ist der Sollwert für die Spritzbeginnregelung, durch die der tatsächliche Spritzbeginn dem gewünschten Wert angepasst wird. Dabei dient ein Nadelbewegungsfühler zur Bestimmung des tatsächlichen Spritzbeginns. Die Regelung gleicht also Parametertoleranzen aus.

Damit eine vollständige Verbrennung des Kraftstoffes erreicht wird, muss das Luft-Kraftstoffverhältnis bestimmte Bedingungen erfüllen. Die Einspritzmenge wird deshalb aus dem gemessenen Luftmassenstrom in den Zylinder und dem gemessenen Gaspedalwinkel berechnet. Da die Einspritzmenge nicht exakt dosiert werden kann und die einzelnen Zylinder ohne eine entsprechende Mengenkorrektur auf Grund von Parametertoleranzen unterschiedliche Kraftstoffmengen verbrennen würden, was zu einer spürbaren Drehzahländerung führen würde, werden die zylinderabhängigen Drehzahländerungen bestimmt und die Einspritzmengen dementsprechend korrigiert.

Bei der Aufstellung des Blockschaltbilds soll der Motor einschließlich des Antriebsstrangs nicht weiter untersucht und dementsprechend durch einen Block dargestellt werden. □

### **2.2.3 Signalflussgraph**

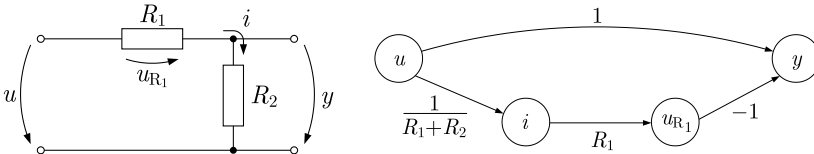
Der *Signalflussgraph* ist ein gerichteter Graph, bei dem die Knoten Signale und die Kanten Signalverknüpfungen beschreiben. Wenn man die Knoten des Signalflussgraphen mit den Blöcken des Blockschaltbilds und die Kanten mit den Pfeilen gleichsetzt, ist die Bedeutung der Knoten und Kanten gegenüber dem Blockschaltbild also gerade vertauscht.

Der Signalflussgraph wird vor allem bei linearen Systemen eingesetzt, bei denen er die folgende Interpretation hat:

Der Wert eines Signals (Knotens) hängt von den Signalen ab, von denen gerichtete Kanten auf das betreffende Signal zeigen. Die Wirkung jeder Kante ergibt sich als Produkt des Kantengewichts und des Wertes des Signals, von dem die Kante ausgeht. Zeigen zwei oder mehrere Kanten auf denselben Knoten, so addieren sich ihre Wirkungen.

### Beispiel 2.6 Signalflussgraph einer einfachen Schaltung

Für die in Abb. 2.13 links dargestellte Schaltung mit der Eingangsspannung  $u(t)$  und der Ausgangsspannung  $y(t)$  kann man die Signale  $u_{R_1}(t)$  und  $i(t)$  einführen und das folgende Gleichungssystem aufstellen:



**Abb. 2.13:** Schaltung (links) und Signalflussgraph der Schaltung (rechts)

$$\begin{aligned} i(t) &= \frac{1}{R_1 + R_2} u(t) \\ u_{R_1}(t) &= R_1 i(t) \\ y(t) &= u(t) - u_{R_1}(t). \end{aligned}$$

Diese Gleichungen führen auf den im rechten Teil von Abb. 2.13 gezeigten Signalflussgraphen, dessen Kanten mit den Parametern  $\frac{1}{R_1 + R_2}$ ,  $R_1$ , 1 und  $-1$  gewichtet sind. Die in der dritten Gleichung angegebene Differenz erhält man anhand der oben genannten Regel durch Überlagerung der beiden auf den Knoten  $y$  zeigenden Kanten.  $\square$

Der Signalflussgraph wird in den nachfolgenden Kapiteln zur Veranschaulichung der Struktur von Zustandsraummodellen verwendet. Dort wird auch das Integrationszeichen als Wichtung der Kanten zugelassen, das die zeitliche Ableitung eines Signals mit dem Signal selbst verknüpft. Der im Kap. 5 eingeführte Strukturgraph ist ein Signalflussgraph, bei dem jede Zustandsvariable  $x_i(t)$  mit ihrer Ableitung  $\dot{x}_i(t)$  durch einen gemeinsamen Knoten dargestellt wird.

## 2.3 Dekomposition und Aggregation von Systemen

Der Begriff des Systems wird nicht nur in der Automatisierungstechnik, sondern auch in vielen anderen Bereichen verwendet und dabei auch mit unterschiedlichen Bedeutungen gebraucht. Unter einem *System* versteht man sehr häufig eine Menge von Elementen, die in einer Beziehung zueinander stehen. Auf diesen Aspekt des Systembegriffs wird im Folgenden eingegangen.

Aus einer sehr „fernen“ Sicht stellt ein dynamisches System ein Objekt dar, das durch Eingangssignale beeinflusst wird und dessen Verhalten man an Ausgangssignalen erkennen kann. Diese Betrachtungsweise wird im Blockschaltbild durch einen Block mit den Eingangs- und Ausgangssignalen ausgedrückt (Abb. 2.14 links).

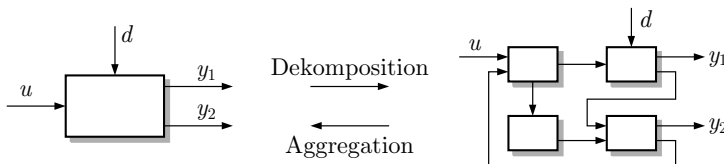


Abb. 2.14: Dekomposition und Aggregation eines Systems

Viele Systeme sind jedoch keine unstrukturierten Objekte, sondern bestehen aus mehreren Teilsystemen. Wenn man sich diese Systeme genauer ansieht, erkennt man, welche Signale auf welche Elemente des betreffenden Systems wirken, wie die Teilsysteme untereinander verknüpft sind und in welchen Teilsystemen die Ausgangssignale entstehen (Abb. 2.14 rechts).

Die Zerlegung eines Systems in seine Teilsysteme nennt man *Dekomposition*. Wichtig ist, dass die dabei voneinander abgegrenzten Teilsysteme nicht isoliert existieren, sondern sich in ihrem Verhalten gegenseitig beeinflussen. Die Kopplungen werden durch Signale repräsentiert, die bei der im linken Teil der Abbildung gezeigten Gesamtsystemsicht nicht vorkommen.

Der umgekehrte Weg, bei dem man Teilsysteme zu in sich geschlossenen Einheiten zusammenfasst, wird *Aggregation* genannt. Man verwendet ihn, wenn man sich nicht für Details eines Systems interessiert, sondern das Gesamtverhalten untersuchen oder das System mit anderen Systemen in Beziehung setzen will.

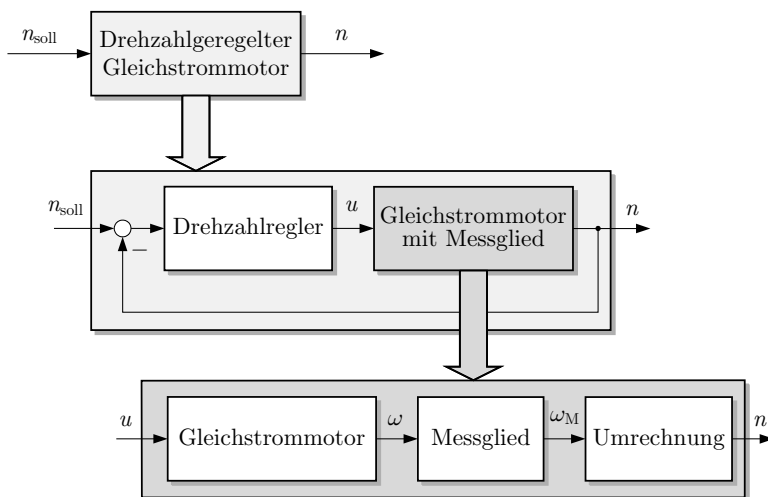
Eine Dekomposition folgt häufig der natürlichen Einteilung des betrachteten Systems in seine Teilsysteme. Aber auch wenn das System keine offensichtliche Untergliederung in Teilsysteme nahe legt, kann man es in Teilsysteme aufspalten. Die Dekomposition vereinfacht jedoch vor allem dann die Systemanalyse, wenn sie auf schwach gekoppelte Teilsysteme führt, bei denen die Analyse der Teilsysteme unter Vernachlässigung der Kopplungen zu ähnlichen Ergebnissen kommt wie eine Analyse unter Berücksichtigung der Verkopplungen. Die Vernachlässigung von Kopplungen führt dann zu einer wesentlichen Komplexitätsreduktion.

Die Dekomposition vereinfacht die Modellbildung, weil man das Modell des Gesamtsystems aus Modellen der Teilsysteme aufbauen kann. Um die Teilsystemmodelle aufzuschreiben, muss man nur das betreffende Teilsystem betrachten und nicht mehr das vollständige Gesamtsystem. Dieses Vorgehen wird als *komponentenorientierte Modellbildung* oder *kompositionale Modellbildung* bezeichnet.

**Hierarchische Modellbildung.** Dekomposition und Aggregation kann man mehrfach hintereinander ausführen. Man kann also die beim ersten Dekompositionsschritt erhaltenen Teilsysteme gegebenenfalls wiederum in Teilsysteme zerlegen, so dass man dann eine hierarchische Gliederung des Gesamtsystems bekommt, wie sie in der Automatisierungstechnik weit verbreitet ist (vgl. Abb. 1.22 auf S. 37). Die auf unterschiedlichen Leitebenen angeordneten Automatisierungsaufgaben betreffen unterschiedliche Teile des zu steuernden Systems und deshalb eine unterschiedliche Auswahl der Teilsysteme, für die die Aufgabe gelöst werden muss. Es entsteht dabei ein Blockschaltbild, dessen Blöcke aus Blöcken einer tieferen Hierarchiestufe zusammengesetzt sind.

**Beispiel 2.7** Hierarchische Modellbildung des drehzahlgeregelten Gleichstrommotors

Abbildung 2.15 zeigt hierarchisch gegliederte Blockschaltbilder eines geregelten Gleichstrommotors. Das Gesamtsystem kann als einzelner Block mit der Eingangsgröße  $n_{\text{soll}}$  (Solldrehzahl) und der Ausgangsgröße  $n$  (Drehzahl) dargestellt werden. Wenn man genauer in dieses System „hineinsieht“, kann man einen Regelkreis erkennen, der aus dem Gleichstrommotor einschließlich dem Messglied als Regelstrecke und dem Drehzahlregler besteht. Betrachtet man den Block „Gleichstrommotor mit Messglied“ noch genauer, so erkennt man, dass dieser Block aus dem Gleichstrommotor, dem Messglied und einem Glied zur Umrechnung des Messwertes  $\omega_M$  in die Drehzahl  $n$  besteht.



**Abb. 2.15:** Hierarchisch strukturiertes Blockschaltbild eines Gleichstrommotors

Je genauer man sich das System ansieht, desto mehr Blöcke und Kopplungssignale zwischen den Blöcken erkennt man. Geht man von einer Abstraktionsebene zur nächsthöheren, so verschwinden die inneren Signale und es bleibt ein Block mit den externen Eingangsgrößen und externen Ausgangsgrößen übrig. Welche Abstraktionsebene die beste ist, hängt von der betrachteten Aufgabe ab. □

## 2.4 Kopplungsanalyse

Für viele automatisierungstechnische Aufgaben ist es wichtig zu wissen, welche Teilsysteme sich gegenseitig beeinflussen und in welcher Richtung die Kopplungen auftreten. Von besonderer Bedeutung sind dabei Teilsysteme, die nicht nur in einer Richtung, sondern in beiden Richtungen untereinander verkoppelt sind. Derartige Teilsysteme heißen *stark zusammenhängend*. Die bereits mehrfach erwähnten Rückkopplungsstrukturen (Kreisstrukturen), die für automatisierungstechnische Anwendungen typisch sind, enthalten stark zusammenhängende Teilsysteme. Andererseits ergeben sich wichtige Vereinfachungen von Analyse- oder Entwurfsaufgaben, wenn man weiß, dass bestimmte Teilsysteme entweder gar nicht oder nur in einer Richtung miteinander in Wechselwirkung stehen.

Die Kopplungsstruktur kann man anhand des Blockschaltbilds analysieren. Jeder Block stellt ein Teilsystem dar. Die Kopplungen sind durch die die Blöcke verbindenden Signale beschrieben, wobei im Folgenden die Begriffe Signal und Kante synonym verwendet werden. Da es lediglich auf die Frage ankommt, ob Teilsysteme gekoppelt sind oder nicht, werden die im Blockschaltbild auftretenden Pfade zwischen den Blöcken untersucht, wobei es nicht auf die physikalische Bedeutung der Signale und auch nicht auf die Anzahl der dieselben Blöcke verbindenden Pfeile ankommt. Um graphentheoretische Begriffe anwenden zu können, fasst man das Blockschaltbild als gerichteten Graphen auf, dessen Knoten  $i$ , ( $i = 1, 2, \dots, N$ ) die Blöcke und dessen Kanten die Kopplungen beschreiben. Dieser Graph wird als *Kopplungsgraph* bezeichnet. Um den direkten Bezug zum Blockschaltbild zu erhalten, wird im Folgenden jedoch mit dem Blockschaltbild gearbeitet und jedes Teilsystem mit dem Index  $i$  des betreffenden Knotens identifiziert.

Vom Teilsystem  $i$  und zum Teilsystem  $j$  gibt es einen Pfad, wenn man vom Teilsystem  $i$  entweder auf einem Pfeil oder über mehrere Blöcke und Pfeile zum Teilsystem  $j$  kommt.

### Definition 2.1 (Stark zusammenhängende Teilsysteme)

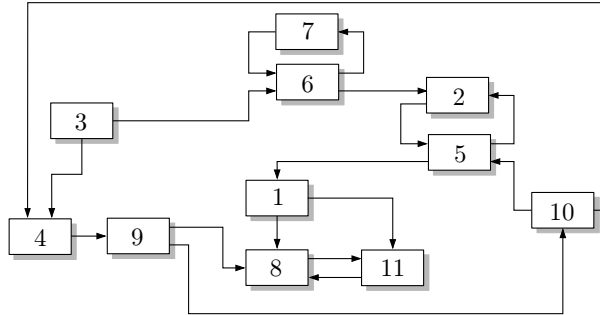
Die Teilsysteme  $i$  und  $j$  heißen *stark zusammenhängend* (*stark verbunden*), wenn es im Blockschaltbild sowohl einen Pfad vom Teilsystem  $i$  zum Teilsystem  $j$  als auch einen Pfad vom Teilsystem  $j$  zum Teilsystem  $i$  gibt.

Diese Definition verwendet direkt die in der Graphentheorie gebräuchlichen Begriffe Pfad und stark zusammenhängend. Man kann deshalb die in der Graphentheorie entwickelten Algorithmen für die Bestimmung stark zusammenhängender Knoten verwenden, wenn man das Blockschaltbild als gerichteten Graphen interpretiert. Dabei reicht es, zwischen zwei Knoten  $i$  und  $j$  nur eine Kante zu zeichnen, selbst wenn die Teilsysteme  $i$  und  $j$  durch mehrere Signale in derselben Richtung verknüpft sind. Sind die Teilsysteme in beide Richtungen verkoppelt, so zeichnet man zwei Kanten, für jede Richtung eine.

Die Graphentheorie zeigt, dass die Eigenschaft „stark zusammenhängend“ eine Äquivalenzrelation ist. Das heißt insbesondere, wenn die Teilsysteme  $i$  und  $j$  sowie  $i$  und  $k$  stark zusammenhängend sind, so sind dies auch die Teilsysteme  $j$  und  $i$  sowie  $j$  und  $k$ . Man kann deshalb die Menge der in einem Blockschaltbild auftretenden Teilsysteme in disjunkte Mengen  $S_i$  (Äquivalenzklassen) zerlegen, wobei zwei Teilsysteme genau dann zur selben Menge  $S_i$  gehören, wenn sie stark zusammenhängend sind. Fasst man die dabei erhaltenen Mengen zu



jeweils einem neuen Teilsystem zusammen, so treten nur einseitige Kopplungen zwischen den neuen Teilsystemen auf. In der Graphentheorie bezeichnet man den neuen Graphen als Kondensation des Kopplungsgraphen.



**Abb. 2.16:** Beispiel für ein gekoppeltes System

#### Beispiel 2.8 Kopplungsanalyse eines Systems

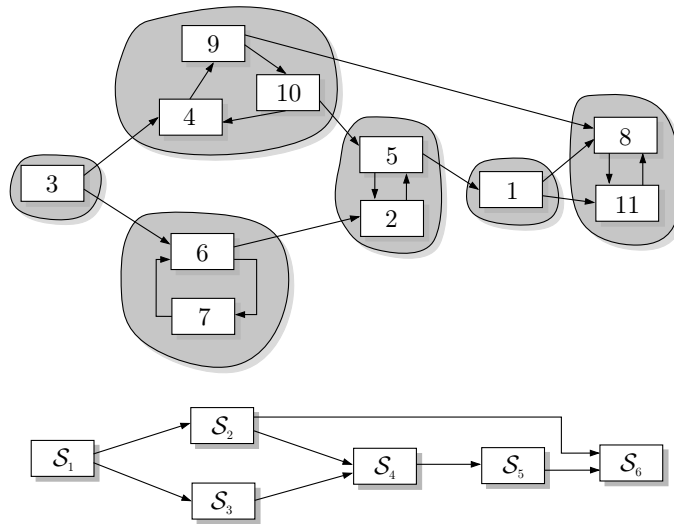
Das in Abb. 2.16 gezeigte System besteht aus elf Teilsystemen, die in einer auf den ersten Blick unübersichtlichen Weise verkoppelt sind. Insbesondere sieht man nicht ohne Weiteres, welche Teilsysteme untereinander stark zusammenhängend sind und zwischen welchen Teilsystemen nur eine einseitige Kopplung besteht. Eine genauere Analyse zeigt jedoch, dass die elf Teilsysteme sechs Mengen stark zusammenhängender Teilsysteme bilden:

$$\begin{aligned} \mathcal{S}_1 &= \{3\}, & \mathcal{S}_2 &= \{4, 9, 10\} \\ \mathcal{S}_3 &= \{6, 7\}, & \mathcal{S}_4 &= \{2, 5\} \\ \mathcal{S}_5 &= \{1\}, & \mathcal{S}_6 &= \{8, 11\}. \end{aligned}$$

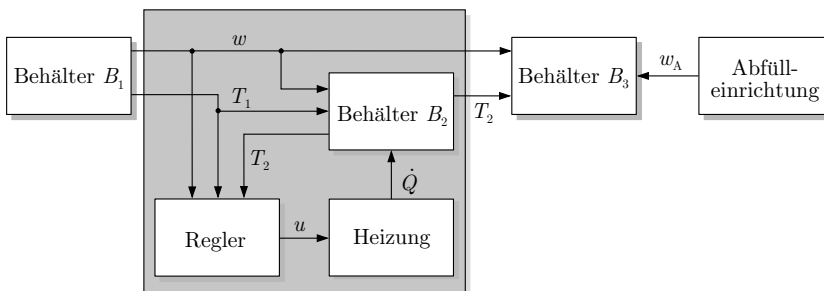
Die Teilsysteme 1 und 3 bilden dabei jeweils eine Menge für sich, weil sie mit keinem anderen Teilsystem stark zusammenhängend sind. Wie Abb. 2.17 zeigt, bilden die Teilsystemmengen  $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_6$  eine Reihen-Parallelschaltung. □

#### Beispiel 2.9 Kopplungsstruktur einer Abfüllanlage

Das im Beispiel 2.4 aufgestellte Blockschaltbild wird jetzt zur Kopplungsanalyse der Abfüllanlage eingesetzt. Dabei stellt sich heraus, dass der Regler, der Behälter  $B_2$  und die Heizung stark zusammenhängend sind, während die anderen Blöcke nur einseitige Kopplungen aufweisen (Abb. 2.18). Das heißt, dass für den Entwurf des Reglers nur der Behälter  $B_1$  zusammen mit der Heizung betrachtet werden muss. □



**Abb. 2.17:** Zerlegung des gekoppelten Systems aus Abb. 2.16 in stark zusammenhängende Teilsysteme



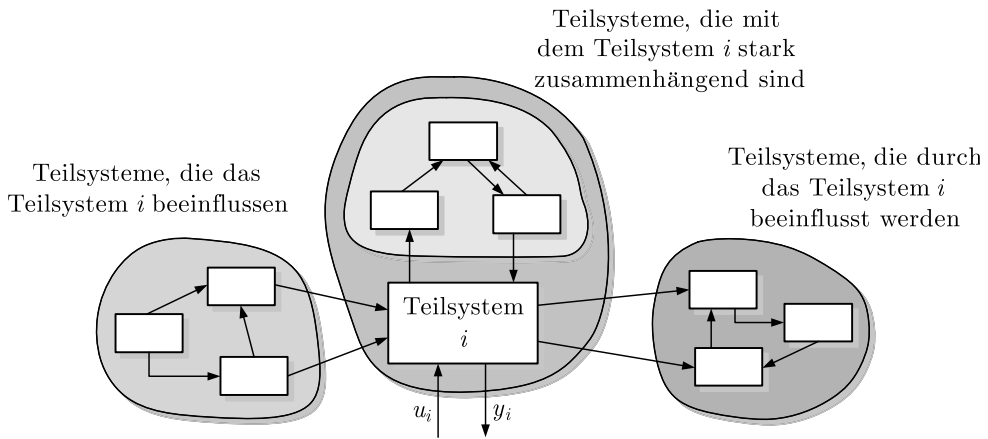
**Abb. 2.18:** Kopplungsstruktur der Abfüllanlage

**Zerlegung von Systemen.** Nach der Zerlegung des Gesamtsystems in Mengen stark zusammenhängender Teilsysteme kann man für jedes Teilsystem  $i$  ablesen,

- welche anderen Teilsysteme mit dem Teilsystem  $i$  stark zusammenhängend sind,
- welche Teilsysteme einen Einfluss auf das Teilsystem  $i$  haben, aber vom Teilsystem  $i$  nicht beeinflusst werden,
- auf welche Teilsysteme das Teilsystem  $i$  einen Einfluss ausübt, ohne von diesen Teilsystemen beeinflusst zu werden.

Folglich kann man jedes Gesamtsystem vom Standpunkt eines Teilsystems  $i$  aus in der in Abb. 2.19 gezeigten Weise zerlegen.

Diese Zerlegung hat unmittelbare Konsequenzen für die Lösung von Automatisierungsaufgaben für das Teilsystem  $i$ . Wenn man das Verhalten des Teilsystems  $i$  bezüglich des Ein-



**Abb. 2.19:** Betrachtung des Gesamtsystems vom Standpunkt des Teilsystems  $i$  aus

gangs  $u_i(t)$  und des Ausgangs  $y_i(t)$  beschreiben will, muss man in das Modell auch alle mit dem Teilsystem  $i$  stark zusammenhängenden Teilsysteme einbeziehen. Im Gegensatz dazu kann man den Einfluss, den die in der Abbildung links eingetragene Gruppe von Teilsystemen auf das Teilsystem  $i$  hat, als Störungen interpretieren, denen das aufzustellende Modell unterliegt. Andererseits spielen die rechts dargestellten Teilsysteme für das Verhalten von Teilsystem  $i$  keine Rolle, denn das Teilsystem  $i$  beeinflusst deren Verhalten, ohne dass eine Rückwirkung existiert.

Was für das Modell gesagt wurde, gilt für die Lösung aller Automatisierungsaufgaben, in denen dieses Modell verwendet wird. So muss eine Regelung oder eine Steuerung des Teilsystems  $i$  die mit diesem Teilsystem stark zusammenhängenden Teilsysteme berücksichtigen.

Aus Abb. 2.19 geht auch hervor, dass sich am Verhalten des Teilsystems  $i$  nichts ändert, wenn weitere Teilsysteme zu der rechts gezeichneten Teilsystemgruppe hinzukommen. Wenn weitere Teilsysteme zu der links dargestellten Gruppe hinzukommen oder sich das Verhalten dieser Teilsysteme ändert, so hat dies einen Einfluss auf das Verhalten von Teilsystem  $i$ , weil sich damit die als Störungen interpretierten Eingangssignale ändern. Den größten Einfluss hat jedoch eine Veränderung der mit dem Teilsystem  $i$  stark zusammenhängenden Teilsysteme, weil dies das Verhalten der gesamten Teilsystemgruppe grundlegend ändern kann.

**Diskussion.** Bei der Analyse des Zusammenhangs von Teilsystemen wird lediglich zwischen vorhandenen und nicht vorhandenen Kopplungen unterschieden, nicht jedoch zwischen starken und schwachen Kopplungen. Das Ergebnis ist eine Zerlegung des Gesamtsystems in Mengen stark zusammenhängender Teilsysteme, wobei Teilsysteme, die zu unterschiedlichen Mengen gehören, nur in einer Richtung oder gar nicht miteinander verkoppelt sind.

Weitere Vereinfachungen von Analyse- und Entwurfsaufgaben erhält man, wenn einige der auftretenden Kopplungen schwach sind. Wenn man weiß, welche Kopplungen schwach sind, kann man *annehmen*, dass diese Kopplungen gar nicht vorhanden sind und das System weiter

zerlegen. Dies ist ein in der Praxis häufig angewendeter Weg zur Vereinfachung von Analyse- oder Entwurfsaufgaben. Das Problem besteht jedoch in der Definition dessen, was eine schwache Kopplung ist. Die Bewertung einer Kopplung als schwach oder stark hängt nämlich nicht nur von den Signalamplituden oder den Verstärkungsfaktoren ab, mit denen diese Signale in das Modell der betreffenden Teilsysteme einkoppeln, sondern auch von den Eigenschaften, die man untersuchen möchte. Es gibt deshalb keine allgemeingültige Definition für schwache Kopplungen.

### **Aufgabe 2.3\*** Analyse der erweiterten Abfüllanlage

Wie verändert sich die Kopplungsstruktur der im Beispiel 2.4 behandelten Abfüllanlage, wenn der Behälter  $B_3$  einen Füllstandsregler erhält, der durch Ansteuerung eines in die Verbindungsleitung zwischen den Behältern  $B_2$  und  $B_3$  zusätzlich eingebauten Ventils den Füllstand  $h$  im Behälter  $B_3$  auf einem vorgegebenen Sollwert hält? Welche Auswirkungen hat dies für das Verhalten der gesamten Anlage?  $\square$

### **Aufgabe 2.4\*** Kopplungsanalyse eines Elektroenergieversorgungsnetzes

Abbildung 2.20 zeigt das Blockschaltbild eines aus drei Teilnetzen bestehenden Elektroenergieversorgungsnetzes, dessen Wirkleistungsverhalten hier betrachtet wird.

Im Netz 1 arbeiten zwei unterschiedliche Kraftwerke, während in den beiden anderen Netzen alle Kraftwerke zu einem Block zusammengefasst sind. Die Verbraucherleistung ist in jedem Netz in einen frequenzabhängigen Anteil  $p_V(t)$ , der Ausgangsgröße des Blocks „Verbraucher“ ist, und einen als Störgröße wirkenden frequenzunabhängigen Anteil  $p_L(t)$  aufgeteilt.

Die Eingangsgrößen der Blöcke „Kraftwerk“ sind die Sollwerte  $p_{GSoll}(t)$  der Leistungsregler, die Ausgangsgrößen die erzeugten Leistungen  $p_G(t)$ . Eine wichtige Regelungsaufgabe besteht darin, die Netzfrequenz konstant zu halten. Damit wird gleichzeitig erreicht, dass die im Netz erzeugte Leistung gleich der im selben Moment verbrauchten Leistung ist.

Eine solche Regelung besteht aus einer Primärregelung und einer Sekundärregelung. Die Primärregelung ist durch die mit  $k_{pi}$  gekennzeichneten Blöcke im Blockschaltbild eingetragen. Demzufolge erhöhen die Kraftwerke ihre Leistung proportional zur Abweichung  $f$  der Frequenz von der Nominalfrequenz (50 Hz). Die Sekundärregelung ist nicht im Blockschaltbild eingetragen. Sie verbindet die angegebenen Ausgangsgrößen  $y_i$  mit den Eingangsgrößen  $u_i$ . Das Bild zeigt also die Regelstrecke für die Sekundärregelung.

1. Untersuchen Sie, welche Kraftwerke und Verbraucher stark zusammenhängend sind.
2. Wie verändert sich die Kopplungsstruktur, wenn das Netz ohne Primärregelung betrachtet wird?
3. Aggregieren Sie die drei Teilnetze zu jeweils einem Block. Sind diese drei Blöcke stark zusammenhängend?
4. Beim Entwurf der in den Kraftwerken installierten Leistungsregler geht man davon aus, dass das Kraftwerk auf ein „starrs Netz“ arbeitet, dessen Netzfrequenz  $f$  von der Leistung des betrachteten Kraftwerks unabhängig ist. Dies hat zur Folge, dass die Frequenz  $f$  kein Signal mehr ist, sondern als fester Parameter in die Modelle der Teilsysteme eingeht, für die sie bisher als Eingangsgrößen fungierte. Wie verändert diese Annahme die Kopplungseigenschaften des Netzes?
5. Ist die Annahme eines „starrs Netzes“ auch für den Entwurf der Sekundärregelung sinnvoll?  $\square$

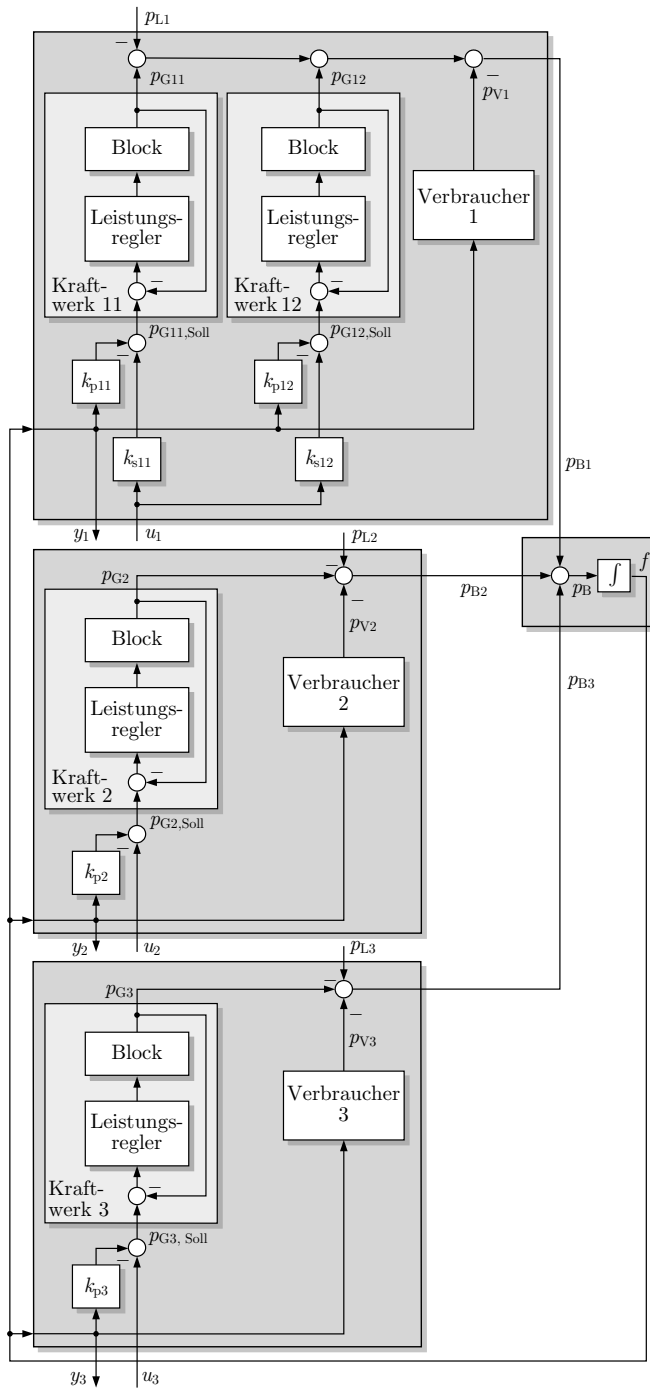


Abb. 2.20: Modell eines Elektroenergienetzes

## 2.5 Steuerungen in der offenen Wirkungskette und im geschlossenen Kreis

Eine wichtige strukturelle Entscheidung beim Aufbau eines Automatisierungssystems betrifft die Frage, ob eine Steuerung in der offenen Wirkungskette oder im geschlossenen Wirkungskreis realisiert werden soll. Abbildung 2.21 zeigt die beiden Alternativen. Beide werden dazu verwendet, das Signal  $y(t)$  dem vorgegebenen Signal  $w(t)$  nachzuführen, so dass (wenigstens näherungsweise) die Beziehung  $y(t) = w(t)$  gilt.

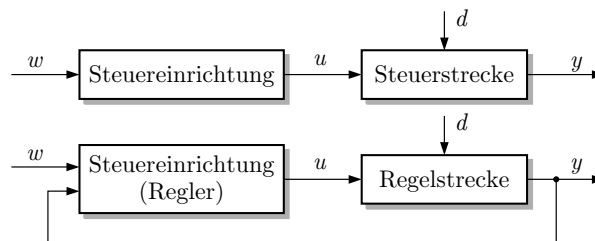


Abb. 2.21: Steuerung in der offenen Wirkungskette (oben) und im geschlossenen Wirkungskreis (unten)

- **Steuerung in der offenen Wirkungskette.** Bei der oben dargestellten Reihenschaltung berechnet die Steuereinrichtung die Stellgröße  $u(t)$  ohne Kenntnis der Ausgangsgröße  $y(t)$  aus der Führungsgröße  $w(t)$ . Dafür ist ein Modell des zu steuernden Prozesses notwendig. Bei dieser Art der Steuerung wird das gesteuerte System als Steuerstrecke (kurz: Strecke) und das steuernde System als Steuereinrichtung (oder Steuerung) bezeichnet.
- **Steuerung im geschlossenen Wirkungskreis (Regelung).** Bei der unten gezeigten Rückkopplungsstruktur bestimmt die Steuereinrichtung die Stellgröße  $u(t)$  in Abhängigkeit von der Vorgabe  $w(t)$  und von der Ausgangsgröße  $y(t)$ . In dieser Struktur bezeichnet man die Steuereinrichtung als Regler und das zu steuernde System als Regelstrecke. Bei diskreten Systemen sind auch in dieser Struktur die Begriffe Steuereinrichtung und Steuerstrecke ge­läufig, bei deren Verwendung man allerdings nicht die Informationsrückkopplung erkennt.

Beide Steuerungsstrukturen haben Vor- und Nachteile, wie der folgende Vergleich zeigt:

- Steuerungen in der offenen Wirkungskette sind i. Allg. schneller, weil die Steuereinrichtung mit einem dynamischen Modell der Strecke die zu verwendende Stellgröße für den gesamten Zeithorizont vorausberechnet. Dabei kann beispielsweise eine zeitoptimale Umsteuerung von einem Arbeitspunkt in einen anderen erreicht werden, die den Bereich der möglichen Stellamplituden vollständig ausnutzt. Voraussetzung für die Anwendung der Steuerung in der offenen Wirkungskette ist allerdings, dass der aktuelle Arbeitspunkt genau bekannt ist, dass das für die Berechnung der Steuerung eingesetzte Modell das Verhalten der Steuerstrecke sehr genau beschreibt und dass keine Störungen auf die Steuerstrecke einwirken.

- Steuerungen im geschlossenen Wirkungskreis berücksichtigen das tatsächliche Verhalten der Steuerstrecke, das sich in der Messgröße  $y$  niederschlägt. Diese Steuerungen erfüllen deshalb ihren Zweck auch bei nicht genau bekanntem Anfangszustand der Strecke, bei ungenauem Streckenmodell und unter der Wirkung von Störungen.

Die genannten Vor- und Nachteile der beiden Steuerungsstrukturen gelten für kontinuierliche und wertdiskrete Systeme gleichermaßen, wie die beiden folgenden Beispiele zeigen.

#### Beispiel 2.10 Aufrichten eines invertierten Pendels

Das in Abb. 2.22 gezeigte invertierte Pendel soll aus der Anfangslage  $\phi(0) = 30^\circ$  durch eine geeignet gewählte Eingangsgröße  $u(t)$  so aufgerichtet werden, dass es im oberen Gleichgewichtspunkt zur Ruhe kommt ( $w(t) = 0$ ). Regelgröße ist der Pendelwinkel  $\phi(t)$  (gemessen im Bogenmaß), Stellgröße die auf das Pendel wirkende Kraft  $u(t)$  (in Newton). Für die Stellgröße gilt die Beschränkung

$$|u(t)| \leq \bar{u} = 19,3 \text{ N.}$$

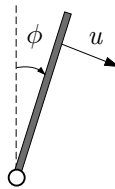


Abb. 2.22: Invertiertes Pendel

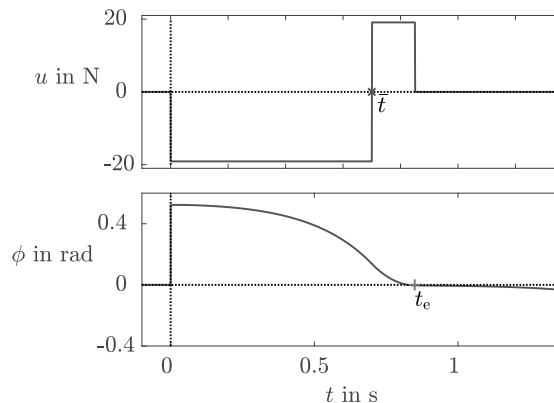
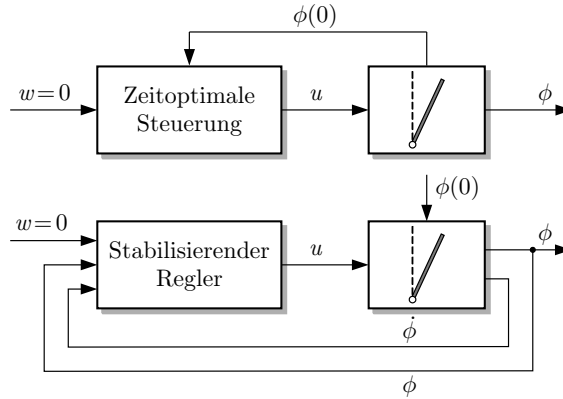


Abb. 2.23: Zeitoptimale Steuerung des Pendels

Es wird zunächst eine zeitoptimale Steuerung betrachtet, bei der die Stellgröße  $u(t)$  so bestimmt wird, dass das Pendel in der kürzestmöglichen Zeit  $t_e$  in die aufrechte Position gelangt und dort zur

Ruhe kommt ( $\phi(t_e) = 0, \dot{\phi}(t_e) = 0$ ). Diese Steuerung nutzt die maximale Stellamplitude  $\bar{u}$  aus und zwar so, dass das Pendel zunächst mit maximaler Kraft in Richtung zur oberen Position beschleunigt und anschließend mit maximaler Kraft gebremst wird (Abb. 2.23). Dabei schaltet die Steuerung zur Zeit  $\bar{t} = 0,7\text{ s}$  um und bringt das Pendel zur Zeit  $t_e = 0,84\text{ s}$  in die gewünschte Endlage.

Diese Steuerung arbeitet in der offenen Wirkungskette, denn der Verlauf der Stellgröße  $u(t)$  wird als Funktion der Zeit  $t$  berechnet und angewandt (Abb. 2.24 oben). Wie die Abbildung zeigt, muss für die Berechnung der Steuerung nicht nur der Sollwert  $w$ , sondern auch der Pendelwinkel  $\phi(0)$  bekannt sein.



**Abb. 2.24:** Steuerung des Pendels in der offenen Wirkungskette und im Regelkreis

Wie empfindlich die Steuerung in der offenen Wirkungskette bezüglich des Anfangswinkels  $\phi(0)$  ist, wird aus Abb. 2.25 offensichtlich. Es wird derselbe Verlauf der Stellgröße  $u(t)$  verwendet, jedoch mit den Anfangsauslenkungen  $\phi(0) = 29,85^\circ$  bzw.  $\phi(0) = 27^\circ$  gearbeitet, für die der Messfehler der Anfangsauslenkung also nur 0,5% bzw. 10% beträgt. In beiden Fällen fällt das Pendel um. Dieses Verhalten ist dadurch zu erklären, dass eine Steuerung in der offenen Wirkungskette exakt an die aktuellen Verhältnisse in der Steuerstrecke angepasst sein muss. Da das Pendel im oberen Gleichgewichtspunkt instabil ist, d. h. umfällt, wenn es nicht exakt in die durch  $\phi(t_e) = 0, \dot{\phi}(t_e) = 0$  beschriebene Lage gebracht wird, muss für eine Steuerung in der offenen Wirkungskette auch die Anfangsauslenkung  $\phi(0)$  präzise bekannt sein.

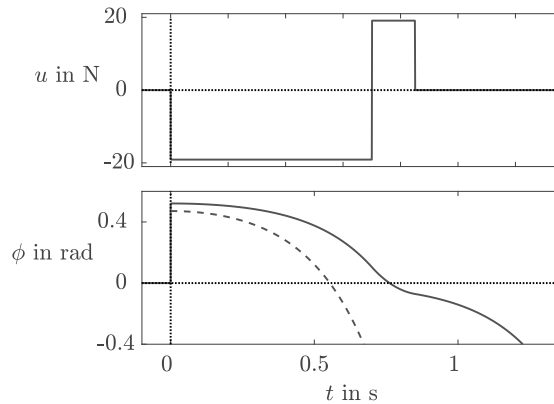
Im Gegensatz dazu kann das Pendel durch eine Regelung stabilisiert werden (Abb. 2.24 unten). Der hier verwendete Regler misst den Winkel  $\phi(t)$  und die Winkelgeschwindigkeit  $\dot{\phi}(t)$  und berechnet die Stellgröße zum Zeitpunkt  $t$  aus diesen gemessenen Größen entsprechend dem Reglergesetz

$$u(t) = -36,47\phi(t) - 3,7\dot{\phi}(t). \quad (2.9)$$

Die Parameter des Reglers wurden so gewählt, dass der Regler die angegebene Stellgrößenbeschränkung ausnutzt, aber nicht überschreitet und dass das Pendel möglichst schnell in die obere Ruhelage überführt wird.

Der linke Teil von Abb. 2.26 zeigt, dass sich das geregelte Pendel wesentlich langsamer bewegt als bei der zeitoptimalen Steuerung, die zum Vergleich durch die gestrichelte Kurve eingetragen ist. Die Stellgröße nutzt nur am Anfang den verfügbaren Stellbereich aus und nähert sich asymptotisch dem Wert null. Der Pendelwinkel verschwindet ebenfalls asymptotisch. Dass die Pendelbewegung wesentlich langsamer ist als bei der Steuerung in der offenen Wirkungskette, liegt an der Rückführstruktur.

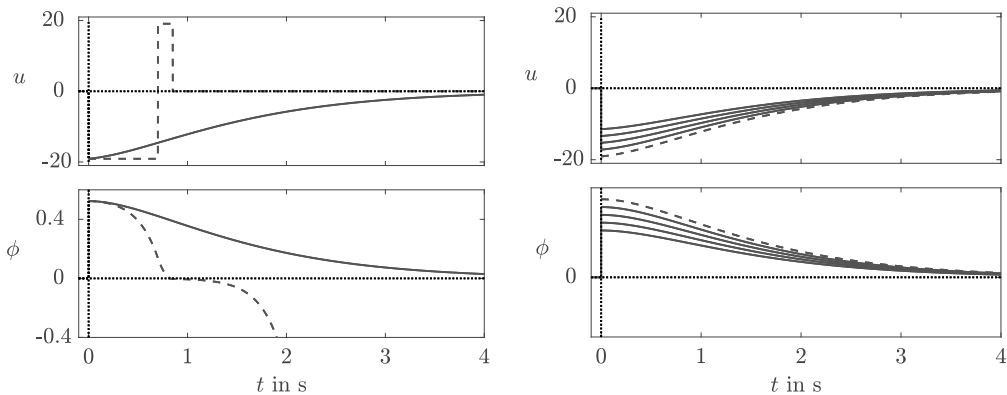




**Abb. 2.25:** Anwendung der zeitoptimalen Steuerung bei veränderter Anfangslage ( $\phi(0) = 29,85^\circ$  — bzw.  $\phi(0) = 27^\circ$  - -)

Der Regler (2.9) berechnet die Stellgröße aus einer Linearkombination der Messgrößen. Je kleiner der Winkel  $\phi(t)$  und die Winkelgeschwindigkeit  $\dot{\phi}(t)$  sind, umso kleiner ist deshalb die Stellgröße  $u(t)$  und umso langsamer bewegt sich das Pendel.

Der Vorteil der Regelung liegt in der Unempfindlichkeit gegenüber Störungen und Änderungen der Anfangsbedingungen. Abbildung 2.26 (rechts) lässt erkennen, dass dieselbe Regelung das Pendel auch dann stabilisiert, wenn der Winkel  $\phi(0)$  um 10%, 20%, 30% bzw. 40% verändert wird. Die gestrichelten Kurven beschreiben das im linken Teil der Abbildung dargestellte Verhalten des Regelkreises. Tatsächlich kann man beweisen, dass das (linearisierte) Pendel für beliebige Anfangsauslenkungen durch die Regelung in die obere Ruhelage überführt und dort stabil gehalten wird (s. Aufg. 6.6).  $\square$



**Abb. 2.26:** Verhalten des geregelten Pendels ( $u$  in N,  $\phi$  in rad)

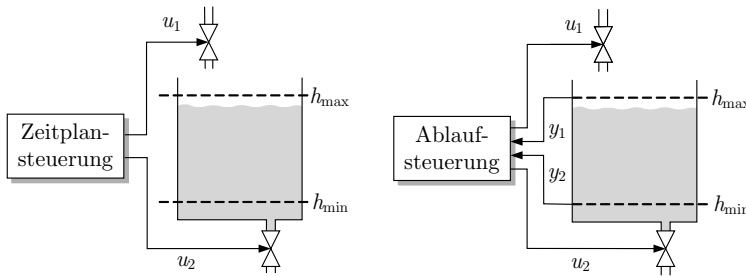
### Beispiel 2.11 Vergleich von Zeitplansteuerung und Ablaufsteuerung eines Behälters

Die in Abb. 2.21 gezeigten Steuerungsstrukturen treten auch bei der Automatisierung diskreter Systeme auf. So zeigt Abb. 2.27 auf der linken Seite eine Zeitplansteuerung, bei der die Ventile in vorgegebenen Zeitintervallen geöffnet und geschlossen werden, um den Behälter zyklisch zu füllen und wieder

zu entleeren. Da sinnvollerweise stets nur eines der beiden Ventile geöffnet ist, kann die gemeinsame Eingangsgröße beider Ventile mit  $u(t)$  bezeichnet werden, wobei  $u = 1$  für geöffnetes Ventil 1 ( $u_1 = 1$ ) und geschlossenes Ventil 2 ( $u_2 = 0$ ) steht und  $u = -1$  für den entgegengesetzten Fall. Die Zeitplansteuerung erzeugt die Eingangsgröße

$$u(t) = \begin{cases} 1 & \text{für } t_{2k} < t \leq t_{2k+1} \\ -1 & \text{für } t_{2k+1} < t \leq t_{2k+2}, \end{cases} \quad k = 0, 1, \dots$$

Wenn die Zeitpunkte  $t_k$ , ( $k = 0, 1, \dots$ ) genau auf die Fließgeschwindigkeiten der Flüssigkeit in bzw. aus dem Behälter eingestellt sind, kann der Füllstand im Behälter mit der angegebenen Steuerung zwischen den beiden Grenzwerten  $h_{\max}$  und  $h_{\min}$  hin und her bewegt werden. Es handelt sich um eine Steuerung in der offenen Wirkungskette, denn in das angegebene Steuerungsgesetz geht der aktuelle Füllstand  $h(t)$  nicht ein.



**Abb. 2.27:** Zeitplansteuerung und Ablaufsteuerung eines Behälters

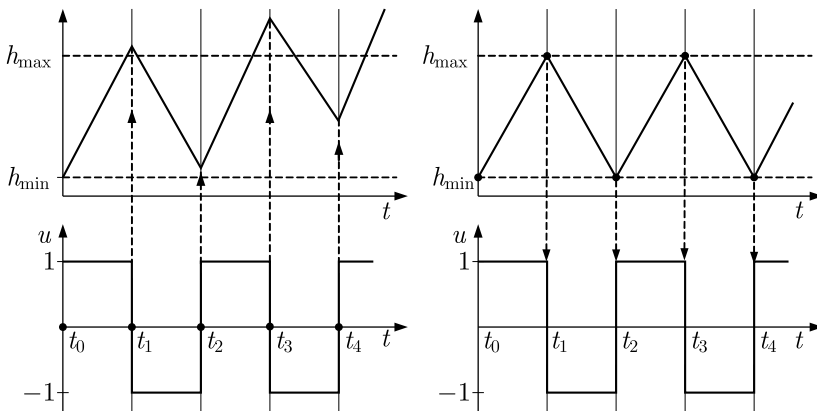
Man kann denselben Prozess durch die im rechten Teil der Abbildung gezeigte Ablaufsteuerung erzeugen, die die Ventile immer dann umschaltet, wenn der Füllstand entweder den oberen oder den unteren Grenzwert erreicht. Dafür werden die Zeiten  $t_k$  des oben angegebenen Steuerungsgesetzes nicht mit einem Streckenmodell vorausberechnet, sondern während des Prozesses so bestimmt, dass

$$h(t_{2k}) = h_{\min}, \quad h(t_{2k+1}) = h_{\max}$$

gilt. Dadurch entsteht eine Steuerung im geschlossenen Wirkungskreis (Regelung). Im störungsfreien Betrieb verhalten sich beide Systeme gleichartig.

Bei der Zeitplansteuerung geht der Informationsfluss von der Steuerung zur Steuerstrecke und nicht umgekehrt. Die Zeitpunkte  $t_0$  bis  $t_4$  im linken Teil von Abb. 2.28 sind durch die Steuerung festgelegt, was durch die gestrichelten Pfeile vom  $u$ -Diagramm zum  $h$ -Diagramm dargestellt ist. Deshalb verfehlt diese Steuerung ihr Ziel, wenn sich etwas an dem gesteuerten Prozess ändert, sei es, dass sich die Ventile nicht mehr vollständig öffnen, sich der Druck vor dem oberen Ventil verändert oder der durch einen Rechner realisierte Steuerungsalgorithmus durch andere Aufgaben des Rechners unterbrochen wird und die Kommandos an die Ventile deshalb nicht exakt zu den vorgegebenen Zeitpunkten gesendet werden.

Demgegenüber reagiert die Ablaufsteuerung auf das tatsächliche Verhalten, das im Verlauf der Füllhöhe erkennbar wird. Für die Realisierung des gewünschten Prozessverlaufes ist es ausreichend, wenn nur die beiden Grenzwerte des Füllstands überwacht werden. Die Zeitpunkte  $t_0$  bis  $t_4$  im rechten Teil von Abb. 2.28 sind diejenigen Zeitpunkte, an denen der Füllstand einen der beiden Grenzwerte erreicht. Sie werden also durch das Verhalten der Strecke festgelegt. Deshalb erzeugt die Steuerung im geschlossenen Wirkungskreis den gewünschten Prozess auch unter der Wirkung der o. g. Störungen.  $\square$

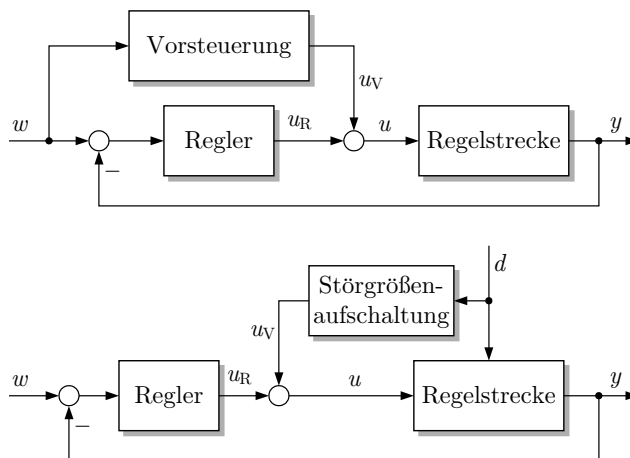


**Abb. 2.28:** Systemverhalten bei Zeitplansteuerung (links) und Regelung (rechts)

**Kombination von Vorsteuerung und Regelung.** Steuerungen in der offenen Wirkungskette und im geschlossenen Wirkungskreis treten häufig in Kombination auf, weil man die Vorteile beider Steuerungsstrukturen miteinander kombinieren will. Für die Stellgröße kontinuierlicher Systeme gilt dann

$$u(t) = u_V(t) + u_R(t), \quad (2.10)$$

wobei  $u_V$  den in der offenen Wirkungskette bestimmten Anteil und  $u_R$  den durch eine rückgeführte Steuerung bestimmten Anteil beschreibt.



**Abb. 2.29:** Vorsteuerung und Störgrößenaufschaltung

Wenn bei einer kontinuierlichen Strecke die Größe  $u_V$  in der offenen Wirkungskette aus der Führungsgröße  $w$  bestimmt wird, spricht man von einer *Vorsteuerung* (Abb. 2.29 oben). Durch eine Vorsteuerung will man erreichen, dass die Strecke möglichst schnell auf einer vorgegebenen Trajektorie  $y(t)$  bzw. in einen Endwert  $y(t_e)$  geführt wird. Dabei wird die Größe  $u_V$  mit einem dynamischen Modell der Steuerstrecke unter der Annahme festgelegt, dass es keine Störungen gibt.

Die der Vorsteuerung überlagerte Regelung hat die Aufgabe, die Stellgröße  $u(t)$  so zu verändern, dass die Solltrajektorie trotz des Wirkens von Störungen und des Vorhandenseins von Modellunbestimmtheiten eingehalten wird. Durch die Kombination von Vorsteuerung und Regelung kann man also die Vorzüge beider Steuerungsstrukturen miteinander kombinieren: Das System reagiert schnell auf Sollwertänderungen, kompensiert Störungen und ist robust gegenüber Modellunbestimmtheiten.

Bei diskreten Systemen besteht die Vorsteuerung z. B. aus einer Zeitplansteuerung, deren Stelleingriffe durch einen Regler in Abhängigkeit vom tatsächlichen Streckenverhalten korrigiert werden. Dort wird die Stellgröße  $u(t)$  auch von beiden Steuerungskomponenten beeinflusst, obwohl sie nicht als Summe (2.10) darstellbar ist.

Bei messbarer Störung  $d$  kann man die Größe  $u_V$  so auswählen, dass die Stellgröße der Störung entgegenwirkt und die Störung folglich ihre vollständige Wirkung auf die Regelgröße  $y$  gar nicht entfalten kann. Man spricht dann von einer *Störgrößenaufschaltung* (Abb. 2.29 unten). Ein typisches Beispiel für Störgrößenaufschaltungen betrifft Systeme, deren Verhalten wesentlich durch eine sich verändernde Umgebungstemperatur gestört wird. Da man die Umgebungstemperatur i. Allg. gut messen kann, kann man ihre Wirkung durch eine Störgrößenaufschaltung mindern. Die Störgrößenaufschaltung bildet mit dem Regelkreis eine Reihenschaltung. Sie ist also eine Steuerung in der offenen Wirkungskette.

### Beispiel 2.12 Steuerung eines elastischen Roboters

Der Greifer eines elastischen Roboters soll zu einer Sollposition  $y(t_e) = y_{\text{Soll}}$  bewegt werden. Dafür muss die Steuerung eine geeignete Solltrajektorie  $w_{[0,t_e]}$  festlegen und diese durch Vorgabe einer geeigneten Stellgröße  $u_{[0,t_e]}$  realisieren.

Die dafür verwendete Steuerung hat den in Abb. 2.30 gezeigten Aufbau. Im Block „Bahnplanung“ wird ein geeigneter Weg des Greifers von der aktuellen zur Sollposition ausgewählt. Dies ist eine Planungsaufgabe im Sinne von Abschn. 1.4.3. Das Ergebnis ist die Führungsgröße  $w_{[0,t_e]}$ .

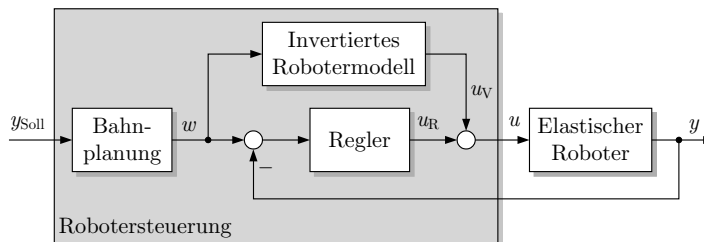


Abb. 2.30: Steuerung eines elastischen Roboters

Die Anpassung der Istgröße  $y$  an die Führungsgröße  $w$  erfolgt durch eine Vorsteuerung und eine Regelung. Die Größe  $u_V$  wird durch Inversion des Robotermodells berechnet. Das heißt, man stellt das Modell, mit dem man typischerweise für eine gegebene Eingangsgröße  $u$  die Greiferposition  $y$  bestimmen kann, so um, dass man umgekehrt aus der vorgegebenen Trajektorie  $y_{[0,t_e]}$  die Stellgröße  $u_{[0,t_e]}$  berechnen kann. Dies ist eine Vorsteuerung, weil der Verlauf der Stellgröße  $u_V$  als Funktion der Zeit bestimmt wird und nicht von der tatsächlichen Bewegung des Roboters abhängt.

Der Roboter wird sich jedoch nur dann genau auf der Trajektorie  $w_{[0,t_e]}$  bewegen, wenn das Modell des Roboters exakt ist und auf den Roboter keine äußeren Störungen einwirken. Dies ist bei elastischen Robotern schon deswegen nicht der Fall, weil man aus Aufwandsgründen für die Vorsteuerung ein Modell verwendet, das die elastischen Eigenschaften nur näherungsweise beschreibt. Man kombiniert deshalb eine Vorsteuerung mit einer Regelung, die den Fehler zwischen der gewünschten Funktion  $w(t)$  und der durch die Stellgröße  $u_V(t)$  erreichten Funktion  $y(t)$  beseitigt.

Bei Robotersteuerungen in der gezeigten Struktur ist der Anteil  $u_V$  der Stellgröße, der durch die Vorsteuerung bestimmt wird, in vielen Anwendungen sehr groß, während der vom Regler bestimmte Anteil  $u_R$  nur eine vergleichsweise kleine Amplitude hat. Diese Tatsache darf jedoch nicht darüber hinwegtäuschen, dass ohne die Korrektur der Stellgröße durch den Regler der Roboter nicht mit der erforderlichen Präzision arbeiten könnte. Die Rückkopplungsstruktur, in der die Stellgröße  $u(t)$  von der aktuellen Bewegung  $y(t)$  des Roboters abhängig ist, ist auch hier der Schlüssel zum Erfolg.  $\square$

#### **Aufgabe 2.5\*** Steuerung einer Abfüllanlage

Eine Abfüllanlage hat die Aufgabe, Behälter bis zu einer vorgegebenen Höhe mit einer Flüssigkeit zu füllen.

1. Diese Aufgabe wird heute meist durch eine Steuerung in der offenen Wirkungskette gelöst, die das Abfüllventil für eine bestimmte Zeit öffnet. Zeichnen Sie die dazugehörige Steuerkette entsprechend Abb. 1.18. Inwiefern muss bei der Festlegung der Steuerung ein Modell der zu füllenden Behälter verwendet werden? Überlegen Sie, was passiert, wenn die Steuerstrecke dadurch verändert wird, dass andersartige Behälter zum Füllen bereitgestellt werden. Was sind denkbare Störungen, auf die die Steuerung nicht reagieren kann?
2. Ergänzen Sie die Abfüllanlage so, dass der Füllvorgang durch eine Regelung beeinflusst wird. Was passiert im Regelkreis, wenn Behälter ausgetauscht werden oder Störungen wirken?  $\square$

#### **Aufgabe 2.6\*** Blockschaltbild einer Raumtemperaturregelung

Stellen Sie das Blockschaltbild für die in Abb. 1.5 auf S. 12 gezeigte Raumtemperaturregelung auf und erklären Sie die physikalische Bedeutung aller auftretenden Signale. Handelt es sich um eine Steuerung in der offenen Wirkungskette oder im geschlossenen Wirkungskreis?

Wie müssen Sie das Blockschaltbild erweitern, wenn Sie auch die Temperatur des Raumes 2 als Ausgangsgröße betrachten? Wird dieser Raum gesteuert oder geregelt?

Wie verändert sich das Blockschaltbild, wenn die Raumtemperaturen beider Räume einzeln durch Thermostatventile an den Heizkörpern geregelt werden und außerdem am Kessel eine Vorlauftemperaturregelung installiert wird?  $\square$

## Literaturhinweise

Zerlegungsmethoden für gekoppelte Systeme wurden vor allem im Zusammenhang mit komplexen Steuerungssystemen entwickelt [68]. Sie führen auf graphentheoretische Probleme, für die es eine ausführliche Literatur gibt, z. B. [26, 91, 103]. Insbesondere können die in dieser Literatur beschriebenen Suchalgorithmen eingesetzt werden, um die Mengen stark zusammenhängender Knoten zu finden und damit die Zerlegung gekoppelter Systeme in stark zusammenhängende Teilsysteme vorzunehmen.

## Teil 2: Automatisierung kontinuierlicher Systeme





# 3

## Beschreibung kontinuierlicher Systeme

*Kontinuierliche Systeme werden durch Zustandsraummodelle beschrieben, die in diesem Kapitel für lineare und nichtlineare Systeme eingeführt werden. Ziel der Linearisierung ist es, nichtlineare Systeme in der Umgebung eines Arbeitspunkts näherungsweise durch ein lineares Modell darzustellen. Bei der kompositionalen Modellbildung entsteht das Zustandsraummodell des Gesamtsystems durch Verkopplung der Teilsystemmodelle.*

### 3.1 Modellbildungsaufgabe

Dieses und die nächsten Kapitel beschäftigen sich mit kontinuierlichen Systemen. Entsprechend Abb. 3.1 haben diese Systeme die folgenden wichtigen Signale

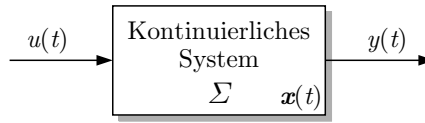
$u$  – Eingangsgröße

$x$  – Zustand

$y$  – Ausgangsgröße,

die alle reellwertig sind.

Es wird jetzt die Aufgabe betrachtet, ein mathematisches Modell aufzustellen, mit dem aus dem Verlauf  $u_{[0,t_e]}$  der Eingangsgröße der Verlauf  $y_{[0,t_e]}$  der Ausgangsgröße berechnet werden kann. Alle dafür notwendigen Schritte gehören zur Modellbildung (oder Modellierung) kontinuierlicher Systeme.



**Abb. 3.1:** Blockschaltbild eines kontinuierlichen Systems

### Modellbildungsaufgabe für kontinuierliche Systeme

Gegeben: Dynamisches System  $\Sigma$   
mit Eingangsgröße  $u$  und Ausgangsgröße  $y$   
Gesucht: Differentialgleichung oder Zustandsraummodell

Dieses Kapitel hat zwei Ziele. Erstens führt es mit der Differentialgleichung und dem Zustandsraummodell zwei Modellformen ein, die im Folgenden für alle in diesem Buch behandelten Automatisierungsaufgaben für kontinuierliche Systeme eingesetzt werden. Es wird erläutert, wie sich die dynamischen Eigenschaften der betrachteten Systeme in diesen Modellen niederschlagen. Zweitens zeigt es den Weg, auf dem man für einen gegebenen technischen Prozess zu diesen Modellen kommt. Für viele Systeme erhält man das Modell durch eine theoretische Prozessanalyse, bei der man die im System wirkenden physikalischen Gesetze aufschreibt und die dabei erhaltenen Gleichungen zu einer Differentialgleichung oder zu einem Zustandsraummodell zusammenfasst. Dieser Weg wird für die meisten in diesem Buch beschriebenen Beispiele beschritten. Es gibt jedoch auch die Möglichkeit, die Modellparameter mit Hilfe von Experimenten zu bestimmen. Man spricht dann von experimenteller Prozessanalyse oder von Identifikation. Auf diesen Weg wird nur kurz im Zusammenhang mit der Kennwertermittlung im Abschn. 4.4 eingegangen.

## 3.2 Systembeschreibung durch lineare Differentialgleichungen

Das in diesem Abschnitt behandelte Modell hat die Form einer Differentialgleichung

$$\Sigma: \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + a_{n-2} \frac{d^{n-2} y(t)}{dt^{n-2}} + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t) = \quad (3.1)$$

$$b_q \frac{d^q u(t)}{dt^q} + b_{q-1} \frac{d^{q-1} u(t)}{dt^{q-1}} + b_{q-2} \frac{d^{q-2} u(t)}{dt^{q-2}} + \dots + b_1 \frac{du(t)}{dt} + b_0 u(t).$$

In dieser Gleichung – wie auch bei allen im Folgenden behandelten Differentialgleichungen – steht die Ausgangsgröße  $y(t)$  mit ihren Ableitungen auf der linken Seite und die Eingangsgröße  $u(t)$  mit ihren Ableitungen auf der rechten Seite. Der Grad der höchsten Ableitung von  $y(t)$  heißt  $n$ . Diese Zahl wird die *dynamische Ordnung* des Systems genannt. Man spricht also beispielsweise von einem System dritter Ordnung, wenn die Ausgangsgröße  $y(t)$  bis zur dritten Ableitung in der Differentialgleichung vorkommt. Der Grad der höchsten Ableitung der Eingangsgröße  $u(t)$  wird mit  $q$  bezeichnet.  $q$  ist stets kleiner oder höchstens gleich  $n$ , wenn die Differentialgleichung (wie im Folgenden immer) ein technisch reales System beschreibt:

für technisch realisierbare Systeme :  $q \leq n$ .

Die Koeffizienten vor der Ausgangsgröße  $y(t)$  und deren Ableitungen heißen  $a_0, a_1, \dots, a_{n-1}$ , die vor der Eingangsgröße  $u(t)$  und den zugehörigen Ableitungen  $b_0, b_1, \dots, b_q$ , wobei der Index mit der Ordnung der betreffenden Ableitung übereinstimmt. Jede Differentialgleichung kann so umgeformt werden, dass der Koeffizient vor der höchsten Ableitung von  $y(t)$  gleich eins ist. Andernfalls heißt dieser Koeffizient  $a_n$ .

Man beachte, dass es auf beiden Seiten der Differentialgleichung nur Terme mit  $y(t)$  und  $u(t)$ , aber keine Konstanten gibt. Der Grund liegt darin, dass das Modell linear sein soll, wobei sich die Eigenschaft der Linearität auf den Zusammenhang zwischen dem Eingang  $u$  und dem Ausgang  $y$  bezieht. Darauf wird auf S. 111 noch genauer eingegangen.

Die erste Ableitung nach der Zeit wird häufig durch einen Punkt über der abgeleiteten Größe, die zweite Ableitung durch zwei Punkte gekennzeichnet:

$$\dot{y}(t) = \frac{dy(t)}{dt}, \quad \ddot{y}(t) = \frac{d^2y(t)}{dt^2} \quad \text{sowie} \quad \dot{u}(t) = \frac{du(t)}{dt}, \quad \ddot{u}(t) = \frac{d^2u(t)}{dt^2}.$$

Um die Differentialgleichung (3.1) lösen zu können, muss man außer dem Verlauf der Eingangsgröße  $u(t)$  für  $t \geq 0$  auch  $n$  Anfangsbedingungen für die Ausgangsgröße  $y(t)$  kennen, die die Größe  $y$  sowie deren Ableitungen bis zum Grad  $n - 1$  zur Zeit  $t = 0$  festlegen. Sie werden in der Form

$$\begin{aligned} y(0) &= y_0 \\ \frac{dy}{dt}(0) &= y_1 \\ \frac{d^2y}{dt^2}(0) &= y_2 \\ &\vdots \\ \frac{d^{n-1}y}{dt^{n-1}}(0) &= y_{n-1} \end{aligned} \tag{3.2}$$

geschrieben, wobei  $y_0, y_1, \dots, y_{n-1}$  die  $n$  bekannten Anfangswerte bezeichnen.

Wenn angegeben werden soll, welches System durch ein Modell dargestellt werden soll, wird links neben die Modellgleichungen der Systemname mit einem Doppelpunkt geschrieben. Gleichung (3.1) repräsentiert also das System  $\Sigma$ . Für Anwendungsbeispiele kann dort auch eine Bezeichnung wie „Fahrzeug“ oder „Schwingkreis“ stehen. Wenn ein System durch unterschiedliche Modelle repräsentiert wird, erscheint bei allen Modellen links derselbe Systemname, während sich die rechts daneben stehenden Gleichungen voneinander unterscheiden. Man sagt dann, dass diese Gleichungen unterschiedliche *Repräsentationen* des Systems sind.

Sämtliche in diesem Buch betrachteten Systeme haben konstante Parameter. Das heißt, dass die in der Differentialgleichung (3.1) vorkommenden Koeffizienten  $a_i$ , ( $i = 0, 1, \dots, n$ ) und  $b_j$ , ( $j = 0, 1, \dots, q$ ) nicht von der Zeit abhängen. Was von der Zeit abhängt, sind die Eingangsgröße  $u(t)$  und die Ausgangsgröße  $y(t)$ . Es gibt also einen klaren Unterschied zwischen den Parametern, die die über die Zeit unveränderlichen dynamischen Eigenschaften des Systems repräsentieren, und den Signalen, die die sich zeitlich ändernde Bewegung des Systems darstellen.

**Modellbildungsschritte.** Der Weg, auf dem man die Differentialgleichung (3.1) durch eine theoretische Prozessanalyse aufstellt, gliedert sich in die folgenden vier Schritte:

---

**Algorithmus 3.1** *Aufstellung einer Differentialgleichung*

---

**Gegeben:** Kontinuierliches System mit Eingang  $u(t)$  und Ausgang  $y(t)$ .

1. **Systemzerlegung:** Zerlegen Sie das System in Komponenten.
2. **Beschreibung der Komponenten:** Schreiben Sie die physikalischen Gesetze auf, die das Verhalten der Komponenten bestimmen.
3. **Beschreibung der Kopplungen:** Beschreiben Sie die Beziehungen, die zwischen den Komponenten bestehen.
4. **Aufstellung der Differentialgleichung:** Fassen Sie die Gleichungen zu einer Differentialgleichung zusammen.

**Ergebnis:** Differentialgleichung der Form (3.1).

---

Beim ersten Schritt wird das System in seine elementaren Bestandteile zerlegt. Dies sind bei elektrischen Schaltungen typischerweise die Bauelemente (Widerstände, Kondensatoren, Spulen, Transistoren), bei verfahrenstechnischen Systemen Apparate und Verbindungskomponenten (Reaktoren, Rohrleitungen, Ventile, Zwischenspeicher), bei mechanischen Systemen Massen, Federn oder Dämpfer. Für diese Komponenten sind die im zweiten Schritt aufzuschreibenden Gleichungen meist Differentialgleichungen erster Ordnung oder algebraische Gleichungen. Die Beziehungen zwischen den Komponenten führen häufig auf algebraische Gleichungen, die Ströme oder Spannungen, Momente oder Kräfte zusammenfassen oder Energieflüsse bilanzieren.

In den ersten drei Modellbildungsschritten werden eine Reihe physikalischer Größen eingeführt, von denen nur die Eingangsgröße  $u(t)$  und die Ausgangsgröße  $y(t)$  in der Differentialgleichung vorkommen dürfen. Der vierte Schritt beinhaltet deshalb im Wesentlichen die Elimination aller zeitabhängigen Größen bis auf  $u(t)$  und  $y(t)$ . Das folgende Beispiel zeigt dieses Vorgehen.

**Beispiel 3.1** *Aufstellung der Differentialgleichung für einen Gleichstrommotor*

Es soll eine Differentialgleichung der Form (3.1) aufgestellt werden, die den Gleichstrommotor aus Abb. 3.2 beschreibt. Die Eingangsgröße ist die Motorspannung  $u(t)$ , die Ausgangsgröße die Drehgeschwindigkeit  $\omega(t)$ . Entsprechend dem Algorithmus 3.1 wird dabei in vier Schritten vorgegangen:

1. **Systemzerlegung:** Wie die Schaltung zeigt, lässt sich der Gleichstrommotor in vier Komponenten zerlegen: den ohmschen Widerstand  $R$ , die Induktivität  $L$  des Ankerkreises, das System Stator-Rotor sowie die sich drehende Masse. Diese vier Elemente kommen nicht als vier voneinander getrennte elektrische bzw. mechanische Bauteile vor, sondern gehen aus einer Idealisierung der elektrischen und mechanischen Vorgänge im Motor hervor. So beinhaltet die rotierende Masse

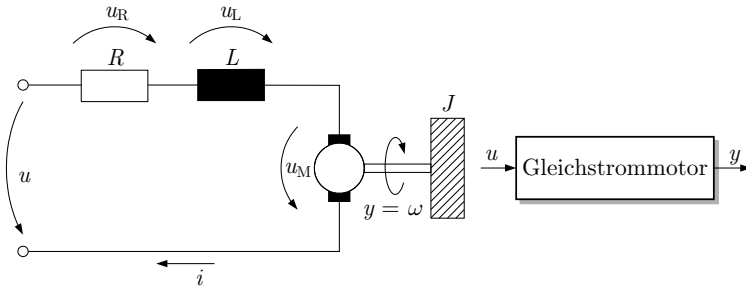


Abb. 3.2: Schaltbild und Blockschaltbild des Gleichstrommotors

nicht nur die angetriebene Last, sondern auch die Masse des Rotors, und der Widerstand  $R$  umfasst auch den ohmschen Widerstand der Anschlussleitungen und des Stators.

## 2. Beschreibung der Komponenten: Widerstand und Spule sind durch die Gleichungen

$$u_R(t) = Ri(t) \quad (3.3)$$

$$u_L(t) = L \frac{di(t)}{dt} \quad (3.4)$$

beschrieben, wobei  $u_R(t)$  und  $u_L(t)$  die Spannungen über den beiden Bauelementen,  $i(t)$  den Strom durch den Motor und  $R$  bzw.  $L$  den ohmschen Widerstand bzw. die Induktivität des Motors bezeichnen. Auf Grund des Induktionsgesetzes induziert der Rotor die Spannung  $u_M(t)$ , die proportional zu seiner Winkelgeschwindigkeit  $\omega(t)$  ist:

$$u_M(t) = k_M \omega(t). \quad (3.5)$$

In dieser Gleichung ist  $k_M$  ein motorspezifischer Parameter. Das durch den Ankerstrom  $i(t)$  erzeugte Magnetfeld beschleunigt den Rotor mit dem Drehmoment  $M(t)$ , das in Richtung der Drehgeschwindigkeit  $\omega(t)$  wirkt und für das die Beziehung

$$M(t) = k_T i(t) \quad (3.6)$$

mit einem Proportionalitätsfaktor  $k_T$  gilt. Die rotierende Masse wird durch die Reibung gebremst, die das Moment

$$M_R(t) = k_R \omega(t) \quad (3.7)$$

erzeugt, das dem Moment  $M(t)$  entgegenwirkt. In den angegebenen Gleichungen beschreiben  $k_M$ ,  $k_T$  und  $k_R$  drei Parameter, die vom Aufbau des Motors abhängen. Das auf den Rotor wirkende Moment  $M_B(t)$  verändert die Winkelgeschwindigkeit entsprechend der Beziehung

$$J\dot{\omega}(t) = M_B(t), \quad (3.8)$$

wobei  $J$  das Trägheitsmoment der rotierenden Masse bezeichnet.

## 3. Beschreibung der Kopplungen: Die Kopplung zwischen den elektrischen Komponenten ist durch den Maschensatz beschrieben, demzufolge die Summe aller Spannungen gleich null ist:

$$u_R(t) + u_L(t) + u_M(t) - u(t) = 0. \quad (3.9)$$

Die Vorzeichen der Summanden sind durch die Pfeilrichtung der Spannungen in Abb. 3.2 festgelegt. Für die mechanischen Größen gilt, dass die Summe aller Momente gleich null ist

$$M(t) - M_R(t) - M_B(t) = 0, \quad (3.10)$$

wobei sich die Vorzeichen der Momente daraus ergeben, dass  $M(t)$  in Richtung der Winkelgeschwindigkeit  $\omega(t)$  wirkt, während  $M_R(t)$  und  $M_B(t)$  der Drehrichtung entgegenwirken.

Als Ergebnis der Modellbildungsschritte 1 – 3 liegt ein Gleichungssystem vor, das den Gleichstrommotor vollständig beschreibt. Dieses Modell hat allerdings nicht die gewünschte Form. Im 4. Modellbildungsschritt wird dieses Gleichungssystem in eine Differentialgleichung umgeformt.

4. **Aufstellung der Differentialgleichung:** Das gesuchte Modell ist eine Differentialgleichung, in der von den jetzt eingeführten Signalen nur noch die Signale  $u(t)$  und  $\omega(t)$  vorkommen. Alle anderen Signale müssen durch eine geeignete Kombination der aufgestellten Gleichungen eliminiert werden. Dabei kann man wie folgt vorgehen.

Man führt zunächst die Gln. (3.3) – (3.5) und (3.9) zusammen und erhält

$$Ri(t) + L \frac{di(t)}{dt} + k_M \omega(t) = u(t). \quad (3.11)$$

Der Strom  $i(t)$  lässt sich entsprechend Gl. (3.6) durch das Moment  $M(t)$  ausdrücken, wodurch man auf die Gleichung

$$\frac{R}{k_T} M(t) + \frac{L}{k_T} \dot{M}(t) + k_M \omega(t) = u(t) \quad (3.12)$$

kommt. Andererseits erhält man aus den Gln. (3.7), (3.8) und (3.10) die Beziehungen

$$\begin{aligned} M(t) &= J\dot{\omega}(t) + k_R \omega(t) \\ \dot{M}(t) &= J\ddot{\omega}(t) + k_R \dot{\omega}(t), \end{aligned} \quad (3.13)$$

mit denen man  $M(t)$  und  $\dot{M}(t)$  aus Gl. (3.12) eliminieren kann:

$$\begin{aligned} u(t) &= \frac{R}{k_T} (J\dot{\omega}(t) + k_R \omega(t)) + \frac{L}{k_T} (J\ddot{\omega}(t) + k_R \dot{\omega}(t)) + k_M \omega(t) \\ &= \frac{JL}{k_T} \ddot{\omega}(t) + \left( \frac{RJ}{k_T} + \frac{Lk_R}{k_T} \right) \dot{\omega}(t) + \left( \frac{Rk_R}{k_T} + k_M \right) \omega(t). \end{aligned}$$

Dividiert man diese Gleichung noch durch den vor  $\ddot{\omega}(t)$  stehenden Faktor  $\frac{JL}{k_T}$ , so erhält man mit

$$\text{Gleichstrommotor: } \ddot{\omega}(t) + \left( \frac{R}{L} + \frac{k_R}{J} \right) \dot{\omega}(t) + \left( \frac{Rk_R}{JL} + \frac{k_M k_T}{JL} \right) \omega(t) = \frac{k_T}{JL} u(t) \quad (3.14)$$

die gesuchte Differentialgleichung, die die Form (3.1) hat. Als Anfangsbedingung braucht man Werte für  $\omega(0)$  und  $\dot{\omega}(0)$ , die mit  $\omega_0$  und  $\omega_1$  bezeichnet werden:

$$\omega(0) = \omega_0 \quad \text{und} \quad \dot{\omega}(0) = \omega_1.$$

Offensichtlich ist der Motor ein System zweiter Ordnung ( $n = 2$ ). In der Differentialgleichung tritt auf der rechten Seite nur die Eingangsgröße  $u(t)$ , nicht jedoch deren Ableitung auf ( $q = 0$ ).

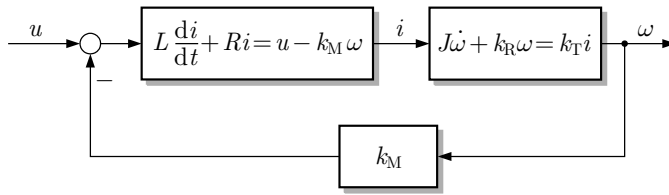


Abb. 3.3: Detailliertes Blockschaltbild des Gleichstrommotors

**Diskussion.** Aus den während der Modellbildung aufgestellten Gleichungen kann man das in Abb. 3.3 gezeigte Blockschaltbild aufstellen, das die im Motor ablaufenden Prozesse gut strukturiert beschreibt. Der linke Block enthält den durch Gl. (3.11) dargestellten Prozess, bei dem sich nach Anlegen einer Spannung  $u(t)$  der Strom solange erhöht, bis die angegebene Gleichung erfüllt ist. Der mit  $k_T$  multiplizierte Strom ergibt das Antriebsmoment  $M(t)$  (vgl. Gl. (3.6)), das zu einer Erhöhung der Winkelgeschwindigkeit  $\omega(t)$  führt. Dieser Prozess ist durch die aus den Gln. (3.6) und (3.13) entstehende Beziehung

$$k_T i(t) = J \dot{\omega}(t) + k_R \omega(t) \quad (3.15)$$

beschrieben. Die Rückführung im Blockschaltbild gibt die Rückwirkung des mechanischen Teils des Gleichstrommotors auf den elektrischen Teil wieder. Demzufolge kann sich der Strom und mit diesem das Antriebsmoment nur so lange erhöhen, bis die durch  $k_M \omega(t)$  beschriebene Rückwirkung so groß wie die Eingangsspannung  $u(t)$  ist.

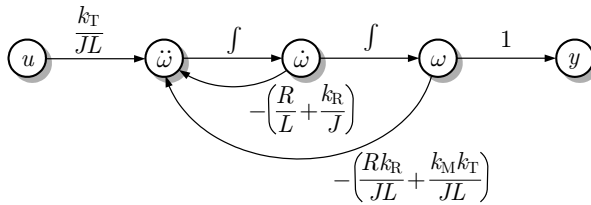


Abb. 3.4: Signalflussgraph des Gleichstrommotors

Für die Differentialgleichung (3.14) des Motors kann man einen Signalflussgraphen aufstellen (Abb. 3.4), der das Motormodell in sehr kompakter Form beschreibt, denn in ihm kommen nur noch die Eingangsspannung  $u(t)$ , die Winkelgeschwindigkeit  $\omega(t)$  zusammen mit ihren ersten beiden Ableitungen und die Ausgangsgröße  $y(t)$ , die gleich der Winkelgeschwindigkeit ist, vor. Im Unterschied zum Blockschaltbild zeigt der Signalflussgraph die Beziehungen zwischen den einzelnen Signalen und deren Ableitungen.

In das Modell gehen mit  $R$ ,  $L$ ,  $k_T$ ,  $k_R$  und  $k_M$  fünf Parameter des elektrischen Teils des Motors sowie mit  $J$  eine die mechanischen Eigenschaften des Rotors und der Last beschreibende Größe ein. Aus diesen Parametern werden die Koeffizienten der Differentialgleichung bestimmt. Bemerkenswerterweise entstehen dabei *drei* Koeffizienten

$$a_1 = \frac{R}{L} + \frac{k_R}{J}, \quad a_0 = \frac{R k_R}{J L} + \frac{k_M k_T}{J L}, \quad b_0 = \frac{k_T}{J L},$$

die aus den *sechs* Motorparametern bestimmt werden. Unterschiedliche Motoren können deshalb auf dieselben Modellparameter führen, sich also trotz ihrer unterschiedlichen physikalischen Parameter dynamisch gleichartig verhalten.  $\square$

### 3.3 Zustandsraummodell linearer Systeme

#### 3.3.1 Zustandsgleichung und Ausgabegleichung

Obwohl sich alle linearen dynamischen Systeme durch Differentialgleichungen (3.1) beschreiben lassen, wird dieses Modell im Folgenden nicht eingesetzt, sondern durch ein anderes Modell ersetzt, das aus  $n$  Differentialgleichungen erster Ordnung besteht. Mathematisch sind beide Modelle äquivalent; für den praktischen Einsatz hat das jetzt einzuführende Zustandsraummodell aber einige Vorteile. Der wichtigste Vorteil ist seine kompakte Darstellung als eine Vektordifferentialgleichung erster Ordnung, aus der u. a. eine gut überschaubare mathematische Beziehung für die Lösung dieser Modellgleichungen und eine einfache rechentechnische Handhabung folgen. Ein weiterer Vorteil besteht darin, dass der in diesem Modell vorkommende Zustandsvektor eine sehr anschauliche ingenieurtechnische Interpretation hat, die später auch eine Analogie zu den diskreten Systemen ermöglicht.

Um die Idee der Zustandsraumdarstellung zu erläutern, wird zunächst noch einmal der Gleichstrommotor betrachtet.

#### Beispiel 3.2 Zustandsraummodell eines Gleichstrommotors

Zum Zustandsraummodell des Gleichstrommotors kommt man durch die Umformung der Gln. (3.11) und (3.15), durch die man die Beziehungen

$$\begin{aligned}\frac{di(t)}{dt} &= -\frac{R}{L}i(t) - \frac{k_M}{L}\omega(t) + \frac{1}{L}u(t) \\ \frac{d\omega(t)}{dt} &= \frac{k_T}{J}i(t) - \frac{k_R}{J}\omega(t)\end{aligned}$$

erhält. Es sind dies zwei Differentialgleichungen erster Ordnung, auf deren linker Seite die beiden Ableitungen  $\frac{di(t)}{dt}$  und  $\frac{d\omega(t)}{dt}$  stehen und auf deren rechter Seite außer den beiden nicht abgeleiteten Größen  $i(t)$  und  $\omega(t)$  nur die Eingangsgröße  $u(t)$  auftritt. Man kann die beiden Gleichungen deshalb in der Matrix-Vektor-Form

$$\text{Gleichstrommotor : } \left\{ \begin{aligned} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} &= \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix} u(t) \\ \begin{pmatrix} i(0) \\ \omega(0) \end{pmatrix} &= \begin{pmatrix} i_0 \\ \omega_0 \end{pmatrix} \\ y(t) &= (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} \end{aligned} \right. \quad (3.16)$$

schreiben, zu der eine zweite Gleichung hinzugefügt wurde, mit der die Ausgangsgröße  $y(t) = \omega(t)$  berechnet wird. In der ersten Gleichung stehen der Vektor

$$\mathbf{x}(t) = \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix}$$

sowie dessen Ableitung  $\dot{\mathbf{x}}(t)$ , so dass die beiden Gleichungen auch in der Form



$$\text{Gleichstrommotor : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix} u(t), & \mathbf{x}(0) = \begin{pmatrix} i_0 \\ \omega_0 \end{pmatrix} \\ y(t) = (0 \quad 1) \mathbf{x}(t) \end{cases}$$

geschrieben werden können. Die Anfangsbedingung  $\mathbf{x}(0)$  für die Vektordifferentialgleichung enthält Werte  $i_0$  und  $\omega_0$  für den Strom und die Winkelgeschwindigkeit zur Zeit  $t = 0$ . Den Wert  $i_0$  erhält man aus den Anfangsbedingungen  $\omega_0$  und  $\omega_1$  der Differentialgleichung (3.14), wenn man für die Zeit  $t = 0$  die Gln. (3.6) – (3.8) in (3.10) einsetzt:

$$\begin{aligned} k_T i(0) - k_R \omega(0) - J \dot{\omega}(0) &= 0 \\ i_0 &= \frac{k_R \omega_0 + J \omega_1}{k_T}. \quad \square \end{aligned}$$

An Stelle der im Beispiel angegebenen zwei Differentialgleichungen erster Ordnung treten i. Allg.  $n$  Differentialgleichungen auf. Das als Vektordifferentialgleichung geschriebene Modell heißt *Zustandsraummodell*. Es hat die folgende allgemeine Form:

Zustandsraummodell linearer Systeme:

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) + d u(t). \end{cases} \quad (3.17)$$

Die erste Gleichung heißt *Zustandsgleichung*, die zweite *Ausgabegleichung*.

Die Größe  $\mathbf{x}(t)$  ist i. Allg. ein  $n$ -dimensionaler Vektor mit den zeitabhängigen Elementen  $x_i(t)$

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{pmatrix},$$

$\mathbf{A}$  eine  $(n \times n)$ -Matrix

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix},$$

die als *Systemmatrix* bezeichnet wird,  $\mathbf{b}$  ein  $n$ -dimensionaler Spaltenvektor

$$\mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

und  $\mathbf{c}^T$  ein  $n$ -dimensionaler Zeilenvektor

$$\mathbf{c}^T = (c_1 \ c_2 \ \dots \ c_n).$$

$d$  ist ein Skalar, der bei vielen technischen Systemen verschwindet. Die Elemente  $n$ ,  $\mathbf{A}$ ,  $\mathbf{b}$ ,  $\mathbf{c}^T$  und  $d$  sind nicht von der Zeit abhängig.  $\mathbf{x}_0$  ist ein  $n$ -dimensionaler Vektor, der die Werte aller Komponenten  $x_i$  von  $\mathbf{x}$  zur Zeit  $t = 0$  enthält. Bei der Anwendung des Modells (3.17) wird i. Allg. vorausgesetzt, dass  $\mathbf{x}_0$  bekannt ist.

Die Dimension  $n$  der Vektoren und der Matrix heißt *dynamische Ordnung* des Systems. Sie stimmt mit der für die Differentialgleichung (3.1) eingeführten Ordnung überein, denn ein durch eine Differentialgleichung  $n$ -ter Ordnung beschriebenes System hat ein Zustandsraummodell  $n$ -ter Ordnung.

Zusammengefasst wird ein lineares System also durch folgende Angaben bestimmt:

$\mathbb{R}$	Wertebereich der Eingangsgröße $u$
$\mathbb{R}^n$	Wertebereich des Systemzustands $\mathbf{x}$ (Zustandsraum)
$\mathbb{R}$	Wertebereich der Ausgangsgröße $y$
$\mathbf{A}, \mathbf{b}, \mathbf{c}^T, d$	Parameter des Zustandsraummodells
$\mathbf{x}_0$	Anfangszustand.

Man kann ein lineares System deshalb auch als 8-Tupel schreiben:

$$\Sigma = (\mathbb{R}^n, \mathbb{R}, \mathbb{R}, \mathbf{A}, \mathbf{b}, \mathbf{c}^T, d, \mathbf{x}_0). \quad (3.18)$$

Diese Gleichung besagt, dass das lineare System  $\Sigma$  durch die drei Signallräume  $\mathbb{R}^n$ ,  $\mathbb{R}$  und  $\mathbb{R}$  für den Zustand, den Eingang und den Ausgang, durch die Matrix  $\mathbf{A}$ , die Vektoren  $\mathbf{b}$  und  $\mathbf{c}^T$  und den Skalar  $d$  sowie durch den Anfangszustand  $\mathbf{x}_0$  festgelegt ist.

**Modellbildung.** Im Vergleich zur Modellierung kontinuierlicher Systeme durch eine Differentialgleichung ändert sich das Modellbildungsziel, das jetzt in der Aufstellung eines Zustandsraummodells besteht. Dieses Modell erhält man in den folgenden Schritten:

---

### Algorithmus 3.2 Aufstellung eines Zustandsraummodells

---

**Gegeben:** Kontinuierliches System mit Eingang  $u(t)$  und Ausgang  $y(t)$ .

1. **Systemzerlegung:** Zerlegen Sie das System in Komponenten.
2. **Beschreibung der Komponenten:** Schreiben Sie die physikalischen Gesetze auf, die das Verhalten der Komponenten bestimmen.
3. **Beschreibung der Kopplungen:** Beschreiben Sie die Beziehungen, die zwischen den Komponenten bestehen.
4. **Aufstellung des Zustandsraummodells:** Fassen Sie die Gleichungen zu einem Zustandsraummodell zusammen.

**Ergebnis:** Zustandsraummodell (3.17).

---

Das Vorgehen unterscheidet sich nur im vierten Schritt vom Algorithmus 3.1. Wie am Beispiel des Gleichstrommotors gezeigt wurde, ist die Aufstellung des Zustandsraummodells häufig einfacher als die Ableitung einer Differentialgleichung, weil die im Vektor  $\mathbf{x}$  stehenden internen

Signale im Modell verbleiben und nicht durch aufwändige Umformungen eliminiert werden müssen. Zum Schritt 4 des Algorithmus gehört die geeignete Wahl des Vektors  $\mathbf{x}$ , auf die im Folgenden eingegangen wird.

### 3.3.2 Zustandsbegriff

Ein für die Anwendung der hier behandelten Modelle sehr wichtiger Begriff ist der des Zustands eines dynamischen Systems. Um zu verdeutlichen, warum man den in Gl. (3.17) vorkommenden Vektor  $\mathbf{x}$  als Systemzustand bezeichnet, soll zunächst auf die Lösbarkeit der Zustandsgleichung

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (3.19)$$

eingegangen und dabei ein System ohne Eingangsgröße (autonomes System)

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (3.20)$$

betrachtet werden. Das Modell ist eine Vektordifferentialgleichung, auf die alle Erkenntnisse über (skalare) lineare Differentialgleichungen angewendet werden können. Diese Differentialgleichung besitzt für eine beliebige durch  $\mathbf{x}_0$  beschriebene Anfangsbedingung eine eindeutige Lösung, die jetzt über das Zeitintervall  $[0, t_e]$  betrachtet und mit  $\mathbf{x}_{[0, t_e]}$  bezeichnet wird. Die Existenz der Lösung kann folgendermaßen interpretiert werden:

|| Wenn der Anfangszustand  $\mathbf{x}_0$  bekannt ist, so kann die Zustandstrajektorie  $\mathbf{x}_{[0, t_e]}$  des autonomen Systems (3.20) für eine beliebige Intervalllänge  $t_e$  berechnet werden.

Das hat insbesondere zur Folge, dass man zur Berechnung von  $\mathbf{x}_{[0, t_e]}$  keine Informationen darüber benötigt, welche Werte der Vektor  $\mathbf{x}(t)$  für  $t < 0$  angenommen hat. In  $\mathbf{x}_0$  stecken also alle diejenigen Informationen über die Bewegung des Systems bis zum Zeitpunkt  $t = 0$ , die für die zukünftige Bewegung des Systems maßgebend sind. Das ist der Grund, weshalb man den Vektor  $\mathbf{x}(t)$  als Systemzustand bezeichnet.

Für das durch die Eingangsgröße  $u(t)$  erregte System (3.19) gelten diese Aussagen, wenn die Eingangsgröße für das betrachtete Zeitintervall  $[0, t_e]$  bekannt ist. Dass man  $u(t)$  im Zeitintervall  $[0, t_e]$  kennen muss, um die Bewegung des Systems für dieses Zeitintervall berechnen zu können, ist klar. Wichtiger ist, dass man den Verlauf der Eingangsgröße für das davor liegende Zeitintervall  $t < 0$  nicht kennen muss:

|| Der Verlauf von  $\mathbf{x}(t)$  im Zeitintervall  $[0, t_e]$  hängt für das System (3.19) bei bekanntem Anfangszustand  $\mathbf{x}(0) = \mathbf{x}_0$  nicht von Werten ab, die die Eingangsgröße  $u(t)$  für  $t < 0$  oder  $t > t_e$  annimmt.

Der Anfangszustand  $\mathbf{x}_0$  enthält also auch alle diejenigen Information über den Verlauf der Eingangsgröße für  $t < 0$ , die für die zukünftige Bewegung des Systems wichtig sind.

Diese Erkenntnisse lassen sich in folgender Zustandsdefinition zusammenfassen:

**Definition 3.1 (Zustand eines kontinuierlichen Systems)**

Ein Vektor  $\mathbf{x}(t)$  ist der Zustand eines kontinuierlichen Systems, wenn für einen beliebigen Zeithorizont  $t_e > 0$  seine Werte  $x_i(0)$ , ( $i = 1, 2, \dots, n$ ) zum Zeitpunkt  $t = 0$  zusammen mit dem Verlauf der Eingangsgröße  $u(\tau)$  für  $0 \leq \tau \leq t_e$  den Wert  $y(t_e)$  der Ausgangsgröße eindeutig bestimmen.  $\mathbf{x}(t)$  heißt auch Zustandsvektor und die Komponenten  $x_i(t)$  von  $\mathbf{x}(t)$  Zustandsvariablen oder Zustandsgrößen.

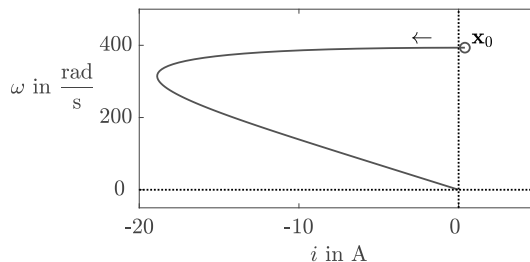
Es hängt vom Anwendungsfall ab, welche Länge  $n$  der Zustandsvektor hat und welche physikalischen Größen in diesem Zustandsvektor stehen. Die Zahl  $n$  der Zustandsvariablen stimmt häufig mit der Anzahl der Energiespeicher überein, die im Signalweg vom Eingang  $u(t)$  zum Ausgang  $y(t)$  auftreten, also bei elektrischen Systemen mit der Anzahl der Kapazitäten und Induktivitäten oder bei mechanischen Systemen mit der Anzahl der Massen und Federn. Die physikalischen Zustandsvariablen sind die den Inhalt dieser Energiespeicher beschreibenden Größen wie Ströme, Spannungen, Kräfte, Geschwindigkeiten oder Stoffkonzentrationen.

Wie später noch gezeigt wird, ist die Wahl der Zustandsvariablen nicht eindeutig. Charakteristisch für ein System ist jedoch ihre Anzahl  $n$ , also die dynamische Ordnung.

Da sich der Zustandsvektor  $\mathbf{x}(t)$  im  $n$ -dimensionalen Raum  $\mathbb{R}^n$  bewegt, nennt man den Raum  $\mathbb{R}^n$  auch den Zustandsraum. Der Verlauf von  $\mathbf{x}(t)$  für  $t \geq 0$  heißt Trajektorie des Systems.

**Beispiel 3.3** Zustandstrajektorie des Gleichstrommotors

Für den Gleichstrommotor aus Beispiel 3.2 ist eine Zustandstrajektorie in Abb. 3.5 gezeigt. Die beiden Zustandsvariablen  $i$  und  $\omega$  sind die Koordinaten des zweidimensionalen Zustandsraumes. Der Gleichstrommotor beginnt seine Bewegung bei  $\mathbf{x}_0 = (0, 393, 393, 4)^T$ , also bei einem Zustand, bei dem der Strom  $i(0) = 0,393$  A durch den Anker fließt und sich der Rotor mit einer Winkelgeschwindigkeit von  $\omega(0) = 393,4 \frac{\text{rad}}{\text{s}}$  bewegt. Der Motor wurde abgeschaltet ( $u(t) = 0, t \geq 0$ ). Wie die Abbildung zeigt, verkleinert sich die Winkelgeschwindigkeit und der Strom kehrt seine Richtung um, bis der Motor im Ursprung des Zustandsraumes zur Ruhe kommt.  $\square$



**Abb. 3.5:** Trajektorie des Gleichstrommotors im Zustandsraum

Entsprechend der Definition des Zustands geht man bei Betrachtung eines dynamischen Systems davon aus, dass sich die Wirkung der Eingangsgröße für negative Zeit  $t$  im Anfangszustand  $\mathbf{x}_0$  widerspiegelt und die Eingangsgröße  $u(t)$ , der Zustand  $\mathbf{x}(t)$  und die Ausgangsgröße  $y(t)$  nur für  $t > 0$  betrachtet werden. Deshalb wird im Folgenden stets angenommen, dass alle Signale für  $t < 0$  verschwinden:

$$\left. \begin{array}{l} u(t) = 0 \\ \mathbf{x}(t) = 0 \\ y(t) = 0 \end{array} \right\} \text{ für } t < 0.$$

Der Zeitpunkt  $t = 0$  stellt also einen „Schnitt“ auf der Zeitachse dar. Alle vorher abgelaufenen Bewegungen sind durch den Zustand  $\mathbf{x}_0$  zum Zeitpunkt  $t = 0$  erfasst.

Bei der Anwendung des Zustandsraummodells über einen langen Zeitraum kann man das betrachtete Zeitintervall in mehrere Abschnitte  $[0, t_1]$ ,  $[t_1, t_2]$ , ...,  $[t_k, t_e]$  unterteilen und den Zustand  $\mathbf{x}(t_i)$  am Ende  $t_i$  eines Zeitintervalls als Anfangszustand für das nächste Zeitintervall verwenden. Dann spielt der Verlauf der Eingangsgröße für  $t < t_i$  für die Berechnungen der Bewegung im Zeitintervall  $[t_i, t_{i+1}]$  keine Rolle mehr (vgl. Beispiel 4.1 auf S. 129).

### 3.3.3 Normierung der Signale und Parameter

Die Parameter und Signale, die im Zustandsraummodell auftreten, sind i. Allg. physikalische Größen, die durch ihren Zahlenwert und ihre physikalische Einheit beschrieben werden. Wenn man das Zustandsraummodell für Analyse- oder Entwurfsaufgaben einsetzt, schreibt man es jedoch als reine *Zahlenwertgleichungen*. Um zu diesen Gleichungen zu kommen, normiert man alle Signale und Parameter bezüglich ihrer physikalischen Einheiten.

Wenn man für alle Größen die Grundeinheiten (Sekunde, Meter, Gramm usw.) verwendet, kommt man von den Größengleichungen zu den Zahlenwertgleichungen, indem man einfach die physikalischen Einheiten weglässt. Diese Vorgehensweise ist jedoch nicht immer zweckmäßig, weil man bei Verwendung der Grundeinheiten oft sehr kleine oder sehr große Zahlen erhält. Deshalb wählt man physikalische Einheiten, die für den betrachteten Anwendungsfall zweckmäßig sind, und misst beispielsweise bei schnellen elektromagnetischen Vorgängen die Zeit in Millisekunden oder bei sehr langsamen verfahrenstechnischen Prozessen die Zeit in Minuten oder Stunden. Man ist bei der Wahl der physikalischen Einheiten für alle Signale und Parameter frei, muss jedoch darauf achten, dass sich die Zahlenwertgleichungen nach diesen Maßeinheiten richten.

Der Übergang von den Größengleichungen zu den Zahlenwertgleichungen wird dadurch vollzogen, dass man die auf die gewählte physikalische Einheit bezogenen Größen einführt und die Maßeinheiten dann aus den Gleichungen herauskürzt. Dies wird schrittweise im folgenden Beispiel vorgerechnet.

**Beispiel 3.4** Normierung des Zustandsraummodells des Gleichstrommotors

Für den Gleichstrommotor aus Beispiel 3.2 auf S. 90 werden folgende Parameterwerte eingeführt:

$$\begin{aligned} R &= 0,5 \, \Omega & k_T &= 30 \cdot 10^{-3} \frac{\text{Nm}}{\text{A}} \\ L &= 50 \, \text{mH} & k_R &= 30 \cdot 10^{-6} \, \text{Nm s} \\ k_M &= 30 \frac{\text{mVs}}{\text{rad}} & J &= 13 \cdot 10^{-4} \, \text{Nm s}^2. \end{aligned}$$

Der Strom  $i$  soll in Ampere, die Spannung  $u$  in Volt, die Winkelgeschwindigkeit  $\omega$  in  $\frac{\text{rad}}{\text{s}}$  und die Zeit  $t$  in Sekunden gemessen werden. Das bedeutet, dass im Modell an Stelle der Größen  $i$ ,  $u$ ,  $\omega$  und  $t$  die neuen Signale

$$\tilde{i} = \frac{i}{\text{A}}, \quad \tilde{u} = \frac{u}{\text{V}}, \quad \tilde{\omega} = \frac{\omega}{\frac{\text{rad}}{\text{s}}} \quad \text{und} \quad \tilde{t} = \frac{t}{\text{s}}$$

auftreten sollen. Dazu kommt man, wenn man in die Größengleichungen die Beziehungen

$$i = \tilde{i} \cdot \text{A}, \quad u = \tilde{u} \cdot \text{V}, \quad \omega = \tilde{\omega} \cdot \frac{\text{rad}}{\text{s}} \quad \text{und} \quad t = \tilde{t} \cdot \text{s}$$

einsetzt, wobei die kursiven Formelzeichen die Variablen und die steil gesetzten Symbole die physikalischen Maßeinheiten darstellen. Für die erste Zeile der Zustandsgleichung (3.16) erhält man auf diese Weise

$$\begin{aligned} \frac{d\tilde{i}}{d\tilde{t}} \frac{\text{A}}{\text{s}} &= -\frac{R}{L} \tilde{i} \frac{\text{A}}{\text{s}} - \frac{k_M}{L} \tilde{\omega} \frac{\text{rad}}{\text{s}} + \frac{1}{L} \tilde{u} \frac{\text{V}}{\text{s}} \\ &= -\frac{0,5 \frac{\text{V}}{\text{A}}}{50 \cdot 10^{-3} \frac{\text{Vs}}{\text{A}}} \tilde{i} \frac{\text{A}}{\text{s}} - \frac{30 \cdot 10^{-3} \frac{\text{Vs}}{\text{rad}}}{50 \cdot 10^{-3} \frac{\text{Vs}}{\text{A}}} \tilde{\omega} \frac{\text{rad}}{\text{s}} + \frac{1}{50 \cdot 10^{-3} \frac{\text{Vs}}{\text{A}}} \tilde{u} \frac{\text{V}}{\text{s}} \\ \frac{d\tilde{i}}{d\tilde{t}} \cdot \frac{\text{A}}{\text{s}} &= -10 \cdot \frac{\text{A}}{\text{s}} \cdot \tilde{i} - 0,6 \cdot \frac{\text{A}}{\text{s}} \cdot \tilde{\omega} + 20 \cdot \frac{\text{A}}{\text{s}} \cdot \tilde{u} \\ \frac{d\tilde{i}}{d\tilde{t}} &= -10 \cdot \tilde{i} - 0,6 \cdot \tilde{\omega} + 20 \cdot \tilde{u}. \end{aligned}$$

In der letzten Beziehung kommen keine physikalischen Maßeinheiten mehr vor. Diese Zahlenwertgleichung ist eine Gleichung aus den mit der Tilde markierten Signalen, die mit den o. a. physikalischen Einheiten multipliziert die ursprünglichen Signale  $i$ ,  $u$ ,  $\omega$  und  $t$  ergeben. Obwohl sich diese Größen in ihrer Bedeutung unterscheiden, lässt man i. Allg. die Tilde weg und geht wieder zu den ursprünglichen Bezeichnungen über:

$$\frac{di(t)}{dt} = -10 i(t) - 0,6 \omega(t) + 20 u(t).$$

In derselben Weise erhält man für die zweite Zeile der Zustandsgleichung die Beziehung

$$\frac{d\omega(t)}{dt} = 2308 i(t) - 2,3 \omega(t),$$

so dass das Zustandsraummodell als Zahlenwertgleichung geschrieben wie folgt lautet:

$$\text{Gleichstrommotor : } \left\{ \begin{aligned} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} &= \begin{pmatrix} -10 & -0,6 \\ 23,08 & -0,023 \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} 20 \\ 0 \end{pmatrix} u(t) \\ \begin{pmatrix} i(0) \\ \omega(0) \end{pmatrix} &= \begin{pmatrix} i_0 \\ \omega_0 \end{pmatrix} \\ y(t) &= (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix}. \end{aligned} \right.$$

**Änderung der Maßeinheit für die Zeit.** Wie sich später zeigen wird, sind die Ausgleichsvorgänge im Gleichstrommotor sehr schnell, so dass es zweckmäßig ist, die Zeit nicht wie bisher in Sekunden, sondern in Millisekunden zu messen. Dann wird an Stelle von  $t = \tilde{t} \cdot s$  mit  $t = \tilde{t} \cdot 10^{-3}s$  gerechnet. Andererseits will man die Kreisfrequenz wie bisher in  $\frac{\text{rad}}{s}$  angeben, weil dies eine übliche Einheit ist.

Durch die Änderung der Maßeinheit für die Zeit erhält man für die erste Zeile der Zustandsgleichung jetzt

$$\begin{aligned}\frac{d\tilde{i}}{d\tilde{t}} \cdot \frac{A}{10^{-3}s} &= -10 \cdot \frac{A}{s} \cdot \tilde{i} - 0,6 \cdot \frac{A}{s} \cdot \tilde{\omega} + 20 \cdot \frac{A}{s} \cdot \tilde{u} \\ \frac{d\tilde{i}}{d\tilde{t}} \cdot \frac{A}{s} &= -0,01 \cdot \frac{A}{s} \cdot \tilde{i} - 0,0006 \cdot \frac{A}{s} \cdot \tilde{\omega} + 0,02 \cdot \frac{A}{s} \cdot \tilde{u}\end{aligned}$$

und nach Kürzen und Weglassen der Tilden

$$\frac{di(t)}{dt} = -0,01 i(t) - 0,0006 \omega(t) + 0,02 u(t)$$

sowie für die zweite Zeile

$$\frac{d\omega(t)}{dt} = 0,02308 i(t) - 0,000023 \omega(t).$$

Das Zustandsraummodell heißt dann

$$\text{Gleichstrommotor : } \begin{cases} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -0,01 & -0,0006 \\ 0,02308 & -0,000023 \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} 0,02 \\ 0 \end{pmatrix} u(t) \\ y(t) = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix}. \end{cases} \quad (3.21)$$

Durch die Veränderung der Maßeinheit für die Zeit haben sich also die Systemmatrix und der vor der Eingangsgröße stehende Vektor verändert. Die Aussage des Modells ist dabei natürlich dieselbe geblieben. Die mit dem Modell erhaltenen Ergebnisse müssen jetzt in Bezug zum neuen Zeitmaßstab interpretiert werden.  $\square$

### 3.3.4 Blockschaltbild und Signalflussgraph des Zustandsraummodells

Die durch die Gl. (3.17) beschriebenen Zusammenhänge zwischen den Signalen  $u(t)$ ,  $x_i(t)$  und  $y(t)$  sind in Abb. 3.6 durch ein Blockschaltbild veranschaulicht. Doppelpfeile stellen vektorielle Signale dar. Vier Blöcke sind statische Übertragungsglieder mit den Übertragungsfaktoren  $A$ ,  $b$ ,  $c^T$  und  $d$ , mit denen die Eingangsgrößen der Blöcke multipliziert werden, um auf die Ausgangsgrößen zu kommen. Der mittlere Block enthält  $n$  Integratoren für die  $n$  Komponenten des Vektors  $\dot{x}(t)$ .

Eine detailliertere Darstellung als das Blockschaltbild bietet der Signalflussgraph in Abb. 3.7, in dem alle Signale einzeln durch Knoten repräsentiert werden. Die gerichteten Kanten zeigen, welche Signale direkt auf welche anderen Signale einwirken. Die Elemente der Matrix  $A$  bzw. der Vektoren  $b$  und  $c^T$  sowie der Skalar  $d$  treten als Kantengewichte auf. Sind die entsprechenden Elemente gleich null, so wird keine Kante in den Signalflussgraphen eingetragen.

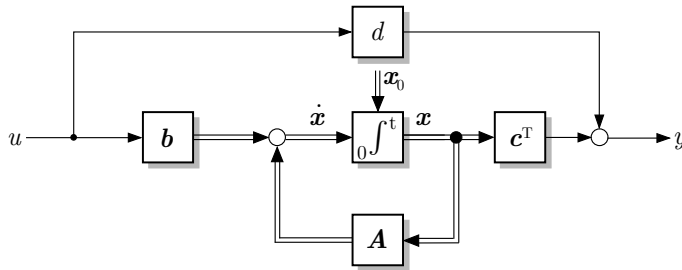


Abb. 3.6: Blockschaltbild der Zustandsraumbeschreibung

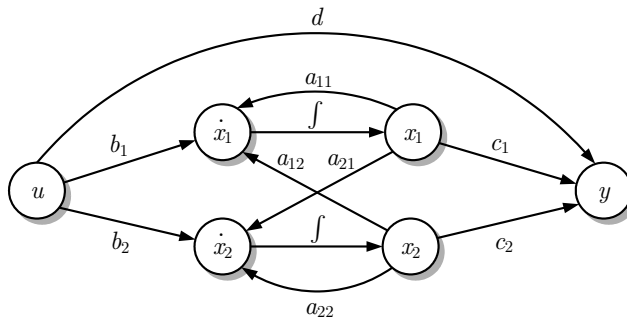


Abb. 3.7: Signalflussgraph eines Systems zweiter Ordnung

**Beispiel 3.5** *Signalflussgraph des Gleichstrommotors*

Abbildung 3.8 zeigt den Signalflussgraphen für den Gleichstrommotor. Das Bild kann direkt aus dem Zustandsraummodell (3.16) abgeleitet werden. An den Kanten stehen die Modellparameter.

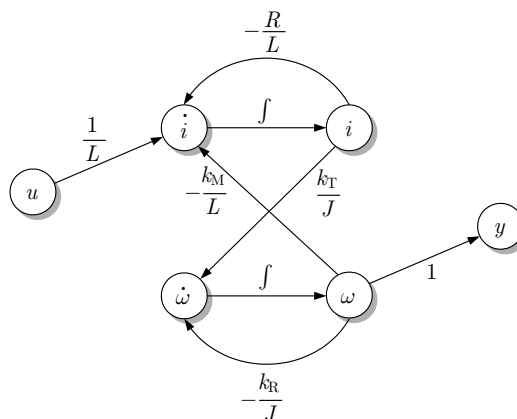


Abb. 3.8: Signalflussgraph des Gleichstrommotors

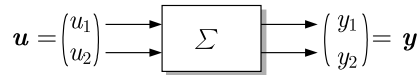


Wichtig bei derartigen grafischen Darstellungen ist häufig, welche Signalkopplungen *nicht* auftreten. Beim Gleichstrommotor wirkt die Eingangsgröße  $u(t)$  nur direkt auf die Ableitung des Stroms  $i(t)$  ein und nicht direkt auf die Ableitung der Drehgeschwindigkeit  $\omega(t)$ . Die Ausgangsgröße hängt nur von der zweiten Zustandsvariablen direkt ab (denn sie ist identisch mit  $\omega(t)$ ). Es gibt keine direkte Kopplung von  $u(t)$  auf  $y(t)$ .  $\square$

### 3.3.5 Zustandsraumdarstellung von Mehrgrößensystemen

Das Zustandsraummodell (3.17) kann für Systeme mit mehr als einer Eingangsgröße  $u(t)$  und mehr als einer Ausgangsgröße  $y(t)$  erweitert werden. Derartige Systeme werden Mehrgrößensysteme genannt. Ihre  $m$  Eingangssignale  $u_i(t)$  und  $p$  Ausgangssignale  $y_i(t)$  werden zu den Vektoren  $\mathbf{u}(t)$  bzw.  $\mathbf{y}(t)$  zusammengefasst (Abb. 3.9):

$$\mathbf{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_m(t) \end{pmatrix}, \quad \mathbf{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_p(t) \end{pmatrix}.$$



**Abb. 3.9:** Mehrgrößensystem mit zwei Eingangsgrößen und zwei Ausgangsgrößen

Das Zustandsraummodell hat dann die allgemeinere Form

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \end{cases}$$

in der die folgenden Größen vorkommen:

Zustandsvektor	$\mathbf{x}$	$n$ -dimensionaler Vektor
Eingangsvektor	$\mathbf{u}$	$m$ -dimensionaler Vektor
Ausgangsvektor	$\mathbf{y}$	$p$ -dimensionaler Vektor
Systemmatrix	$\mathbf{A}$	$(n \times n)$ -Matrix
Steuermatrix, Eingangsmatrix	$\mathbf{B}$	$(n \times m)$ -Matrix
Beobachtungsmatrix, Ausgangsmatrix	$\mathbf{C}$	$(p \times n)$ -Matrix
Durchgangsmatrix	$\mathbf{D}$	$(p \times m)$ -Matrix.

In der in Gl. (3.18) eingeführten Schreibweise ist das lineare Mehrgrößensystem durch das 8-Tupel

$$\Sigma = (\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}^p, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{x}_0)$$

beschrieben.

### 3.3.6 Gleichgewichtszustand linearer Systeme

Für eine konstante Eingangsgröße  $u(t) = \bar{u}$  besitzt ein dynamisches System einen Gleichgewichtszustand  $\bar{x}$ , wenn in diesem Zustand die Beziehung  $\dot{x} = 0$  gilt und das System folglich den Zustand  $\bar{x}$  nicht verlässt. Aus der Zustandsgleichung (3.17) erhält man für  $\bar{x}$  die Beziehung

$$0 = A\bar{x} + b\bar{u}$$

und für Systeme mit  $\det A \neq 0$  daraus

$$\bar{x} = -A^{-1}b\bar{u}. \quad (3.22)$$

Diese Gleichung zeigt, wie die Gleichgewichtslage von der konstanten Eingangsgröße abhängt. Ist  $\bar{u} = 0$ , so liegt der Gleichgewichtszustand im Ursprung des Zustandsraumes:

$$\bar{x} = 0.$$

Die Ausgangsgröße  $\bar{y}$  im Gleichgewichtszustand folgt aus der Ausgangsgleichung:

$$\bar{y} = (d - c^T A^{-1}b) \bar{u}. \quad (3.23)$$

Das Verhältnis des Ausgangs- zum Eingangssignal im Gleichgewichtszustand wird als statischer Verstärkungsfaktor  $k_s$  bezeichnet:

Statischer Verstärkungsfaktor :  $k_s = \frac{\bar{y}}{\bar{u}} = d - c^T A^{-1}b.$

(3.24)

Das statische Verhalten ist dann durch die Beziehung

$$\bar{y} = k_s \bar{u} \quad (3.25)$$

beschrieben. Dieses Modell ist eine statische Beschreibung, die für das dynamische System nur bei konstanter Eingangsgröße und nur dann gültig ist, nachdem alle dynamischen Übergangsvorgänge abgeklungen sind.

Lineare Systeme mit regulärer Systemmatrix  $A$  haben also genau einen Gleichgewichtszustand. Im Kap. 6 wird gezeigt werden, dass alle asymptotisch stabilen Systeme eine reguläre Systemmatrix  $A$  haben, so dass für diese Systeme die hier durchgeführten Untersuchungen zutreffen. Systeme mit singulärer Systemmatrix ( $\det A = 0$ ) haben unendlich viele Gleichgewichtszustände. Dies sind alle Zustände  $\bar{x}$ , für die die Beziehung  $A\bar{x} = 0$  gilt.

#### Beispiel 3.6 Statisches Verhalten des Gleichstrommotors

Für den Gleichstrommotor (3.21) führt Gl. (3.22) für die Eingangsspannung  $\bar{u} = 12 \text{ V}$  auf den Gleichgewichtszustand

$$\bar{x} = \begin{pmatrix} 0,393 \\ 393,4 \end{pmatrix}.$$

Die beiden Komponenten dieses Vektors besagen, dass sich bei der konstanten Eingangsspannung von 12 Volt im statischen Zustand ein Strom von 0,393 Ampere einstellt und sich der Motor mit einer Winkelgeschwindigkeit von  $393,4 \frac{\text{rad}}{\text{s}}$  dreht. Der statische Verstärkungsfaktor ergibt sich aus Gl. (3.24). Er hat den Wert

$$k_s = 32,97$$

und die Maßeinheit  $\frac{\text{rad}}{\text{V}}$ . Für die verwendete Eingangsspannung von 12 Volt dreht sich der Motor mit der Winkelgeschwindigkeit

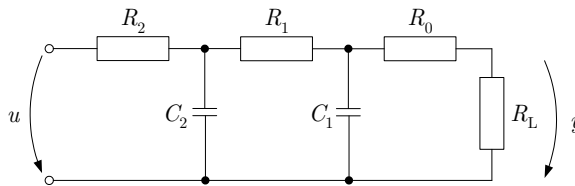
$$\bar{\omega} = k_s \bar{u} = 32,97 \frac{\text{rad}}{\text{V}} \cdot 12 \text{ V} = 393,4 \frac{\text{rad}}{\text{s}}. \quad \square$$

### Beispiel 3.7 Gleichgewichtszustände eines rollenden Balls

Um zu zeigen, dass es Systeme mit unendlich vielen Gleichgewichtszuständen gibt, wird ein Ball betrachtet, der auf der Ebene rollt. Die beiden Ortskoordinaten sind zwei Zustandsvariablen, die Geschwindigkeiten in beide Richtungen zwei weitere Zustandsvariablen. Auf Grund der Reibung kommt der Ball nach einiger Zeit zur Ruhe, nachdem die Eingangsgröße, die die Kraft auf den Ball darstellt, verschwunden ist. Es ist offensichtlich, dass der Ball an jedem Punkt in der Ebene liegen bleiben kann. Das betrachtete System hat also unendlich viele Gleichgewichtslagen, die durch beliebige Werte für die Ortskoordinaten und verschwindende Geschwindigkeiten beschrieben sind.  $\square$

### Aufgabe 3.1\* Zustandsraummodell einer elektrischen Schaltung

Bei der in Abb. 3.10 gezeigten Schaltung ist die Spannung  $u(t)$  die Eingangsgröße und die Spannung  $y(t)$  die Ausgangsgröße.



**Abb. 3.10:** Schaltung mit Eingangsspannung  $u(t)$  und Ausgangsspannung  $y(t)$

1. Welche dynamische Ordnung  $n$  hat dieses System? Was sind geeignete Zustandsvariablen?
2. Stellen Sie das Zustandsraummodell der Schaltung auf.
3. Zeichnen Sie den Signalfussgraphen des Systems und bestimmen Sie, mit welchen physikalischen Größen die Kanten gewichtet sind.
4. Welche Maßeinheiten haben die in den Gleichungen auftretenden Signale und Parameter?  $\square$

**Aufgabe 3.2\*** Zustandsraummodell des invertierten Pendels

Stellen Sie ein Zustandsraummodell für das in Abb. 2.22 auf S. 73 gezeigte invertierte Pendel auf. Die Eingangsgröße  $u(t)$  ist das von einem Elektromotor auf das Pendel ausgeübte Drehmoment. Das Trägheitsmoment  $J$  des einseitig gelagerten Stabs berechnet sich aus der Masse  $m$  und der Länge  $l$  entsprechend

$$J = \frac{ml^2}{3}.$$

Das Modell soll für kleine Winkel  $\phi(t)$  gültig sein. Die Reibung kann vernachlässigt werden.  $\square$

### 3.4 Zustandsraummodell nichtlinearer Systeme

Für viele technischen Systeme entsteht an Stelle des linearen Modells (3.17) ein nichtlineares Zustandsraummodell, das in der Form

Zustandsraummodell nichtlinearer Systeme:

$$\Sigma_{\text{NL}} : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = h(\mathbf{x}(t), u(t)) \end{cases} \quad (3.26)$$

geschrieben wird. Ausführlich geschrieben heißt Gl. (3.26)

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \vdots \\ \dot{x}_n(t) \end{pmatrix} = \begin{pmatrix} g_1(x_1(t), x_2(t), \dots, x_n(t), u(t)) \\ g_2(x_1(t), x_2(t), \dots, x_n(t), u(t)) \\ \vdots \\ g_n(x_1(t), x_2(t), \dots, x_n(t), u(t)) \end{pmatrix}$$

$$y(t) = h(x_1(t), x_2(t), \dots, x_n(t), u(t)).$$

Diese Schreibweise zeigt, dass die Vektorfunktion  $\mathbf{g}(\mathbf{x}, u)$  aus  $n$  Funktionen der Variablen  $x_1, x_2, \dots, x_n$  und  $u$  besteht, mit denen die erste Ableitung der  $n$  Zustandsvariablen bestimmt wird. Da die Argumente Signale sind, hängen sie von der Zeit ab.

Wie bei linearen Systemen hängt die Ausgangsgröße  $y$  nur bei sprunghafigen Systemen direkt von der Eingangsgröße  $u$  ab. Für nicht sprunghafige Systeme heißt die Ausgabe Gleichung

$$y(t) = h(\mathbf{x}(t)). \quad (3.27)$$

Für Mehrgrößensysteme stehen in Gl. (3.26) Vektoren  $\mathbf{u}$  und  $\mathbf{y}$  mit allen Eingangsgrößen bzw. Ausgangsgrößen und die skalare Funktion  $h$  wird durch eine Vektorfunktion  $\mathbf{h}$  ersetzt:

$$\Sigma_{\text{NL}} : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)). \end{cases}$$

Wenn man die in Gl. (3.18) eingeführte Schreibweise für nichtlineare Systeme anwenden will, muss man die Parameter  $\mathbf{A}$ ,  $\mathbf{b}$ ,  $\mathbf{c}^T$  und  $d$  durch die Funktionen  $\mathbf{g}$  und  $h$  bzw.  $\mathbf{h}$  (für Mehrgrößensysteme) ersetzen. Ein nichtlineares Mehrgrößensystem ist also durch das Sixtupel

$$\Sigma_{\text{NL}} = (\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}^p, \mathbf{f}, \mathbf{g}, \mathbf{x}_0) \quad (3.28)$$

beschrieben.

**Gleichgewichtszustände nichtlinearer Systeme.** Für eine konstante Eingangsgröße  $u(t) = \bar{u}$  nimmt das nichtlineare System (3.26) einen Gleichgewichtszustand  $\bar{\mathbf{x}}$  an, wenn  $\dot{\mathbf{x}} = \mathbf{0}$  gilt:

$$\mathbf{0} = \mathbf{g}(\bar{\mathbf{x}}, \bar{u}). \quad (3.29)$$

Diese Gleichung kann gar keine, eine oder mehrere Lösungen haben, je nachdem, wie die Funktion  $\mathbf{g}$  beschaffen ist. Im Gegensatz zu linearen Systemen, die für  $\det \mathbf{A} \neq 0$  genau einen Gleichgewichtszustand  $\bar{\mathbf{x}}$  haben, besitzen nichtlineare Systeme häufig mehrere Gleichgewichtspunkte.

In den Gleichgewichtszuständen stellt sich die folgende Ausgangsgröße ein:

$$\bar{y} = h(\bar{\mathbf{x}}, \bar{u}). \quad (3.30)$$

**Beispiele.** Im Folgenden werden zwei Beispiele behandelt, die zeigen, wie man zu nichtlinearen Zustandsraummodellen kommt und wie man diese Modelle anhand des Blockschaltbilds und der Signalflussgraphen interpretieren kann. Die Beispiele sind ausführlich dargestellt, weil sie auch in Aufgaben und in späteren Kapiteln verwendet werden.

#### Beispiel 3.8 Nichtlineares Zustandsraummodell eines Unterwasserfahrzeugs

Autarke Unterwasserfahrzeuge (Drifter) werden zur Erforschung und Überwachung der Meere eingesetzt. Sie werden von Schiffen ausgesetzt und sollen sich auf einem vorgegebenen Kurs bewegen und Messungen durchführen. Für die Regelung des Tiefgangs ist nur das Volumen des zylinderförmigen Fahrzeugs von Interesse. Der in Abb. 3.11 gezeigte Drifter hat eine Tiefenregelung, bei der die Stellgröße  $u(t)$  das Drifervolumen durch Bewegung eines Kolbens verändert.

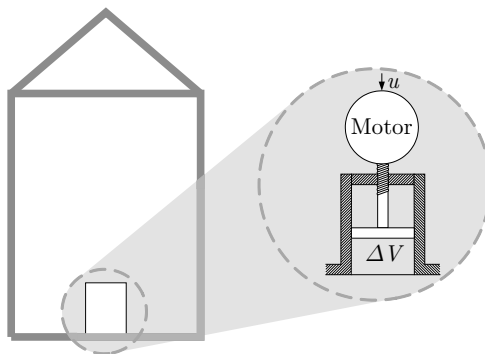


Abb. 3.11: Aufbau eines Unterwasserfahrzeugs

**Beschreibung des Unterwasserfahrzeugs durch Differentialgleichungen.** Ein Modell, das die vertikale Bewegung des Unterwasserfahrzeugs beschreibt, erhält man mit den in Abb. 3.12 angegebenen

Bezeichnungen folgendermaßen. Der Drifter wird als Punktmasse aufgefasst, dessen Bewegung in  $z$ -Richtung durch die drei in der Abbildung eingetragenen Kräfte beeinflusst wird. Aus dem Kräftegleichgewicht folgt die Beziehung

$$m\ddot{z}(t) = F_s - F_a(t) - F_w(t), \quad (3.31)$$

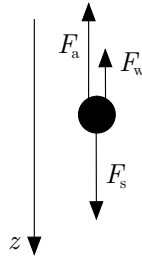
wobei

$$F_s = mg$$

die Schwerkraft,  $F_a(t)$  die Auftriebskraft,  $F_w(t)$  die Widerstandskraft,  $m$  die Masse des Drifters und  $g$  die Erdbeschleunigung bezeichnen. Setzt man die Widerstandskraft proportional zur Geschwindigkeit  $\dot{z}(t)$  an, so gilt

$$F_w(t) = k_W \dot{z}(t), \quad (3.32)$$

wobei die Konstante  $k_W$  den Fluidwiderstand des Drifters beschreibt. Die Auftriebskraft  $F_a(z(t))$  hängt von der Position des Drifters ab. Ihre Berechnung erfordert einige Zwischenschritte, bei denen die Zeitabhängigkeit der Position  $z$  keine Rolle spielt und deshalb nicht kenntlich gemacht wird.



**Abb. 3.12:** Beschreibung des Unterwasserfahrzeugs als Punktmasse

Die Auftriebskraft hängt nach dem archimedischen Prinzip vom Verdrängungsvolumen  $V(z)$  und der Meerwasserdichte  $\rho(z)$  ab:

$$F_a(z) = V(z)\rho(z)g.$$

Sowohl das Volumen als auch die Wasserdichte verändern sich mit der Tiefe  $z(t)$ , was man mit Hilfe der Kompressibilität

$$\kappa = -\frac{1}{V} \frac{dV}{dp}$$

berechnen kann, die die Volumenänderung  $dV$  bei einer Druckveränderung  $dp$  beschreibt.  $\kappa_D$  ist die Kompressibilität des Drifters und  $\kappa_W$  die des Wassers. Wenn  $V_0$ ,  $p_0$  und  $\rho_0$  die bei  $z = 0$  gültigen Werte von  $V(z)$ ,  $p(z)$  und  $\rho(z)$  bezeichnen, so gilt für das Volumen  $V(z)$  des Fahrzeugs in der Position  $z$  näherungsweise die Beziehung

$$\kappa_D = -\frac{1}{V_0} \frac{V(z) - V_0}{p(z) - p_0},$$

woraus man

$$V_0 \kappa_D (p(z) - p_0) = V_0 - V(z)$$

und

$$V(z) = V_0 (1 - \kappa_D (p(z) - p_0))$$

erhält. Auf ähnlichem Weg bekommt man für die durch die Volumenänderung des Wassers hervorgerufene Dichteänderung die Gleichung

$$\rho(z) = \frac{\rho_0}{1 - \kappa_W (p(z) - p_0)}.$$

Daraus folgt für die Auftriebskraft die Darstellung

$$F_a(z) = V_0 (1 - \kappa_D(p(z) - p_0)) \cdot \frac{\rho_0}{1 - \kappa_W(p(z) - p_0)} \cdot g$$

und bei Vernachlässigung quadratischer oder höherer Terme von  $p(z) - p_0$

$$F_a(z) = V_0 \rho_0 (1 + (\kappa_W - \kappa_D)(p(z) - p_0)) \cdot g.$$

Die Druckdifferenz  $p(z) - p_0$  lässt sich näherungsweise aus der aktuellen Tiefe  $z$  und der in der Tiefe  $z = 0$  vorhandenen Wasserdichte  $\rho_0$  berechnen

$$p(z) - p_0 = \rho_0 g z,$$

so dass für die Auftriebskraft schließlich die Beziehung

$$F_a(z) = V_0 \rho_0 g (1 + (\kappa_W - \kappa_D) \rho_0 g z)$$

entsteht.

Das Modell (3.31) hat damit die Form

$$\text{Drifter: } m\ddot{z}(t) = mg - V_0 \rho_0 g (1 + (\kappa_W - \kappa_D) \rho_0 g z(t)) - k_W \dot{z}(t). \quad (3.33)$$

Die Steuerung des Unterwasserfahrzeugs erfolgt durch eine Veränderung des Volumens  $V_0$ . Ein Motor bewegt einen Kolben in einem Zylinder, so dass das durch den Drifter verdrängte Wasservolumen größer bzw. kleiner wird (Abb. 3.11). Nimmt man an, dass sich der Kolben mit einer zur Stellgröße  $u(t)$  proportionalen Geschwindigkeit bewegt, so gilt für die Volumenänderung  $\dot{V}_0$  die Beziehung

$$\dot{V}_0(t) = ku(t), \quad (3.34)$$

wobei  $k$  einen vom Motor und vom Kolben abhängigen Proportionalitätsfaktor kennzeichnet.

**Umformung der Differentialgleichungen in ein Zustandsraummodell.** Zu einem Zustandsraummodell kommt man, indem man den Zustandsvektor

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} V_0(t) \\ z(t) \\ \dot{z}(t) \end{pmatrix}$$

wählt und die Gln. (3.33) und (3.34) in die Form (3.26) überführt. Da man sich für die Tiefe  $z(t)$  interessiert, heißt die Ausgabegleichung

$$y(t) = x_2(t)$$

und das Modell

$$\text{Drifter : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = \begin{pmatrix} ku(t) \\ x_3(t) \\ g - \frac{\rho_0 g}{m} x_1(t) (1 + (\kappa_W - \kappa_D) \rho_0 g x_2(t)) - \frac{k_W}{m} x_3(t) \end{pmatrix} \\ \mathbf{x}(0) = \begin{pmatrix} V_0(0) \\ z(0) \\ \dot{z}(0) \end{pmatrix} \\ y(t) = (0 \ 1 \ 0) \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} \end{array} \right. \quad (3.35)$$

Vergleicht man dieses Modell mit der allgemeinen Darstellung (3.26) des nichtlinearen Zustandsraummodells, so erkennt man folgende Korrespondenzen:

$$g_1(x_1(t), x_2(t), x_3(t), u(t)) = ku(t)$$

$$g_2(x_1(t), x_2(t), x_3(t), u(t)) = x_3(t)$$

$$g_3(x_1(t), x_2(t), x_3(t), u(t)) = g - \frac{\rho_0 g}{m} x_1(t) (1 + (\kappa_W - \kappa_D) \rho_0 g x_2(t)) - \frac{k_W}{m} x_3(t)$$

$$h(x_1(t), x_2(t), x_3(t), u(t)) = x_2(t).$$

In den Funktionen  $g_1$ ,  $g_2$ ,  $g_3$  und  $h$  kommen nicht alle möglichen Argumente  $x_1(t)$ ,  $x_2(t)$ ,  $x_3(t)$  und  $u(t)$  vor. Beispielsweise hängt  $g_1$  nur von  $u(t)$  ab. Die Funktion  $g_3$  ist nichtlinear, denn sie enthält das Produkt  $x_1(t)x_2(t)$  zweier Zustandsvariablen.

**Gleichgewichtszustand des Unterwasserfahrzeugs.** Gleichgewichtszustände erhält man für konstante Eingangsgröße  $u(t) = \bar{u}$ . Die Bedingung (3.29) ist erfüllt, wenn  $\bar{x}_1$ ,  $\bar{x}_2$ ,  $\bar{x}_3$  und  $\bar{u}$  folgende Gleichungen erfüllen:

$$\bar{u} = 0$$

$$\bar{x}_3 = 0$$

$$g - \frac{\rho_0 g}{m} \bar{x}_1 (1 + (\kappa_W - \kappa_D) \rho_0 g \bar{x}_2) = 0.$$

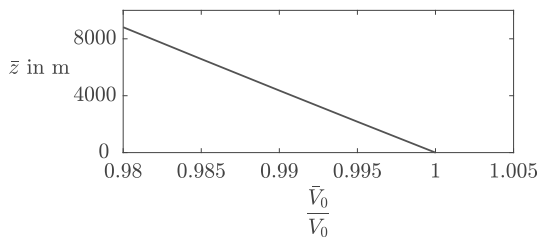
Die letzte Zeile kann in

$$\bar{x}_2 = \frac{m - \bar{x}_1 \rho_0}{\bar{x}_1 \rho_0^2 (\kappa_W - \kappa_D) g}$$

umgeformt werden, die mit den ursprünglich verwendeten Bezeichnungen als

$$\bar{z} = \frac{m - \bar{V}_0 \rho_0}{\bar{V}_0 \rho_0^2 (\kappa_W - \kappa_D) g}$$

geschrieben werden kann. Sie zeigt, dass durch Wahl des Volumens  $\bar{V}_0$  jede beliebige Tiefe  $\bar{z}$  als Gleichgewichtslage des Unterwasserfahrzeugs eingestellt werden kann. Allerdings ist der Zusammenhang zwischen dem einzustellenden Volumen  $\bar{V}_0$  und der damit erreichten Tiefe  $\bar{z}$  nichtlinear.



**Abb. 3.13:** Stationäre Lage des Unterwasserfahrzeugs

Abbildung 3.13 zeigt die stationäre Lage  $\bar{z}$  (gemessen in Metern unter der Ausgangsposition  $z = 0$ ) in Abhängigkeit vom Volumen  $\bar{V}_0$ , das auf das Driftvolumen

$$V_0 = \frac{m}{\rho_0}$$

in der Position  $z = 0$  normiert wurde. Dabei wurden folgende Parameter verwendet:



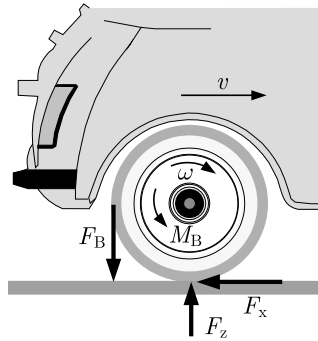
$\rho_0 = 10^3 \frac{\text{kg}}{\text{m}^3}$	Dichte des Wassers
$V_0 = 0,164 \text{ m}^3$	Nominales Volumen des Drifters
$\kappa_W = 0,00049 \cdot 10^{-6} \frac{\text{m}^2}{\text{N}}$	Kompressibilität des Wassers
$\kappa_D = 0,000254 \cdot 10^{-6} \frac{\text{m}^2}{\text{N}}$	Kompressibilität des Drifters.

Das Bild zeigt, dass eine geringe Volumenänderung zu einer großen Veränderung der statischen Position  $\bar{z}$  führt. Um den Drifter in dem im Diagramm gezeigten Bereich von  $\bar{z}$  zu positionieren, reicht eine 2%-ige Volumenänderung. Der Drifter reagiert also sehr empfindlich auf Stellgrößenänderungen. Deshalb erscheint der nichtlineare Zusammenhang zwischen  $\bar{V}_0$  und  $\bar{z}$  in dem angegebenen Bereich als Gerade.

Die Tatsache, dass die Gleichgewichtslage beliebig vorgegeben werden kann, bedeutet nicht, dass sich das Unterwasserfahrzeug stabil in dieser Gleichgewichtslage aufhält. Das Stabilitätsproblem wird in Aufgabe 6.3 behandelt.  $\square$

### Beispiel 3.9 Modell für den Abbremsvorgang eines Fahrzeugs

Ein Modell, das die Geschwindigkeit  $v(t)$  oder den von einem Fahrzeug zurückgelegten Weg  $s(t)$  in Abhängigkeit von der Bremsbeschleunigung beschreibt, enthält als wichtigste Nichtlinearität die Schlupfkennlinie und führt deshalb auf ein nichtlineares Zustandsraummodell. Dieses Modell wird jetzt aufgestellt.



**Abb. 3.14:** Kräfte an einem abgebremsen Rad

Bei dem in Abb. 3.14 gezeigten Rad ist das durch die Bremse erzeugte Bremsmoment  $M_B(t)$  durch die Kraft  $F_B(t) = M_B(t)/r$  ersetzt, wobei  $r$  den Radius des Rades bezeichnet. Die Umfangskraft  $F_x(t)$ , die die Fahrbahn auf den Reifen ausübt, beschleunigt das Rad, so dass die Beziehung

$$J\dot{\omega}(t) = rF_x(t) - rF_B(t), \quad \omega(0) = \omega_0 \quad (3.36)$$

mit  $J$  als Trägheitsmoment des Rades,  $\omega(t)$  als Winkelgeschwindigkeit und  $\dot{\omega}(t)$  als Winkelbeschleunigung gilt.  $\omega_0$  ist die Winkelgeschwindigkeit zur Zeit  $t = 0$ .

Die Kraft  $F_x(t)$  hängt von der Radlast

$$F_z = mg$$

ab, in die neben der Erdbeschleunigung  $g$  der auf das Rad wirkende Anteil  $m$  der Fahrzeugmasse eingeht. Das Verhältnis beider Kräfte heißt Kraftschlussbeanspruchung

$$\mu(t) = \frac{F_x(t)}{F_z}.$$

Es hängt von der relativen Bewegung der Reifenaufstandsfläche (Latsch) zur Fahrbahn ab, die durch den *Schlupf*

$$\lambda(t) = \frac{v(t) - r\omega(t)}{v(t)} = f_1(v(t), \omega(t)) \quad (3.37)$$

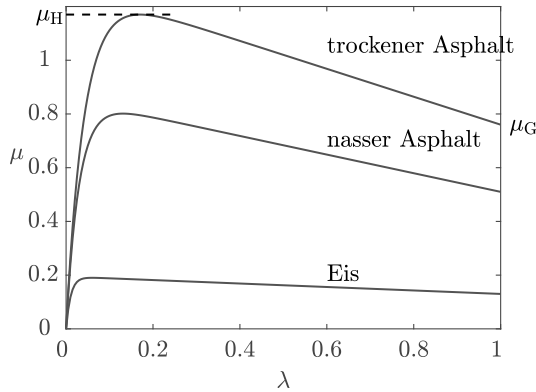
beschrieben wird. In Gl. (3.37) ist  $r\omega(t)$  die Geschwindigkeit, mit der sich die Reifenoberfläche bewegt. Für ein frei rollendes Rad gilt  $v(t) = r\omega(t)$  und folglich  $\lambda(t) = 0$ . Blockiert das Rad beim Bremsvorgang, so ist  $\omega(t) = 0$  und deshalb  $\lambda(t) = 1$ . Wie die Gleichung zeigt, ist der Schlupf nichtlinear von der Geschwindigkeit  $v(t)$  und der Winkelgeschwindigkeit  $\omega(t)$  des Rades abhängig. Diese nichtlineare Funktion wird mit  $f_1$  bezeichnet. Es handelt sich hierbei um eine statische Nichtlinearität, die für die aktuellen Werte  $v(t)$ ,  $\omega(t)$  und  $\lambda(t)$  zu jedem Zeitpunkt  $t$  gilt.

Die nichtlineare Beziehung

$$\mu(t) = f_2(\lambda(t)) \quad (3.38)$$

zwischen dem Schlupf  $\lambda(t)$  und der Kraftschlussbeanspruchung  $\mu(t)$  ist in Abb. 3.15 für unterschiedliche Straßenbeläge dargestellt. Die Funktion  $f_2$  wurde experimentell für Normreifen ermittelt. Nach einem näherungsweise linearen Anstieg erreicht die Kennlinie ihr Maximum, das Haftbeiwert  $\mu_H$  genannt wird, und fällt dann auf den Gleitbeiwert  $\mu_G$  ab, der für das blockierte Rad gilt. Auch hier handelt es sich um eine statische Nichtlinearität, die für jeden Zeitpunkt  $t$  gilt. Unter Verwendung der eingeführten Bezeichnungen erhält man für die Kraft  $F_x(t)$  die Beziehung

$$F_x(t) = F_z \mu(t) = m g \mu(t).$$



**Abb. 3.15:**  $\mu(\lambda)$ -Kennlinien

Diese Kraft erzeugt die Bremsbeschleunigung für das Fahrzeug. Es gilt also

$$\begin{aligned} m\dot{v}(t) &= -F_x(t) \\ \dot{v}(t) &= -\frac{F_x(t)}{m} = -\frac{F_z}{m}\mu(t) \end{aligned}$$

und

$$\dot{v}(t) = -g\mu(t), \quad v(0) = v_0, \quad (3.39)$$

wobei  $v_0$  die Anfangsgeschwindigkeit bezeichnet. Die Bremsbeschleunigung des Fahrzeugs ergibt sich also aus dem Produkt der Erdbeschleunigung mit der Kraftschlussbeanspruchung  $\mu(t)$ . Diese Gleichung und Abb. 3.15 zeigen, dass ein Fahrzeug unter günstigsten Straßenbedingungen nur mit höchstens dem 1,2-fachen der Erdbeschleunigung abgebremst werden kann (was man sich nebenbei merken sollte!).

Um  $\mu(t)$  aus dem Schlupf  $\lambda(t)$  berechnen zu können, muss zunächst  $\omega(t)$  bestimmt werden. Aus Gl. (3.36) erhält man

$$\dot{\omega}(t) = \frac{r}{J} F_z \mu(t) - \frac{r}{J} F_B(t)$$

und

$$\dot{\omega}(t) = \frac{mr}{J} \left( g\mu(t) - \frac{F_B(t)}{m} \right). \quad (3.40)$$

Der Abbremsvorgang ist durch Gl. (3.39) zur Berechnung des Geschwindigkeitsverlaufs, Gl. (3.40) zur Berechnung der Drehgeschwindigkeit des Rades sowie durch die  $\mu(\lambda)$ -Kennlinie (3.38) beschrieben. Die vom Fahrer mit der Bremse vorgegebene Beschleunigung ist die Eingangsgröße

$$u(t) = \frac{F_B(t)}{m}$$

des betrachteten Systems.

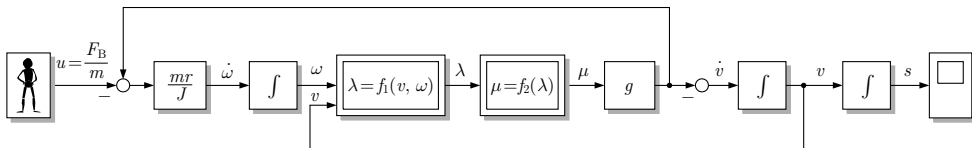
Wenn man den beim Abbremsen zurückgelegten Weg  $s(t)$  berechnen will, so muss man noch die Differentialgleichung

$$\dot{s}(t) = v(t), \quad s(0) = s_0$$

hinzunehmen, bei der  $s_0$  die Anfangsposition des Fahrzeugs angibt. In der Form (3.26) geschrieben heißt das Modell

$$\text{Fahrzeug : } \begin{cases} \begin{pmatrix} \dot{s}(t) \\ \dot{v}(t) \\ \dot{\omega}(t) \end{pmatrix} = \begin{pmatrix} v(t) \\ -gf_2(f_1(v(t), \omega(t))) \\ \frac{mr}{J}(gf_2(f_1(v(t), \omega(t))) - u(t)) \end{pmatrix}, & \begin{pmatrix} s(0) \\ v(0) \\ \omega(0) \end{pmatrix} = \begin{pmatrix} s_0 \\ v_0 \\ \omega_0 \end{pmatrix} \\ s(t) = (1 \ 0 \ 0) \begin{pmatrix} s(t) \\ v(t) \\ \omega(t) \end{pmatrix}. \end{cases}$$

Es ist nichtlinear, denn in der zweiten und dritten Komponente tritt die Kombination  $f_2(f_1(v, \omega))$  der beiden Nichtlinearitäten  $f_1$  und  $f_2$  auf. Der Anfangszustand setzt sich aus den Anfangswerten der drei während der Modellbildung eingeführten Differentialgleichungen zusammen.



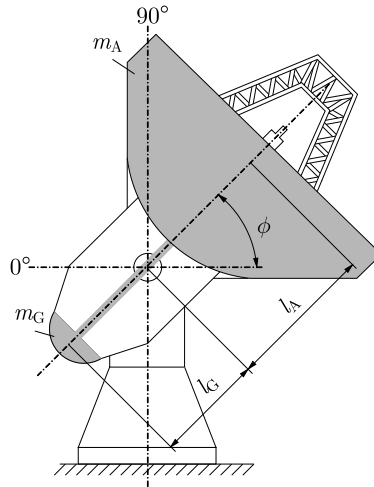
**Abb. 3.16:** Blockschaltbild des Modells zur Beschreibung des Abbremsvorgangs eines Fahrzeugs

Abbildung 3.16 zeigt das zugehörige Blockschaltbild. Es enthält zwei nichtlineare Blöcke, die die Definitionsgleichung (3.37) des Schlupfs und die  $\mu(\lambda)$ -Kennlinie (3.38) beinhalten. Die drei in der Abbildung gezeigten Integratoren haben die Anfangswerte  $\omega_0$ ,  $v_0$  und  $s_0$ .

**Diskussion.** Gleichung (3.40) hat eine interessante Interpretation. Die Größe  $a_{\text{Ist}}(t) = g\mu(t)$  ist die auf das Fahrzeug wirkende Istbeschleunigung und die Größe  $a_{\text{Soll}}(t) = \frac{F_B(t)}{m} = u(t)$  ist die vom Fahrer gewünschte Bremsbeschleunigung. Ihre Differenz ändert die Drehgeschwindigkeit des Rades. Dies geschieht so lange, bis sich ein Schlupf eingestellt hat, bei dessen Kraftschlussbeanspruchung  $\mu(t)$  die Istbeschleunigung gleich der Sollbeschleunigung ist, also

$$\mu(t) = \frac{F_B(t)}{mg}$$

gilt. Diese Ruhelage kann sich natürlich nur dann einstellen, wenn die Bremskraft  $F_B(t)$  die Beziehung  $\frac{F_B(t)}{mg} < \mu_H$  erfüllt, wobei  $\mu_H$  der Haftbeiwert des aktuellen Fahrbahnbelags ist (Abb. 3.15).  $\square$



**Abb. 3.17:** Aufbau eines Radioteleskops

### Aufgabe 3.3\* Zustandsraummodell eines Radioteleskops

Radioteleskope müssen sehr genau positioniert werden, um radioastronomische Messungen durchführen zu können. Neben der Einstellung des Teleskops auf die gewünschte Position muss die Regelung die durch Wind hervorgerufenen Störungen ausgleichen und die Position des Teleskops der Erddrehung nachführen.

In dieser Aufgabe soll ein einfaches Zustandsraummodell des Radioteleskops als Regelstrecke aufgestellt werden. Von den beiden Drehachsen Azimut und Elevation wird hier nur die zweite untersucht, wobei die Position des Teleskops durch den Winkel  $\phi(t)$  beschrieben ist. Eingangsgröße ist das Antriebsmoment  $u(t) = M_\phi(t)$  in Richtung des Winkels  $\phi(t)$ , Ausgangsgröße die Position  $\phi(t)$ . Die

Antenne hat die Masse  $m_A$  und der Ballastausleger die Masse  $m_G$ , die beide als punktförmige Massen betrachtet werden und in den Entfernungen  $l_A$  bzw.  $l_G$  vom Drehpunkt angeordnet sind.

Stellen Sie ein Zustandsraummodell auf, das den Zusammenhang zwischen dem Antriebsmoment und der Position beschreibt. Nehmen Sie an, dass die bei der Bewegung des Teleskops auftretende Reibung proportional zur Drehgeschwindigkeit ist.  $\square$

#### Aufgabe 3.4\* Kontinuierliches Modell eines Behältersystems

Stellen Sie das Zustandsraummodell des in Abb. 3.18 gezeigten Behältersystems auf. Die zylindrischen Behälter haben die Querschnitte  $A_1$  und  $A_2$ . Das Verbindungsrohr befindet sich auf der Höhe  $h_V$  über dem Behälterboden.  $P$  ist eine Pumpe und  $V_1$  und  $V_2$  sind zwei Ventile.

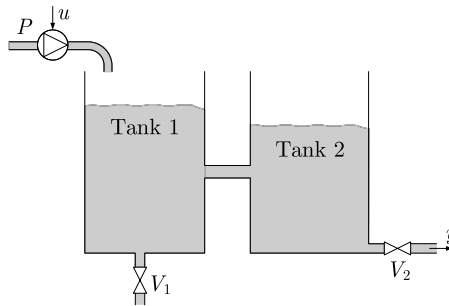


Abb. 3.18: Gekoppelte Behälter

Das nichtlineare Verhalten entsteht durch die nach TORICELLI benannte Beziehung zwischen dem Füllstand  $h$  über der Auslaufstelle und dem am Auslauf anliegenden Druck  $p$

$$p = \sqrt{2gh},$$

bei dem  $g$  die Erdbeschleunigung bezeichnet. Der Durchfluss durch ein Ventil oder eine Rohrleitung ist proportional zum Druckabfall, wobei der Proportionalitätsfaktor  $s$  als Ventilkonstante bezeichnet wird.  $\square$

### 3.5 Linearisierung

Obwohl die meisten Systeme mit hoher Genauigkeit nur durch ein nichtlineares Zustandsraummodell beschrieben werden können, werden für ihre Analyse und den Entwurf von Regelungen häufig lineare Modelle eingesetzt. Dies hat zwei Gründe. Erstens stimmt bei vielen Systemen für kleine Amplituden des Eingangssignals die mit einem linearen Modell berechnete Ausgangsgröße zufriedenstellend mit der wahren Ausgangsgröße überein. Zweitens vereinfacht die Linearitätseigenschaft des Modells die Analyse und den Steuerungsentwurf wesentlich. Deshalb versucht man, auch bei nichtlinearen Systemen mit linearen Modellen zu arbeiten und

fordert dabei, dass das linearisierte Modell das nichtlineare Systemverhalten in der Umgebung eines Arbeitspunkts möglichst genau wiedergibt.

Ausgangspunkt für die Linearisierung muss deshalb die Festlegung eines Arbeitspunktes sein. Der Arbeitspunkt wird durch die Größen  $\bar{x}$ ,  $\bar{u}$  und  $\bar{y}$  beschrieben, die die Bedingungen (3.29) und (3.30) erfüllen. Die erste Bedingung besagt, dass  $\bar{x}$  ein Gleichgewichtszustand des Systems ist, der sich bei der konstanten Eingangsgröße  $u(t) = \bar{u}$  einstellt. Mit der zweiten Bedingung kann man den Wert  $\bar{y}$  der Ausgangsgröße im Arbeitspunkt berechnen.

An Stelle der Originalgrößen  $x(t)$ ,  $u(t)$  und  $y(t)$  stehen im linearisierten Modell die Abweichungen dieser Größen von den Arbeitspunktwerten  $\bar{x}$ ,  $\bar{u}$  und  $\bar{y}$ :

$$\begin{aligned}\delta x(t) &= x(t) - \bar{x} \\ \delta u(t) &= u(t) - \bar{u} \\ \delta y(t) &= y(t) - \bar{y}.\end{aligned}\tag{3.41}$$

Aus (3.26) und (3.41) erhält man

$$\frac{d\delta x(t)}{dt} = \dot{x}(t) = g(\bar{x} + \delta x(t), \bar{u} + \delta u(t)).$$

Wird die Vektorfunktion  $g$  um den Arbeitspunkt  $(\bar{u}, \bar{x})$  in eine Taylorreihe entwickelt, so erhält man

$$\frac{d\delta x(t)}{dt} = g(\bar{x}, \bar{u}) + \left(\frac{\partial g}{\partial x}\right)_{x=\bar{x}, u=\bar{u}} \delta x(t) + \left(\frac{\partial g}{\partial u}\right)_{x=\bar{x}, u=\bar{u}} \delta u(t) + r(\delta x(t), \delta u(t)),$$

wobei  $r(\delta x(t), \delta u(t))$  das Restglied der Taylorreihe darstellt und der erste Summand auf der rechten Seite auf Grund der Bedingung (3.29) verschwindet. Eine häufig, aber nicht immer erfüllte Voraussetzung für diese Vorgehensweise ist, dass die Funktion  $g(x, u)$  im Arbeitspunkt differenzierbar ist.

Die beiden Differentialquotienten beschreiben die Ableitungen der Vektorfunktion  $g(x, u)$  nach dem Vektor  $x$  bzw. dem Skalar  $u$ . Sie stellen demzufolge eine Matrix bzw. einen Vektor dar, die mit  $A$  bzw.  $b$  bezeichnet werden:

$$\left(\frac{\partial g}{\partial x}\right)_{x=\bar{x}, u=\bar{u}} = \begin{pmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \dots & \frac{\partial g_1}{\partial x_n} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \dots & \frac{\partial g_2}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial g_n}{\partial x_1} & \frac{\partial g_n}{\partial x_2} & \dots & \frac{\partial g_n}{\partial x_n} \end{pmatrix}_{x=\bar{x}, u=\bar{u}} = A \tag{3.42}$$

$$\left(\frac{\partial g}{\partial u}\right)_{x=\bar{x}, u=\bar{u}} = \begin{pmatrix} \frac{\partial g_1}{\partial u} \\ \frac{\partial g_2}{\partial u} \\ \vdots \\ \frac{\partial g_n}{\partial u} \end{pmatrix}_{x=\bar{x}, u=\bar{u}} = b. \tag{3.43}$$

Die Matrix (3.42) heißt *Jacobimatrix* oder *Funktionalmatrix*.

Bewegt sich das System in der näheren Umgebung des gegebenen Arbeitspunkts, so sind  $\delta \mathbf{x}(t)$  und  $\delta u(t)$  betragsmäßig klein und das Restglied  $\mathbf{r}(\delta \mathbf{x}, \delta u)$  kann gegenüber den Gliedern erster Ordnung vernachlässigt werden. Damit erhält man die lineare, näherungsweise gültige Beziehung

$$\frac{d \delta \mathbf{x}(t)}{dt} \approx \mathbf{A} \delta \mathbf{x}(t) + \mathbf{b} \delta u(t), \quad \delta \mathbf{x}(0) = \mathbf{x}_0 - \bar{\mathbf{x}}.$$

In gleicher Weise kann die nichtlineare Ausgabegleichung linearisiert werden, wobei die Beziehung

$$\delta y(t) \approx \mathbf{c}^T \delta \mathbf{x}(t) + d \delta u(t)$$

mit

$$\left\| \left( \frac{\partial h}{\partial \mathbf{x}} \right)_{\mathbf{x}=\bar{\mathbf{x}}, u=\bar{u}} = \mathbf{c}^T \right. \quad (3.44)$$

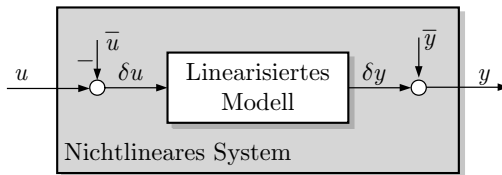
$$\left\| \left( \frac{\partial h}{\partial u} \right)_{\mathbf{x}=\bar{\mathbf{x}}, u=\bar{u}} = d \right. \quad (3.45)$$

entsteht. Vorausgesetzt wird hier, dass die nichtlineare Funktion  $h(\mathbf{x}, u)$  im Arbeitspunkt nach  $\mathbf{x}$  und  $u$  differenzierbar ist.

Da man linearisierte Näherungsmodelle immer unter der Maßgabe verwendet, dass sich das System in der Nähe des Arbeitspunkts bewegt, lässt man das Delta vor den Signalen weg und ersetzt das Rundungszeichen durch das Gleichheitszeichen:

$$\Sigma_{\text{NL}} : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 - \bar{\mathbf{x}} \\ y(t) = \mathbf{c}^T \mathbf{x}(t) + du(t). \end{cases} \quad (3.46)$$

Bei der Interpretation der Ergebnisse, die mit diesem Modell erhalten werden, muss man sich diese vereinfachte Schreibweise ins Gedächtnis zurückrufen und – wie Abb. 3.19 zeigt – von der Eingangsgröße  $u(t)$  den Arbeitspunktwert  $\bar{u}$  abziehen, um den Modelleingang  $\delta u(t)$  zu erhalten. Der Modellausgang  $\delta y(t)$  gibt dann die Abweichung von  $\bar{y}$  an.



**Abb. 3.19:** Anwendung des linearisierten Modells

Gleichung (3.46) stellt ein lineares Zustandsraummodell dar, das die Bewegung des nichtlinearen Systems (3.26) um den Arbeitspunkt  $(\bar{u}, \bar{\mathbf{x}}, \bar{y})$  beschreibt. Das Modell gilt mit guter

Näherung, solange das Restglied  $r(\delta x(t), \delta u(t))$  der Taylorreihenentwicklung vernachlässigbar klein ist. Man sagt auch, dass das Modell lokal um den Arbeitspunkt gilt. Streng genommen müsste man alle Analyse- und Entwurfsergebnisse, die man mit linearisierten Modellen erhalten hat, dahingehend überprüfen, ob sie auch für das nichtlineare System gelten. Man müsste also prüfen, ob die bei der Anwendung der Ergebnisse auftretenden Restglieder der Taylorreihe tatsächlich vernachlässigbar sind. Vielfach werden jedoch linearisierte Modelle eingesetzt, ohne dass man den Gültigkeitsbereich ermittelt.

Als Bezeichnung des Zustandsraummodells wurde der Name  $\Sigma_{NL}$  des Systems (3.26) beibehalten, weil auch das linearisierte Modell eine Repräsentation von  $\Sigma_{NL}$  ist. Wenn man betonen will, dass sich diese Darstellung auf die Abweichungen vom Arbeitspunkt bezieht und außerdem in einer größeren Umgebung des Arbeitspunktes nur näherungsweise gilt, kann man dem linearisierten Modell auch einen anderen Namen geben.

**Aufgabe 3.5\*** *Linearisierung des Modells eines Unterwasserfahrzeugs*

Linearisieren Sie das nichtlineare Modell (3.35) des Drifters um einen geeignet gewählten Arbeitspunkt. □

**Aufgabe 3.6\*** *Linearisierte Beschreibung des Abbremsvorgangs eines Fahrzeugs*

Im Beispiel 3.9 wurde das nichtlineare Modell (3.39), (3.40) aufgestellt, das zusammen mit der in Abb. 3.15 gezeigten Kennlinie den Abbremsvorgang eines Fahrzeugs beschreibt. Wenn man sich, wie beim Entwurf eines Antiblockiersystems (ABS), nur für den Zusammenhang zwischen der gewünschten Bremsbeschleunigung  $u(t) = F_B(t)/m$  und dem Schlupf  $\lambda(t)$  interessiert, so kann man die Geschwindigkeit als konstant annehmen, weil sich auf Grund der erheblichen Massenunterschiede die Drehgeschwindigkeit  $\omega(t)$  des Rades viel schneller ändern kann als die Geschwindigkeit  $v(t)$  des Fahrzeugs. Man setzt also  $v = \text{konst.}$  und verwendet Gl. (3.40) zur Beschreibung von  $\omega(t)$ .

1. Wie vereinfacht sich das Blockschaltbild 3.16 unter den genannten Annahmen? Für welche Analyse- und Entwurfsaufgaben kann dieses Modell eingesetzt werden, für welche nicht?
2. Formen Sie die Modellgleichungen in die Standardform (3.26) um.
3. Wählen Sie einen geeigneten Arbeitspunkt für die Linearisierung.
4. Linearisieren Sie das Modell um den Arbeitspunkt.
5. Stellen Sie das lineare Modell als Blockschaltbild dar. □

**Aufgabe 3.7\*** *Modell eines Wärmeübertragers*

Zustandsraummodelle, die die Form

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \tilde{\mathbf{A}}\mathbf{x}(t) + \mathbf{e}, & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \tilde{\mathbf{c}}^T \mathbf{x}(t) \end{cases} \quad (3.47)$$

haben, werden *affin* genannt. Der Vektor  $\mathbf{e}$  ist konstant. Beispielsweise führt die Modellierung des in Abb. 3.20 dargestellten Wärmeübertragers auf ein affines Modell, wie die folgenden Betrachtungen zeigen.



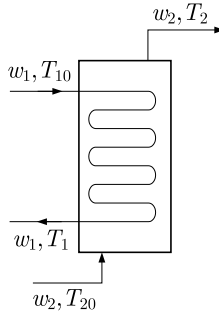


Abb. 3.20: Wärmeübertrager

Die mit dem Massenstrom  $w_1$  und der Temperatur  $T_{10}$  einströmende Flüssigkeit wird im Wärmeübertrager auf die Temperatur  $T_1(t)$  abgekühlt, währenddessen sich die Kühlflüssigkeit mit dem Massenstrom  $w_2$  von der Temperatur  $T_{20}$  auf die Temperatur  $T_2(t)$  erwärmt. Das Modell des Wärmeübertragers erhält man aus einer Energiebilanz, bei der der Einfachheit halber angenommen wird, dass sich die Temperaturen beider Flüssigkeiten nicht in Abhängigkeit vom Ort ändern, die Flüssigkeiten also homogen durchmischt sind (was eine grobe Näherung bedeutet). Eine Änderung der Temperatur  $T_1$  wird durch die Differenz der einströmenden Energiemenge  $w_1 c_1 T_{10}$  und der ausströmenden Energiemenge  $w_1 c_1 T_1(t)$  abzüglich der Wärme  $kA(T_1(t) - T_2(t))$ , die über die Austauschfläche  $A$  zwischen beiden Flüssigkeiten an die Kühlflüssigkeit abgegeben wird, hervorgerufen:

$$m_1 c_1 \dot{T}_1(t) = w_1 c_1 T_{10} - w_1 c_1 T_1(t) - kA(T_1(t) - T_2(t)).$$

Dabei bezeichnet  $m_1$  die Masse der sich in der Rohrleitung befindenden Flüssigkeit,  $c_1$  die spezifische Wärmekapazität und  $k$  einen Koeffizienten, der den Wärmeaustausch pro Flächeneinheit und Temperaturdifferenz beschreibt. Dies sind konstante Parameter, die vom Aufbau und dem Betrieb des Wärmeübertragers abhängen. Für das Kühlmittel gilt dementsprechend die Gleichung

$$m_2 c_2 \dot{T}_2(t) = w_2 c_2 T_{20} - w_2 c_2 T_2(t) + kA(T_1(t) - T_2(t)).$$

1. Wenn die Matrix  $\tilde{A}$  regulär ist, hat das System (3.47) einen eindeutigen Arbeitspunkt. Linearisieren Sie das Modell um diesen Arbeitspunkt. Begründen Sie, weshalb das Modell (3.47) nichtlinear ist.
2. Zeigen Sie, dass sich die für den Wärmeübertrager angegebenen Gleichungen zu einem affinen Zustandsraummodell (3.47) zusammenfassen lassen, wenn die Größen  $w_1$ ,  $w_2$ ,  $T_{10}$  und  $T_{20}$  konstant sind und die Temperatur  $T_1(t)$  die Ausgangsgröße ist.
3. Wie heißt der Arbeitspunkt des Wärmeübertragers und wie das um diesen Arbeitspunkt linearisierte Modell? Was bedeutet der Arbeitspunkt physikalisch?  $\square$

### 3.6 Kompositionale Modellbildung kontinuierlicher Systeme

Da die meisten technischen Systeme aus klar abgegrenzten Teilsystemen bestehen, ist es bei der Modellbildung zweckmäßig, das Modell des Gesamtsystems schrittweise aus den Modellen der Teilsysteme aufzustellen. Man bildet dabei zunächst die Zustandsraummodelle der Teilsysteme und verknüpft diese dann entsprechend den Verkopplungen der Teilsysteme (kompositionale Modellbildung).

Die Idee und die wichtigsten Modellbildungsschritte werden im Folgenden an den drei Grundstrukturen erläutert, in denen zwei kontinuierliche Teilsysteme verkoppelt sein können: der Reihenschaltung, der Parallelschaltung und der Rückführschaltung. Die gezeigten Kompositionsschritte können bei Systemen mit mehr als zwei Teilsystemen mehrfach nacheinander angewendet werden. Die Vorgehensweise, mit dem Blockschaltbild das Gesamtsystem als Verkopplung von Teilsystemen darzustellen, dann für die Teilsysteme Zustandsraummodelle abzuleiten und schließlich diese Modelle zu einem Modell des Gesamtsystems zu verknüpfen, ist eine typische Vorgehensweise, mit der einerseits die Komplexität des Modellbildungsprozesses reduziert wird und die andererseits sehr zum Verständnis des Systemverhaltens beiträgt.

Die beiden zu verknüpfenden Teilsysteme  $S_1$  und  $S_2$  sind durch die linearen Zustandsraummodelle

$$S_1 : \begin{cases} \dot{\mathbf{x}}_1(t) = \mathbf{A}_1 \mathbf{x}_1(t) + \mathbf{b}_1 u_1(t), & \mathbf{x}_1(0) = \mathbf{x}_{10} \\ y_1(t) = \mathbf{c}_1^T \mathbf{x}_1(t) \end{cases} \quad (3.48)$$

und

$$S_2 : \begin{cases} \dot{\mathbf{x}}_2(t) = \mathbf{A}_2 \mathbf{x}_2(t) + \mathbf{b}_2 u_2(t), & \mathbf{x}_2(0) = \mathbf{x}_{20} \\ y_2(t) = \mathbf{c}_2^T \mathbf{x}_2(t) \end{cases} \quad (3.49)$$

beschrieben. Die drei Kopplungsstrukturen unterscheiden sich in der Frage, wie die Eingänge und Ausgänge der beiden Teilsysteme untereinander und mit der Eingangsgröße und Ausgangsgröße des Gesamtsystems in Beziehung stehen. Das Gesamtsystem ist in jedem Fall ein System mit dem Eingang  $u(t)$ , dem Zustand  $\mathbf{x}(t)$  und dem Ausgang  $y(t)$ , für das ein lineares Zustandsraummodell der Form

$$S : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) \end{cases} \quad (3.50)$$

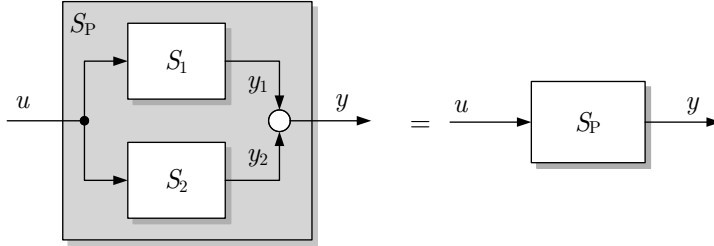
aufgestellt werden soll. Die zu beantwortende Frage ist, wie sich der Zustand  $\mathbf{x}$ , die Matrix  $\mathbf{A}$  und die Vektoren  $\mathbf{b}$  und  $\mathbf{c}^T$  aus den Modellelementen der Teilsysteme zusammensetzen.

Gemeinsam gilt für alle drei Strukturen, dass man den Zustand der Zusammenschaltung entsprechend

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{pmatrix} \quad (3.51)$$

aus den Zustandsvektoren  $\mathbf{x}_1(t)$  und  $\mathbf{x}_2(t)$  der Teilsysteme erhält, indem man diese Vektoren untereinander schreibt, denn das „Gedächtnis“ des Gesamtsystems muss die in den Zuständen beider Teilsysteme gespeicherten Informationen aufnehmen. Nur in Ausnahmefällen stehen im Vektor  $\mathbf{x}(t)$  redundante Informationen, so dass man im Zustandsraummodell des Gesamtsystems mit einem kürzeren Vektor arbeiten kann. Der Anfangszustand des Gesamtsystems ist folglich

$$\mathbf{x}_0 = \begin{pmatrix} \mathbf{x}_{10} \\ \mathbf{x}_{20} \end{pmatrix}.$$



**Abb. 3.21:** Parallelschaltung der Teilsysteme  $S_1$  und  $S_2$

**Parallelschaltung.** In einer Parallelschaltung  $S_P$  sind die beiden Teilsysteme entsprechend der Gleichungen

$$\begin{aligned} u(t) &= u_1(t) = u_2(t) \\ y(t) &= y_1(t) + y_2(t) \end{aligned} \quad (3.52)$$

verkoppelt (Abb. 3.21). Durch Untereinanderschreiben der Zustandsgleichungen (3.48) und (3.49) erhält man

$$\begin{pmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{pmatrix} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{A}_2 \end{pmatrix} \begin{pmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{pmatrix} + \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix} u(t)$$

sowie aus den Ausgabegleichungen und Gl. (3.52) die Beziehung

$$y(t) = (\mathbf{c}_1^T \quad \mathbf{c}_2^T) \begin{pmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{pmatrix}.$$

Unter Beachtung der Zustandsdefinition (3.51) ergibt sich daraus das Zustandsraummodell (3.50) der Parallelschaltung mit

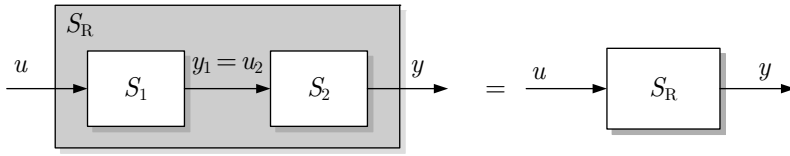
$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{O} \\ \mathbf{O} & \mathbf{A}_2 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix}, \quad \mathbf{c}^T = (\mathbf{c}_1^T \quad \mathbf{c}_2^T). \quad (3.53)$$

**Reihenschaltung.** Bei der Reihenschaltung  $S_R$  gelten für die Eingangs- und Ausgangsgrößen die Beziehungen

$$u(t) = u_1(t) \quad (3.54)$$

$$y(t) = y_2(t) \quad (3.55)$$

$$u_2(t) = y_1(t) \quad (3.56)$$



**Abb. 3.22:** Reihenschaltung der Teilsysteme  $S_1$  und  $S_2$

(Abb. 3.22). Die Zustandsgleichungen (3.48) und (3.49) führen zusammen mit den Gln. (3.54) und (3.56) auf

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} A_1 & O \\ b_2 c_1^T & A_2 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} b_1 \\ 0 \end{pmatrix} u(t).$$

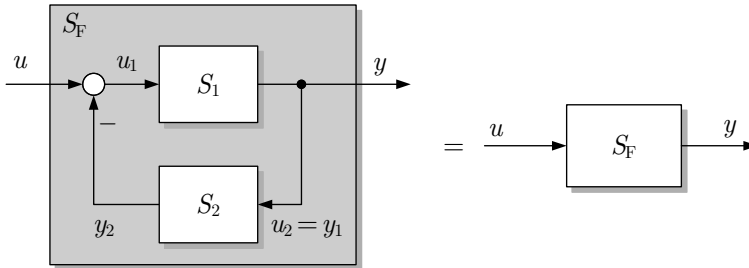
Mit Gl. (3.55) erhält man für die Ausgangsgröße die Beziehung

$$y(t) = (0^T \quad c_2^T) \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}.$$

Die Reihenschaltung ist folglich durch das Zustandsraummodell (3.50) mit

$$A = \begin{pmatrix} A_1 & O \\ b_2 c_1^T & A_2 \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ 0 \end{pmatrix}, \quad c^T = (0^T \quad c_2^T) \quad (3.57)$$

beschrieben.



**Abb. 3.23:** Rückführschaltung der Teilsysteme  $S_1$  und  $S_2$

**Rückführschaltung.** Bei der Rückführschaltung  $S_F$  wird die Ausgangsgröße  $y_1(t)$  des sich im Vorwärtszweig befindenden Teilsystems über das zweite Teilsystem auf den Eingang  $u_1(t)$  zurückgeführt, so dass die Teilsysteme stark zusammenhängend sind (vgl. Definition 2.1 auf S. 66). Das Minuszeichen in der Summationsstelle ist hier eingeführt, weil derartige Rückführschaltungen vor allem bei der Betrachtung von Regelkreisen eine Rolle spielen, bei denen das rückgeführte Signal stets mit einem Minuszeichen an den Eingang des Vorwärtszweiges gekoppelt wird. Die Koppelbeziehungen heißen demzufolge

$$u_1(t) = u(t) - y_2(t) \quad (3.58)$$

$$y(t) = y_1(t) \quad (3.59)$$

$$u_2(t) = y_1(t). \quad (3.60)$$

Aus Gl. (3.58) erhält man

$$u_1(t) = u(t) - \mathbf{c}_2^T \mathbf{x}_2(t),$$

so dass die Zustandsgleichung

$$\begin{pmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{pmatrix} = \begin{pmatrix} \mathbf{A}_1 & -\mathbf{b}_1 \mathbf{c}_2^T \\ \mathbf{b}_2 \mathbf{c}_1^T & \mathbf{A}_2 \end{pmatrix} \begin{pmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{pmatrix} + \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{0} \end{pmatrix} u(t)$$

entsteht. Für das Zustandsraummodell (3.50) des Gesamtsystems ergibt sich damit

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & -\mathbf{b}_1 \mathbf{c}_2^T \\ \mathbf{b}_2 \mathbf{c}_1^T & \mathbf{A}_2 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{0} \end{pmatrix}, \quad \mathbf{c}^T = (\mathbf{c}_1^T \quad \mathbf{0}^T). \quad (3.61)$$

**Rückführschaltung mit algebraischer Schleife.** Die Zusammenfassung der drei Schaltungen wurde hier anhand der beiden Zustandsraummodelle (3.48) und (3.49) erläutert, in denen die Ausgangsgrößen  $y_1(t)$  und  $y_2(t)$  nicht direkt von den Eingangsgrößen  $u_1(t)$  bzw.  $u_2(t)$  abhängen. An der beschriebenen Vorgehensweise ändert sich für die Parallelschaltung und die Reihenschaltung wenig, wenn diese direkten Abhängigkeiten durch zusätzliche Summanden  $+d_1 u_1(t)$  bzw.  $+d_2 u_2(t)$  in den Ausgabegleichungen eingeführt werden, so wie es in der allgemeinen Form (3.17) des Zustandsraummodells vorgesehen ist. Bei der Rückführschaltung tritt jedoch eine Schwierigkeit hinzu, auf die hier kurz eingegangen werden soll.

Verknüpft man die erweiterten Zustandsraummodelle

$$S_1 : \begin{cases} \dot{\mathbf{x}}_1(t) = \mathbf{A}_1 \mathbf{x}_1(t) + \mathbf{b}_1 u_1(t), & \mathbf{x}_1(0) = \mathbf{x}_{10} \\ y_1(t) = \mathbf{c}_1^T \mathbf{x}_1(t) + d_1 u_1(t) \end{cases} \quad (3.62)$$

und

$$S_2 : \begin{cases} \dot{\mathbf{x}}_2(t) = \mathbf{A}_2 \mathbf{x}_2(t) + \mathbf{b}_2 u_2(t), & \mathbf{x}_2(0) = \mathbf{x}_{20} \\ y_2(t) = \mathbf{c}_2^T \mathbf{x}_2(t) + d_2 u_2(t) \end{cases} \quad (3.63)$$

zu einer Rückführschaltung, so erhält man aus den Gln. (3.58), (3.60), (3.62) und (3.63) die Beziehung

$$\begin{aligned} u_1(t) &= u(t) - y_2(t) \\ &= u(t) - \mathbf{c}_2^T \mathbf{x}_2(t) - d_2 u_2(t) \\ &= u(t) - \mathbf{c}_2^T \mathbf{x}_2(t) - d_2 y_1(t) \\ &= u(t) - \mathbf{c}_2^T \mathbf{x}_2(t) - d_2 \mathbf{c}_1^T \mathbf{x}_1(t) - d_2 d_1 u_1(t), \end{aligned}$$

in der die Eingangsgröße  $u_1(t)$  des ersten Teilsystems auf beiden Seiten vorkommt. Diese Eingangsgröße hängt also direkt von sich selbst ab. Die erhaltene Beziehung lässt sich in die Gleichung

$$(1 + d_1 d_2) u_1(t) = u(t) - \mathbf{c}_2^T \mathbf{x}_2(t) - d_2 \mathbf{c}_1^T \mathbf{x}_1(t) \quad (3.64)$$

umformen, die nur dann eine Lösung  $u_1$  hat, wenn

$$d_1 d_2 \neq -1 \quad (3.65)$$

gilt. Man spricht bei der direkten Abhängigkeit von  $u_1(t)$  von sich selbst von einer *algebraischen Schleife*, weil diese Abhängigkeit durch eine algebraische Gleichung (3.64) beschrieben wird. Diese Schleife wird durch die beiden rückgekoppelten Teilsysteme gebildet (Abb. 3.23). Wenn  $d_1$  oder  $d_2$  verschwindet, ist  $u_1(t)$  nur über Differentialgleichungen von sich selbst abhängig und es tritt keine algebraische Schleife auf.

Unter der Lösbarkeitsbedingung (3.65) gilt

$$u_1(t) = \frac{1}{1 + d_1 d_2} u(t) - \frac{1}{1 + d_1 d_2} \mathbf{c}_2^T \mathbf{x}_2(t) - \frac{d_2}{1 + d_1 d_2} \mathbf{c}_1^T \mathbf{x}_1(t),$$

so dass für die Rückkopplungsschaltung das Zustandsraummodell

$$S_F : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t) \\ y(t) = \mathbf{c}^T \mathbf{x}(t) + d u(t) \end{cases}$$

mit

$$\begin{aligned} \mathbf{A} &= \begin{pmatrix} \mathbf{A}_1 - \frac{d_2}{1 + d_1 d_2} \mathbf{b}_1 \mathbf{c}_1^T & -\frac{1}{1 + d_1 d_2} \mathbf{b}_1 \mathbf{c}_2^T \\ \frac{1}{1 + d_1 d_2} \mathbf{b}_2 \mathbf{c}_1^T & \mathbf{A}_2 - \frac{d_1}{1 + d_1 d_2} \mathbf{b}_2 \mathbf{c}_2^T \end{pmatrix} \\ \mathbf{b} &= \begin{pmatrix} \frac{1}{1 + d_1 d_2} \mathbf{b}_1 \\ \frac{d_1}{1 + d_1 d_2} \mathbf{b}_2 \end{pmatrix} \\ \mathbf{c}^T &= \begin{pmatrix} \frac{1}{1 + d_1 d_2} \mathbf{c}_1^T & \frac{d_1}{1 + d_1 d_2} \mathbf{c}_2^T \end{pmatrix} \\ d &= \frac{d_1}{1 + d_1 d_2} \end{aligned} \quad (3.66)$$

entsteht. Auch in der Rückführschaltung ist die Ausgangsgröße direkt von der Eingangsgröße abhängig.

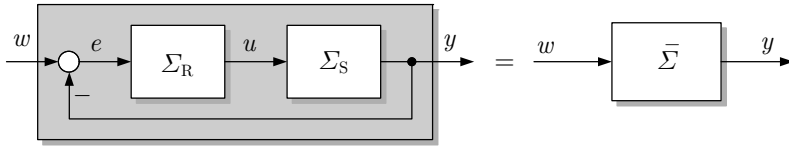
### Aufgabe 3.8 Zustandsraummodell eines Regelkreises

Ein Regelkreis hat die in Abb. 3.24 gezeigte Struktur (vgl. Abb. 2.21 auf S. 72). Die Regelstrecke wird durch ein Zustandsraummodell der Form (3.17) beschrieben. Als Regler wird ein PI-Regler

$$\Sigma_R : \begin{cases} \dot{x}_r(t) = y(t) - w(t), & x_r(0) = x_{r0} \\ u(t) = -k_I x_r(t) - k_P (y(t) - w(t)) \end{cases}$$

verwendet (dessen Elemente im Abschn. 7.4.1 auf S. 234 noch ausführlich erläutert werden). Dieser Regler hat den Zustand  $x_r(t)$ . Die Regelabweichung ist  $e(t) = w(t) - y(t)$ .

Fassen Sie beide Modelle zum Regelkreis  $\bar{\Sigma}$  zusammen. Tritt eine algebraische Schleife auf?  $\square$



**Abb. 3.24:** Blockschaltbild eines Regelkreises

## Literaturhinweise

Die Modellbildung setzt die Kenntnis der in dem betreffenden System ablaufenden physikalischen Vorgänge voraus. Die dafür notwendigen Grundkenntnisse sind in der für das betreffende Fachgebiet einschlägigen Literatur bzw. in Handbüchern zu finden.

So wird die thermodynamische Modellierung von Wärmeübertragern in [7] beschrieben. Eine kurze Einführung in die regelungstechnischen Probleme bei der Positionierung von Radioteleskopen steht in [48]. Dort ist auch beschrieben, wie das in Aufgabe 3.3 behandelte Modell für die Praxis erweitert werden muss und wie die in dem Modell vorkommenden Parameter experimentell bestimmt werden können. Das im Beispiel 3.9 behandelte Modell für das Abbremsen eines Fahrzeugs wurde unter Verwendung der Literatur [15, 75, 87] aufgestellt. Eine Beschreibung des Einsatzgebietes und der Steuerungsprobleme für das im Beispiel 3.8 beschriebene Unterwasserfahrzeug ist in [24, 29] zu finden.





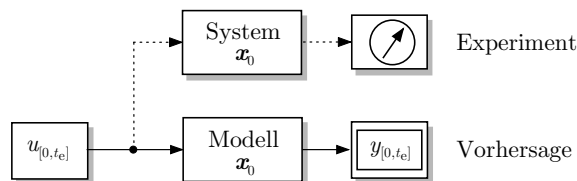
# 4

## Verhalten kontinuierlicher Systeme

*Dieses Kapitel behandelt Methoden für die Vorhersage des Systemverhaltens mit Hilfe des Zustandsraummodells. Dabei wird offensichtlich, wie die Bewegung eines dynamischen Systems vom Anfangszustand und vom Verlauf der Eingangsgröße abhängt.*

### 4.1 Vorhersage des Systemverhaltens

Die im Kap. 3 aufgestellten dynamischen Modelle können für unterschiedliche Automatisierungsaufgaben eingesetzt werden. In diesem Kapitel wird untersucht, wie mit diesen Modellen das Verhalten eines kontinuierlichen Systems vorhergesagt werden kann.



**Abb. 4.1:** Vorhersageaufgabe

Zu bestimmen ist der Verlauf der Ausgangsgröße  $y$  für ein Zeitintervall  $0 \leq t \leq t_e$ , der mit  $y_{[0,t_e]}$  bezeichnet wird. Das Zustandsraummodell zeigt, dass für die Vorhersageaufgabe neben sämtlichen Modellparametern zwei Dinge bekannt sein müssen: der Anfangszustand  $x_0$  und

der Verlauf der Eingangsgröße  $u(t)$  für das betrachtete Zeitintervall  $0 \leq t \leq t_e$ , der mit  $u_{[0,t_e]}$  bezeichnet wird.

Abbildung 4.1 veranschaulicht die Vorhersageaufgabe. Im oberen Teil der Abbildung ist ein Experiment dargestellt, bei dem das reale System mit der verwendeten Eingangsgröße erregt und der Systemausgang gemessen wird. Bei der Vorhersage (Prädiktion) wird der im unteren Teil der Abbildung gezeigte Weg beschritten, bei dem aus der Eingangsgröße mit Hilfe eines Modells die Ausgangsgröße berechnet (und grafisch dargestellt) wird. Wenn idealisierender Weise angenommen wird, dass das Modell das gegebene System exakt repräsentiert und der Anfangszustand des Systems genau bekannt ist, so stimmen beide Ergebnisse überein.

#### Vorhersageaufgabe für kontinuierliche Systeme

Gegeben: Zustandsraummodell  
 Anfangszustand  $\mathbf{x}_0$   
 Verlauf der Eingangsgröße  $u_{[0,t_e]}$   
 Gesucht: Verlauf der Ausgangsgröße  $y_{[0,t_e]}$

Bei bestimmten Anwendungen ist an Stelle des Systemausgangs der Zustand  $\mathbf{x}_{[0,t_e]}$  gesucht.

Die folgenden Abschnitte zeigen, dass man das Verhalten linearer Systeme aus der Lösung des linearen Zustandsraummodells gewinnen kann, wobei es möglich ist, die Lösung explizit in Abhängigkeit vom Anfangszustand und von der Eingangsgröße darzustellen. Da nichtlineare Differentialgleichungen nur in Ausnahmefällen geschlossene Lösungen haben, müssen nichtlineare Zustandsraummodelle i. Allg. numerisch gelöst werden.

## 4.2 Verhalten linearer Systeme

### 4.2.1 Lösung der Zustandsgleichung

Die Zustandsgleichung

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (4.1)$$

ist eine lineare Differentialgleichung, für deren Lösung bekanntlich zunächst die homogene Gleichung

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (4.2)$$

betrachtet wird. In dem Lösungsansatz

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{k} \quad (4.3)$$

ist  $\mathbf{k}$  ein  $n$ -dimensionaler Vektor. Die  $(n \times n)$ -Matrix  $e^{\mathbf{A}t}$  ist die *Matrixexponentialfunktion*, die durch die Reihe

$$e^{\mathbf{A}t} = \sum_{i=0}^{\infty} \frac{\mathbf{A}^i t^i}{i!} = \mathbf{I} + \mathbf{A}t + \frac{\mathbf{A}^2}{2!} t^2 + \frac{\mathbf{A}^3}{3!} t^3 + \dots \quad (4.4)$$

definiert ist. Diese Reihe konvergiert für alle quadratischen Matrizen  $\mathbf{A}$  und alle Werte für  $t$ , so dass die Differentiation mit der Summenbildung vertauscht werden kann und man für die Ableitung die Beziehung

$$\begin{aligned}\frac{d}{dt}e^{\mathbf{A}t} &= \mathbf{A} + \mathbf{A}^2t + \frac{\mathbf{A}^3}{2!}t^2 + \dots = \mathbf{A} \left( \mathbf{I} + \mathbf{A}t + \frac{\mathbf{A}^2}{2!}t^2 + \frac{\mathbf{A}^3}{3!}t^3 + \dots \right) \\ &= \mathbf{A} e^{\mathbf{A}t} = e^{\mathbf{A}t} \mathbf{A}\end{aligned}$$

erhält. Die letzte Beziehung ist analog zu der für e-Funktionen bekannten Differentiationsregel  $\frac{d}{dt}e^{at} = ae^{at}$ .

Dass mit dem Ansatz (4.3) tatsächlich die allgemeine Lösung der homogenen Differentialgleichung gefunden ist, erkennt man durch Einsetzen dieses Ansatzes in Gl. (4.2).

Für die Lösung der inhomogenen Differentialgleichung (4.1) wird entsprechend der Methode der Variation der Konstanten der Vektor  $\mathbf{k}$  als zeitabhängige Größe betrachtet, was auf den Ansatz

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{k}(t)$$

führt. Differenziert man den Ansatz und setzt ihn in die Zustandsgleichung ein, so erhält man

$$\mathbf{A}e^{\mathbf{A}t} \mathbf{k}(t) + e^{\mathbf{A}t} \dot{\mathbf{k}}(t) = \mathbf{A}e^{\mathbf{A}t} \mathbf{k}(t) + \mathbf{b}u(t).$$

Die Matrixexponentialfunktion  $e^{\mathbf{A}t}$  ist für beliebige Matrizen  $\mathbf{A}$  und Zeiten  $t$  invertierbar, wobei

$$\left(e^{\mathbf{A}t}\right)^{-1} = e^{-\mathbf{A}t} = e^{\mathbf{A}(-t)}$$

gilt. Multipliziert man die vorherige Gleichung mit  $\left(e^{\mathbf{A}t}\right)^{-1}$  von links, so erhält man

$$\dot{\mathbf{k}}(t) = e^{-\mathbf{A}t} \mathbf{b}u(t),$$

woraus sich durch Integration die gesuchte Beziehung für  $\mathbf{k}(t)$

$$\begin{aligned}\int_0^t \dot{\mathbf{k}}(\tau) d\tau &= \mathbf{k}(t) - \mathbf{k}(0) = \int_0^t e^{-\mathbf{A}\tau} \mathbf{b}u(\tau) d\tau \\ \mathbf{k}(t) &= \mathbf{k}(0) + \int_0^t e^{-\mathbf{A}\tau} \mathbf{b}u(\tau) d\tau\end{aligned}$$

ergibt. Nachdem  $\mathbf{k}(t)$  in den Ansatz eingesetzt wurde

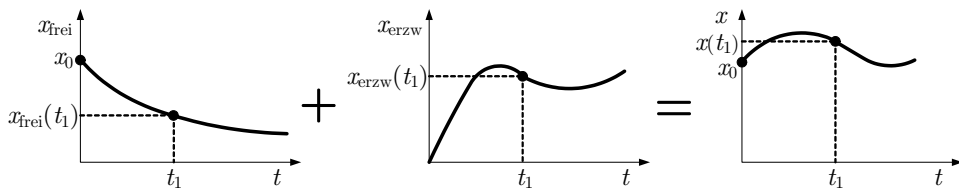
$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{k}(0) + e^{\mathbf{A}t} \int_0^t e^{-\mathbf{A}\tau} \mathbf{b}u(\tau) d\tau,$$

erhält man unter Verwendung der Anfangsbedingung die Lösung der Zustandsgleichung:

Bewegungsgleichung:  $x(t) = e^{At} x_0 + \int_0^t e^{A(t-\tau)} b u(\tau) d\tau.$

(4.5)

Gleichung (4.5) heißt *Bewegungsgleichung* des Systems. Die zweimal in dieser Gleichung vorkommende Matrix  $e^{At}$  wird als *Übergangsmatrix*, *Transitionsmatrix* oder *Fundamentalmatrix* bezeichnet. Die Lösung (4.5) der Zustandsgleichung existiert für beliebige Modellparameter und einen beliebigen Anfangszustand  $x_0$ .



**Abb. 4.2:** Zerlegung der Bewegung  $x(t)$  in die Eigenbewegung  $x_{\text{frei}}(t)$  und die erzwungene Bewegung  $x_{\text{erzw}}(t)$

**Diskussion der Lösung.** Die Bewegung  $x(t)$  setzt sich aus zwei Komponenten zusammen

$$x(t) = x_{\text{frei}}(t) + x_{\text{erzw}}(t), \quad (4.6)$$

die die Eigenbewegung

$$x_{\text{frei}}(t) = e^{At} x_0 \quad (4.7)$$

und die erzwungene Bewegung

$$x_{\text{erzw}}(t) = \int_0^t e^{A(t-\tau)} b u(\tau) d\tau \quad (4.8)$$

des Systems beschreiben (Abb. 4.2). Die Eigenbewegung  $x_{\text{frei}}(t)$  wird durch den Anfangszustand  $x_0$  hervorgerufen. Wird das System nicht durch eine Eingangsgröße erregt ( $u = 0$ ), so tritt ausschließlich diese Komponente der Bewegung auf und es gilt  $x(t) = x_{\text{frei}}(t)$ . Da diese Bewegung das von der Eingangsgröße „freie“ System betrifft, wird sie auch als freie Bewegung bezeichnet. Sie entfällt nur, wenn sich das System zum Zeitpunkt  $t = 0$  in der Ruhelage befindet ( $x_0 = 0$ ).

Die erzwungene Bewegung  $x_{\text{erzw}}(t)$  wird durch die Eingangsgröße  $u(t)$  hervorgerufen. Freie und erzwungene Bewegung überlagern sich additiv, weil das System linear ist. Dies ist für den Zeitpunkt  $t_1$  in Abb. 4.2 für ein System erster Ordnung mit dem skalaren Zustand  $x(t)$  gezeigt.

Die hier angegebene Zerlegung der Bewegung in zwei unabhängige Komponenten für die Eigenbewegung und die erzwungene Bewegung ist nur bei linearen Systemen möglich. Bei nichtlinearen Systemen bestimmen der Anfangszustand  $x_0$  und die Eingangsgröße  $u(t)$  gemeinsam die Bewegung  $x(t)$ , ohne dass man ihre Beiträge wie hier durch zwei unabhängige Summanden angeben könnte.

### 4.2.2 Berechnung der Ausgangsgröße

Mit Hilfe der Ausgabegleichung

$$y(t) = \mathbf{c}^T \mathbf{x}(t) + du(t)$$

kann aus der Lösung der Zustandsgleichung die Ausgangsgröße  $y(t)$  berechnet werden:

Bewegungsgleichung für den Ausgang:

$$y(t) = \mathbf{c}^T e^{\mathbf{A}t} \mathbf{x}_0 + \int_0^t \mathbf{c}^T e^{\mathbf{A}(t-\tau)} \mathbf{b} u(\tau) d\tau + du(t). \quad (4.9)$$

Dies ist eine Gleichung der Form (2.6) auf S. 51, mit der aus dem Anfangszustand  $\mathbf{x}_0$  und der Eingangsgröße  $u(\tau)$  für  $0 \leq \tau \leq t$  die Ausgangsgröße zum Zeitpunkt  $t$  berechnet wird. Der erste Summand beschreibt die Eigenbewegung und der zweite und dritte Summand gemeinsam die erzwungene Bewegung, wobei sich beide Begriffe jetzt auf den Ausgang  $y$  und nicht wie bisher auf den Zustand  $\mathbf{x}$  beziehen und die Summanden deshalb als  $y_{\text{frei}}(t)$  bzw.  $y_{\text{erzw}}(t)$  bezeichnet werden:

$$y(t) = y_{\text{frei}}(t) + y_{\text{erzw}}(t). \quad (4.10)$$

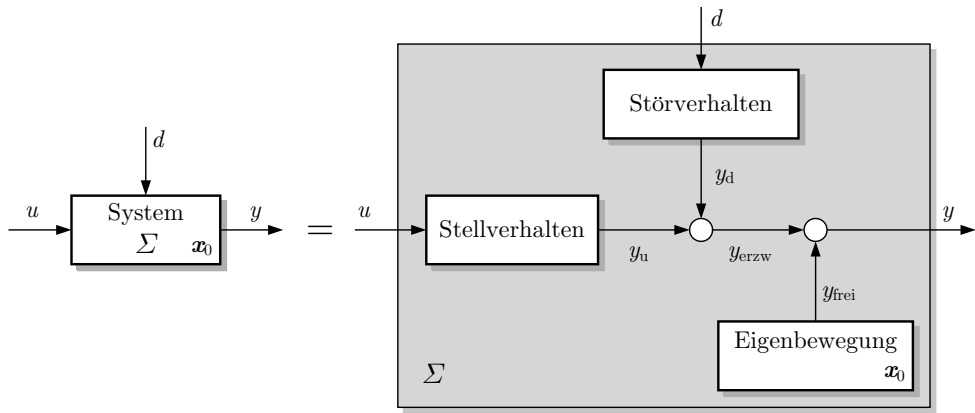
Die Bewegungsgleichung zeigt, dass – wie in der Aufgabenstellung für die Vorhersage des Systemverhaltens im Abschn. 4.1 angegeben – zur Berechnung von  $y_{[0,t_e]}$  der Anfangszustand  $\mathbf{x}_0$  und der Verlauf der Eingangsgröße  $u_{[0,t_e]}$  über den betrachteten Zeitabschnitt  $0 \leq t \leq t_e$  bekannt sein müssen. Das heißt insbesondere, dass ohne die Kenntnis des Anfangszustands keine Vorhersage möglich ist und dass die Vorhersage nur für den Zeitabschnitt erfolgen kann, für den auch die Eingangsgröße bekannt ist. Beide Voraussetzungen bereiten in der Praxis oft Schwierigkeiten. Sie müssen gegebenenfalls durch Schätzwerte für den Anfangszustand und durch Annahmen bezüglich der zukünftigen Eingangsgröße erfüllt werden. Man kann auch Szenarienanalysen durchführen, bei denen man von unterschiedlichen Anfangszuständen ausgeht und das System nacheinander für mehrere mögliche Verläufe der Eingangsgröße untersucht.

Das *Eingangs-Ausgangsverhalten* (E/A-Verhalten) beschreibt die Reaktion eines Systems aus der Ruhelage  $\mathbf{x}_0 = \mathbf{0}$  auf eine Eingangsgröße  $u(t)$ . Entsprechend Gl. (4.9) gilt

E/A-Verhalten:

$$y(t) = \int_0^t \mathbf{c}^T e^{\mathbf{A}(t-\tau)} \mathbf{b} u(\tau) d\tau + du(t).$$

**E/A-Verhalten gestörter Systeme.** Viele Systeme unterliegen nicht nur dem Einfluss einer zielgerichtet vorgegebenen Stellgröße  $u(t)$ , sondern auch einer nicht beeinflussbaren Störgröße  $d(t)$ . Auf Grund der Linearität des Systems überlagern sich beide Wirkungen additiv. Im Zustandsraummodell erhält man einen zusätzlichen Term  $+\mathbf{g}d(t)$ , bei dem der  $n$ -dimensionale Vektor  $\mathbf{g}$  bestimmt, welche Zustandsvariablen  $x_i(t)$  direkt durch die Störung beeinflusst werden und wie groß dieser Einfluss ist:



**Abb. 4.3:** Zerlegung des Systemverhaltens in die Anteile, die durch den Anfangszustand  $\mathbf{x}_0$ , die Stellgröße  $u(t)$  und die Störgröße  $d(t)$  hervorgerufen werden

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}d(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) + du(t) + hd(t). \end{cases} \quad (4.11)$$

Damit erweitert sich die Bewegungsgleichung für den Ausgang  $y(t)$  folgendermaßen:

$$\begin{aligned} y(t) = \mathbf{c}^T \mathbf{e}^{\mathbf{A}t} \mathbf{x}_0 + \int_0^t \mathbf{c}^T \mathbf{e}^{\mathbf{A}(t-\tau)} \mathbf{b}u(\tau) d\tau + du(t) + \\ + \int_0^t \mathbf{c}^T \mathbf{e}^{\mathbf{A}(t-\tau)} \mathbf{g}d(\tau) d\tau + hd(t). \end{aligned} \quad (4.12)$$

Da der üblichen Bezeichnungsweise entsprechend der Parameter  $d$  in der Ausgangsgleichung und die Störung  $d(t)$  mit demselben Symbol gekennzeichnet sind, wird zur Unterscheidung beider Größen stets die Zeitabhängigkeit der Störung angegeben. Für die meisten technischen Systeme gilt  $d = 0$ , so dass in den Gleichungen nur die Störung  $d(t)$  auftritt und keine Verwechslung möglich ist. Der Vektor  $\mathbf{g}$  und der Skalar  $h$  sind zusätzliche Modellparameter. Auf die letzte Gleichung kommt man, wenn man die beiden rechten Summanden in Gl. (4.11) mit  $\tilde{u}(t)$  bezeichnet

$$\tilde{u}(t) = \mathbf{b}u(t) + \mathbf{g}d(t)$$

und  $\tilde{u}(\tau)$  an Stelle von  $\mathbf{b}u(\tau)$  in Gl. (4.9) einsetzt. Der zusätzliche Summand  $hd(t)$  kommt aus der erweiterten Ausgangsgleichung hinzu.

Gleichung (4.12) zeigt, dass sich die Ausgangsgröße  $y(t)$  jetzt aus drei Summanden zusammensetzt

$$y(t) = y_{\text{frei}}(t) + y_u(t) + y_d(t),$$

von denen der erste Summand

$$y_{\text{frei}}(t) = \mathbf{c}^T \mathbf{e}^{\mathbf{A}t} \mathbf{x}_0$$

die Eigenbewegung, der zweite Summand

$$y_u(t) = \int_0^t \mathbf{c}^T \mathbf{e}^{\mathbf{A}(t-\tau)} \mathbf{b} u(\tau) d\tau + d u(t)$$

die durch die Eingangsgröße  $u(t)$  erzwungene Bewegung und der dritte Summand

$$y_d(t) = \int_0^t \mathbf{c}^T \mathbf{e}^{\mathbf{A}(t-\tau)} \mathbf{g} d(\tau) d\tau + h d(t)$$

die durch die Störgröße  $d(t)$  hervorgerufene Bewegung beschreibt. Man kann deshalb das betrachtete System entsprechend Abb. 4.3 in drei Blöcke zerlegen, deren Ausgangsgrößen gerade diese drei Summanden sind. Das Signal  $y_{\text{erzw}}(t)$  beschreibt die durch  $u(t)$  und  $d(t)$  gemeinsam erzwungene Bewegung.

#### Beispiel 4.1 Längsbewegung eines Fahrzeugs

Zur Illustration der bisherigen Ergebnisse wird die Bewegung eines Fahrzeugs in der Ebene und an einem Anstieg untersucht (Abb. 4.4).

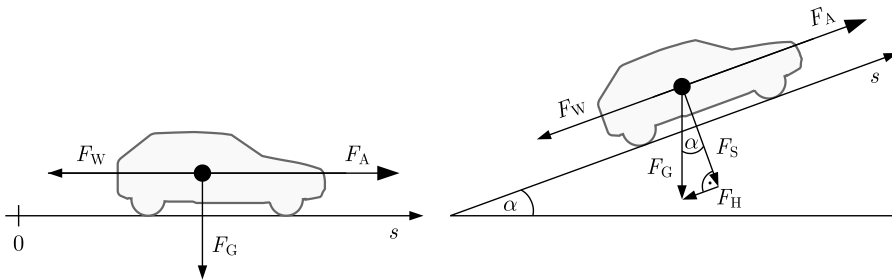


Abb. 4.4: Kräfte, die die Längsbewegung eines Fahrzeugs beeinflussen

**Fahrzeugbewegung in der Ebene.** In der Ebene sind nur die durch den Fahrzeugantrieb erzeugte Kraft  $F_A(t)$  und die durch den Luftwiderstand und die Reibung erzeugte Kraft  $F_W(t)$  von Bedeutung, so dass die Beziehung

$$m\ddot{s}(t) = F_A(t) - F_W(t)$$

gilt, in der  $m$  die Fahrzeugmasse,  $s(t)$  den bis zur Zeit  $t$  zurückgelegten Weg und  $\ddot{s}(t)$  die Fahrzeugbeschleunigung bezeichnen. Es wird angenommen, dass die Kraft  $F_W(t)$  proportional zur Geschwindigkeit  $\dot{s}(t)$  ist, so dass die Beziehung

$$F_W(t) = c\dot{s}(t)$$

mit einer Widerstandskonstanten  $c$  gilt.

Zum Zustandsraummodell kommt man nach Einführung des Zustands  $x(t) = \dot{s}(t)$ , womit man

$$\text{Fahrzeug in der Ebene : } \begin{cases} \dot{x}(t) = -\frac{c}{m}x(t) + \frac{1}{m}u(t), & x(0) = v_0 \\ y(t) = x(t) \end{cases} \quad (4.13)$$

erhält.  $v_0$  bezeichnet die Anfangsgeschwindigkeit,  $u(t)$  die als Eingangsgröße wirkende Antriebskraft  $F_A(t)$ .

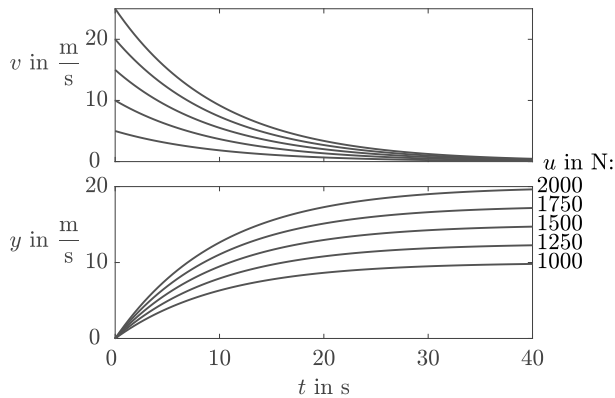
Für eine konstante Eingangsgröße  $u(t) = \bar{u}$  stellt sich die stationäre Geschwindigkeit

$$\bar{y} = \bar{x} = \frac{1}{c}\bar{u} \quad (4.14)$$

ein, die nur von der Widerstandskonstanten  $c$ , aber nicht von der Masse  $m$  abhängt. Die Eigenbewegung des Fahrzeugs ergibt sich entsprechend Gl. (4.7) aus der Beziehung

$$y_{\text{frei}}(t) = x_{\text{frei}}(t) = e^{-\frac{c}{m}t}v_0.$$

Die Geschwindigkeit des nicht angetriebenen Fahrzeugs nimmt von der Anfangsgeschwindigkeit  $v_0$  nach einer e-Funktion ab. Dabei wirkt das Verhältnis  $\frac{m}{c} = T$  von Masse und Widerstandskonstante als eine Zeitkonstante  $T$ . Je größer dieses Verhältnis ist, desto langsamer sinkt die Geschwindigkeit. Nach dem hier verwendeten linearen Modell zu urteilen, kommt das Fahrzeug erst für  $t \rightarrow \infty$  zum Stillstand.



**Abb. 4.5:** Eigenbewegung und erzwungene Bewegung des Fahrzeugs in der Ebene

Abbildung 4.5 (oben) zeigt die Eigenbewegung eines Fahrzeugs mit  $m = 1000 \text{ kg}$  und  $c = 100 \frac{\text{kg}}{\text{s}}$  bei unterschiedlichen Anfangsgeschwindigkeiten. Das Fahrzeug kommt nach 30 bis 40 Sekunden (fast) zum Stillstand, wobei die Fahrzeit von der Anfangsgeschwindigkeit unabhängig ist. Die Zeit von 40 Sekunden ist das Vierfache der Zeitkonstanten

$$T = \frac{m}{c} = \frac{1000 \text{ kg s}}{100 \text{ kg}} = 10 \text{ s}.$$

Für die erzwungene Bewegung erhält man aus Gl. (4.8) die Darstellung

$$y_{\text{erzw}}(t) = x_{\text{erzw}}(t) = \int_0^t e^{-\frac{c}{m}(t-\tau)} \frac{1}{m} u(\tau) d\tau,$$

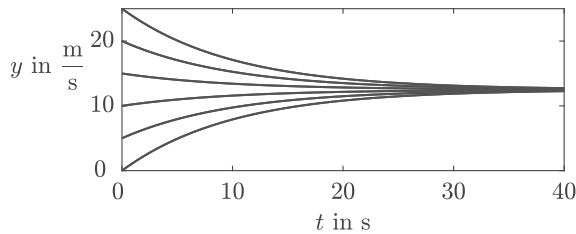


aus der man für die konstante Eingangsgröße  $u(t) = \bar{u}$  die Beziehung

$$\begin{aligned}
 y_{\text{erzw}}(t) &= \int_0^t e^{-\frac{c}{m}(t-\tau)} \frac{1}{m} \bar{u} d\tau \\
 &= e^{-\frac{c}{m}t} \frac{\bar{u}}{m} \int_0^t e^{\frac{c}{m}\tau} d\tau \\
 &= e^{-\frac{c}{m}t} \frac{\bar{u}}{m} \frac{m}{c} e^{\frac{c}{m}\tau} \Big|_0^t \\
 &= \frac{\bar{u}}{c} \left(1 - e^{-\frac{c}{m}t}\right)
 \end{aligned}$$

bekommt. Die Geschwindigkeit steigt entsprechend einer e-Funktion an, bis sie für  $t \rightarrow \infty$  den statischen Endwert  $\frac{\bar{u}}{c}$  erreicht. Dieser Endwert stimmt natürlich mit dem in Gl. (4.14) berechneten Wert überein.

Abbildung 4.5 (unten) zeigt die erzwungene Bewegung für unterschiedliche Amplituden  $\bar{u}$  der Eingangsgröße. Die statischen Endwerte der Bewegung werden unabhängig von  $\bar{u}$  nach etwa 40 Sekunden erreicht, also in derselben Zeitspanne, in der auch die Eigenbewegung abklingt.



**Abb. 4.6:** Geschwindigkeit des Fahrzeugs mit unterschiedlicher Anfangsgeschwindigkeit und konstanter Eingangsgröße

Bewegt sich das Fahrzeug ausgehend von einer Anfangsgeschwindigkeit  $v_0$  unter der Wirkung der Antriebskraft  $u(t)$ , so wird seine Bewegung durch die Addition der Eigenbewegung und der freien Bewegung beschrieben:

$$y(t) = x(t) = e^{-\frac{c}{m}t} v_0 + \int_0^t e^{-\frac{c}{m}(t-\tau)} \frac{1}{m} u(\tau) d\tau.$$

Abbildung 4.6 zeigt die so berechnete Fahrzeuggeschwindigkeit bei unterschiedlicher Anfangsgeschwindigkeit unter dem Einfluss einer konstanten Eingangsgröße, deren Amplitude in allen Fällen gleich groß ist. Die Bewegung beginnt bei der Anfangsgeschwindigkeit und endet in allen Fällen bei einer konstanten Geschwindigkeit, die nur von der Eingangsgröße, aber nicht von der Anfangsgeschwindigkeit  $v_0$  abhängt, denn für große Zeiten  $t$  hat die Eigenbewegung keinen Einfluss mehr auf die aktuelle Geschwindigkeit des Fahrzeugs und die Bewegung besteht (näherungsweise) nur noch aus der erzwungenen Bewegung.

**Fahrzeugbewegung an einem Anstieg.** Die Hangabtriebskraft  $F_H(t)$  an einem Anstieg hängt vom Neigungswinkel  $\alpha(t)$  ab und kann für kleine Winkel durch die Beziehung

$$F_H(t) = F_G \alpha(t)$$

angenähert werden, wobei  $F_G$  die Gewichtskraft

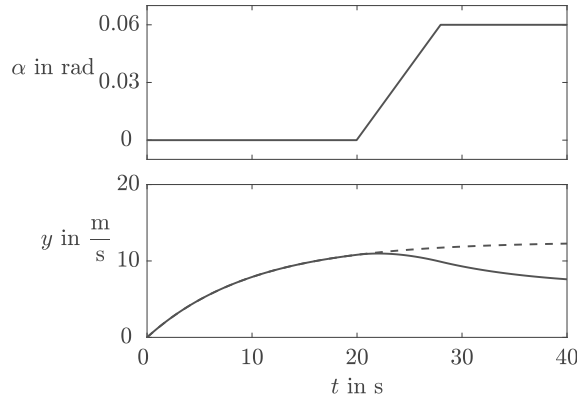
$$F_G = mg$$

und  $g$  die Erdbeschleunigung bezeichnet (Abb. 4.4 rechts). Damit erhält man als neues Modell die Beziehungen

$$\text{Fahrzeug am Anstieg : } \begin{cases} \dot{x}(t) = -\frac{c}{m}x(t) + \frac{1}{m}u(t) - gd(t), & x(0) = v_0 \\ y(t) = x(t), \end{cases} \quad (4.15)$$

in der die erste Gleichung gegenüber Gl. (4.13) um den Term  $-gd(t)$  erweitert ist, wobei  $d(t) = \alpha(t)$  der als Störgröße wirkende Neigungswinkel der Fahrbahn ist. In der Bewegungsgleichung tritt jetzt ein zusätzlicher Summand auf, der von der Störgröße abhängt:

$$y(t) = x(t) = e^{-\frac{c}{m}t}v_0 + \int_0^t e^{-\frac{c}{m}(t-\tau)} \frac{1}{m}u(\tau) d\tau - \int_0^t e^{-\frac{c}{m}(t-\tau)}gd(\tau) d\tau.$$



**Abb. 4.7:** Neigungswinkel  $\alpha(t)$  des Anstiegs und Fahrzeuggeschwindigkeit  $y(t)$  bei konstanter Eingangsgröße

Die in Abb. 4.7 dargestellte Bewegung des Fahrzeugs auf einer Fahrbahn mit unterschiedlicher Steigung beginnt in der Ebene ( $\alpha(t) = 0$ ) aus dem Stillstand. Bis zur Zeit 20 Sekunden ist der Anstiegswinkel gleich null, so dass die Geschwindigkeit durch die erzwungene Bewegung bestimmt ist, die mit der aus Abb. 4.5 (unten) für dieselbe Eingangsgröße  $u(t) = 1250$  N übereinstimmt. Im Zeitintervall zwischen 20 und 28 Sekunden nimmt die Steigung zu und bleibt anschließend konstant bei 6%. Die Wirkung dieser Störung ist ein Geschwindigkeitsabfall, dessen statischer Endwert erst nach dem hier gezeigten Zeitintervall erreicht wird. Die gestrichelte Kurve zeigt den Geschwindigkeitsverlauf in der Ebene. Die Differenz zwischen gestrichelter und durchgezogener Kurve beschreibt die Wirkung der Störung (Steigung).  $\square$

### 4.2.3 Anmerkungen zum Zustandsbegriff

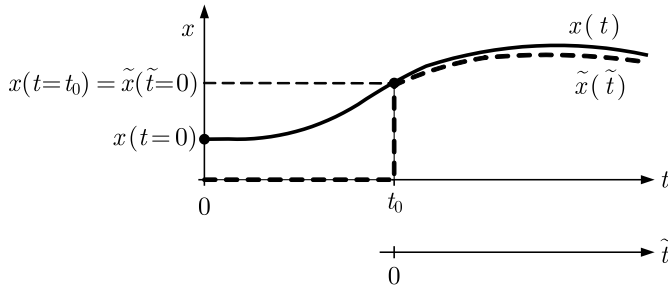
Der Systemzustand  $x(t)$  wurde in Definition 3.1 auf S. 94 als die Zusammenfassung derjenigen Informationen eingeführt, die für die eindeutige Vorhersage des zukünftigen Systemverhaltens notwendig sind. Mit Hilfe der Bewegungsgleichung kann man diesen Sachverhalt verdeutlichen (Abb. 4.8). Zur Vereinfachung der Betrachtungen wird zunächst angenommen, dass das System keine Eingangsgröße hat

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0$$

und folglich die vereinfachte Bewegungsgleichung

$$x(t) = e^{At}x_0, \quad t \geq 0 \quad (4.16)$$

gilt. Gleichung (4.16) zeigt, dass man aus der Kenntnis des Anfangszustands  $x_0$  den Zustand  $x(t)$  für jeden zukünftigen Zeitpunkt  $t \geq 0$  berechnen kann.



**Abb. 4.8:** Berechnung der Zustandstrajektorie für zwei unterschiedliche Zeitachsen

Mit derselben Argumentation kann man die Zustandstrajektorie für das Zeitintervall  $t_0 \leq t < \infty$  berechnen, wenn man den Zustand zur Zeit  $t_0$  kennt. Dafür muss man nur eine neue Zeitachse für die Zeit  $\tilde{t}$  einführen, deren Nullpunkt bei der Zeit  $t_0$  der  $t$ -Achse liegt (Abb. 4.8). Für diese Zeitachse gilt dann

$$x(\tilde{t}) = e^{A\tilde{t}}x(0), \quad \tilde{t} \geq 0,$$

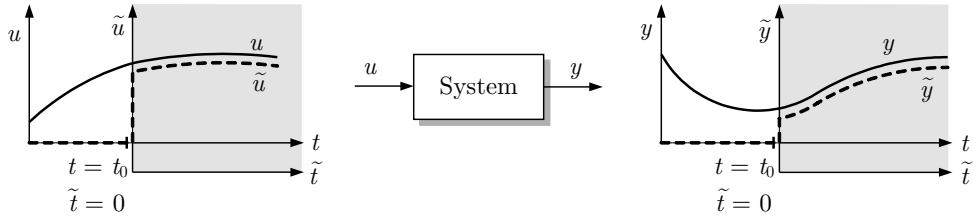
wobei  $x(0)$  jetzt den Zustand zum Zeitpunkt  $\tilde{t} = 0$  bezeichnet. Um die über unterschiedliche Zeitachsen definierten Signale zu unterscheiden, wird im Folgenden für den Zustand  $\tilde{x}$  geschrieben, wenn die  $\tilde{t}$ -Zeitachse gemeint ist

$$\tilde{x}(\tilde{t}) = e^{A\tilde{t}}\tilde{x}_0, \quad \tilde{t} \geq 0,$$

wobei  $\tilde{x}_0 = \tilde{x}(\tilde{t}=0)$  ist.

Diese Betrachtungen zeigen, dass man den Zustand  $x$  zu einem beliebigen Zeitpunkt  $t_0$  kennen muss, um die Trajektorie für das nach diesem Zeitpunkt liegende Zeitintervall berechnen zu

können. Bei Kenntnis des Zustands zum Zeitpunkt  $t_0$  braucht man die Bewegung des Systems im Zeitintervall  $-\infty < t < t_0$  nicht zu kennen, um die Bewegung für  $t \geq t_0$  zu berechnen. Diese Vereinfachung kann man bei Vorhersageaufgaben anwenden, wenn die Zeitspanne, die zwischen dem Anfangszustand und dem aktuell zu berechnenden Zustand liegt, nicht zu groß werden soll.



**Abb. 4.9:** Darstellung des E/A-Verhaltens mit unterschiedlichen Zeitachsen

Für Systeme mit dem Eingang  $u(t)$  bedeutet die Kenntnis des Zustands zum Zeitpunkt  $t_0$  nicht nur, dass man die vorherige Bewegung des Zustands  $x$  ignorieren kann, wenn man sich für das zukünftige Verhalten ( $t > t_0$ ) interessiert, sondern auch, dass die Eingangsgröße für das Zeitintervall  $-\infty < t \leq t_0$  unbedeutend ist. Diese Tatsache wird in Abb. 4.9 veranschaulicht. Die Signale  $u$  und  $y$  beziehen sich auf die Zeit  $t$ . Wird eine neue Zeitachse  $\tilde{t}$  eingeführt, bezüglich derer der Eingang jetzt mit  $\tilde{u}$  und der Ausgang mit  $\tilde{y}$  bezeichnet wird, so spielen die Werte der Signale  $u$  und  $y$ , die vor dem Nullpunkt der  $\tilde{t}$ -Zeitachse aufgetreten sind, für die Berechnung des E/A-Verhaltens für  $\tilde{t} > 0$  keine Rolle, denn ihr Einfluss wird durch den Zustand  $x(\tilde{t}=0)$  wiedergegeben. Entsprechen sich die Zeitpunkte

$$t = t_0 \quad \text{und} \quad \tilde{t} = 0,$$

so gilt für  $\tilde{t} > 0$

$$u(\tilde{t} + t_0) = \tilde{u}(\tilde{t}) \quad \text{und} \quad y(\tilde{t} + t_0) = \tilde{y}(\tilde{t}). \quad (4.17)$$

**Herleitung der Beziehung (4.17).** Betrachtet man zunächst die Zustandstrajektorie, so gilt für die  $t$ -Achse die Bewegungsgleichung (4.5)

$$\dot{x}(t) = A t x_0 + \int_0^t e^{A(t-\tau)} b u(\tau) d\tau.$$

In Analogie dazu gilt für  $\tilde{x}$  bezüglich der  $\tilde{t}$ -Zeitachse die Beziehung

$$\dot{\tilde{x}}(\tilde{t}) = A \tilde{t} \tilde{x}_0 + \int_0^{\tilde{t}} e^{A(\tilde{t}-\tilde{\tau})} b \tilde{u}(\tilde{\tau}) d\tilde{\tau},$$

wobei  $\tilde{x}_0$  den Zustand zum Zeitpunkt  $\tilde{t} = 0$  darstellt.  $\tilde{x}_0$  kann man aus der ersten Gleichung wie folgt berechnen:

$$\tilde{x}_0 = x(t_0) = e^{A t_0} x_0 + \int_0^{t_0} e^{A(t_0 - \tau)} b u(\tau) d\tau. \quad (4.18)$$

Für die Eingangsgröße gilt die linke Seite von Gl. (4.17) nach Voraussetzung. Es ist zu zeigen, dass die analoge Beziehung auch für den Zustand richtig ist, woraus dann die rechte Seite von Gl. (4.17) folgt. Dafür berechnet man aus der Bewegungsgleichung (4.5) den Zustand zum Zeitpunkt  $\tilde{t} + t_0$

$$\begin{aligned} x(\tilde{t} + t_0) &= e^{A(\tilde{t} + t_0)} x_0 + \int_0^{\tilde{t} + t_0} e^{A(\tilde{t} + t_0 - \tau)} b u(\tau) d\tau \\ &= e^{A\tilde{t}} e^{A t_0} x_0 + \int_0^{t_0} e^{A(\tilde{t} + t_0 - \tau)} b u(\tau) d\tau + \int_{t_0}^{\tilde{t}} e^{A(\tilde{t} + t_0 - \tau)} b u(\tau) d\tau \\ &= e^{A\tilde{t}} \left( e^{A t_0} x_0 + \int_0^{t_0} e^{A(t_0 - \tau)} b u(\tau) d\tau \right) + \int_{t_0}^{\tilde{t}} e^{A(\tilde{t} + t_0 - \tau)} b u(\tau) d\tau \\ &= e^{A\tilde{t}} x(t_0) + \int_0^{\tilde{t} - t_0} e^{A(\tilde{t} - \tilde{\tau})} b u(\tilde{\tau} + t_0) d\tilde{\tau}, \end{aligned}$$

wobei die letzte Gleichung durch Einsetzen des Zustands  $x(t_0)$  aus Gl. (4.18) und durch Substitution  $\tilde{\tau} = \tau - t_0$  hervorging. Da  $u(\tilde{\tau} + t_0) = \tilde{u}(\tilde{\tau})$  gilt, besteht zwischen  $x$  und  $\tilde{x}$  tatsächlich der o. a. Zusammenhang  $x(\tilde{t} + t_0) = \tilde{x}(\tilde{t})$ , womit die rechte Seite der Beziehung (4.17) nachgewiesen ist.

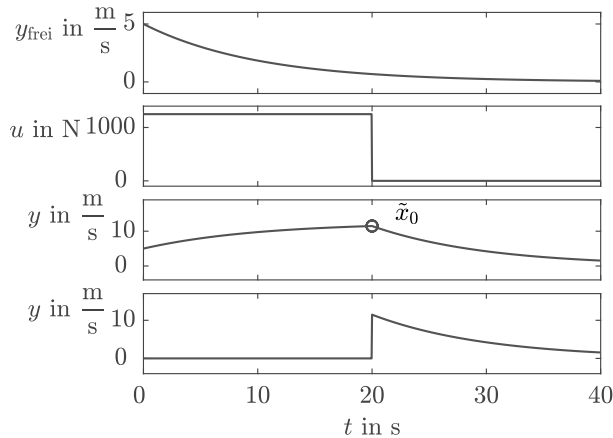
Die Bedeutung des Zustands als Sammlung aller Informationen über die vergangene Bewegung des Systems, die für die Vorhersage der zukünftigen Bewegung notwendig sind, gilt auch für nichtlineare Systeme. Auch für diese Systeme kann man bei Kenntnis von  $x(t_0)$  die Bewegung bis zum Zeitpunkt  $t_0$  „vergessen“, wenn man sich nur für das Systemverhalten für  $t \geq t_0$  interessiert. Im Unterschied zu linearen Systemen gibt es jedoch i. Allg. keine mit Gl. (4.16) vergleichbare explizite Darstellung für den Zusammenhang zwischen  $x_0$  und  $x(t)$ . Auch überlagern sich Eigenbewegung und erzwungene Bewegung nicht mehr additiv.

#### Beispiel 4.1 (Forts.) Längsbewegung eines Fahrzeugs

Die abgeleiteten Beziehungen kann man sich bei der Berechnung der Fahrzeugbewegung zu Nutze machen. Das Fahrzeug startet mit der Anfangsgeschwindigkeit  $v_0 = 5 \frac{m}{s}$  und erhält die in der zweiten Zeile von Abb. 4.10 gezeigte Eingangsgröße  $u(t)$ . Der dabei entstehende Geschwindigkeitsverlauf ist in der dritten Zeile der Abbildung zu sehen. Er wurde mit der Bewegungsgleichung (4.9) berechnet und berücksichtigt sowohl die freie Bewegung als auch die Eigenbewegung.

Obwohl die Eingangsgröße für  $t > 20$  s verschwindet, geht sie in die Berechnung von  $y(t)$  für  $t > 20$  s ein, denn das Integral in der Bewegungsgleichung hängt auch von zurückliegenden Werten der Eingangsgröße ab.

Wenn man allerdings die Fahrzeuggeschwindigkeit zur Zeit  $t = 20$  s kennt, kann man die Bewegung des Fahrzeugs für das Zeitintervall  $t \geq 20$  s als eine Eigenbewegung interpretieren. Es ist allerdings nicht die Eigenbewegung zum Anfangszustand  $x(0) = v_0$ , sondern die Eigenbewegung, die



**Abb. 4.10:** Berechnung der Fahrzeugbewegung mit Hilfe des Zustands zum Zeitpunkt  $t_0 = 20$

zum Zustand  $\tilde{x}_0 = x(t = 20)$  gehört, der in der Abbildung durch einen Kreis gekennzeichnet ist. Da bei dem hier verwendeten Modell erster Ordnung Zustand und Ausgang übereinstimmen, ist dieser „Anfangszustand“ gerade der Wert der Ausgangsgröße, der zum Zeitpunkt  $t = 20$  s auftritt, also zu dem Zeitpunkt, von dem ab die Eingangsgröße verschwindet. In der unteren Zeile der Abbildung ist diese Eigenbewegung aufgetragen. Sie stimmt mit dem Verlauf der Ausgangsgröße aus dem darüber liegenden Diagramm für den Zeitraum  $t \geq 20$  überein.

Das Beispiel zeigt, wie man durch Nutzung von „Zwischenzuständen“ die Berechnung des Systemverhaltens vereinfachen kann.  $\square$

#### 4.2.4 Bewegungsgleichung in kanonischer Form

Eine besonders einfache Form erhält man für die Bewegungsgleichung, wenn die Systemmatrix  $A$  diagonalähnlich ist, was insbesondere dann der Fall ist, wenn alle Eigenwerte dieser Matrix einfach sind. Dann kann  $A$  durch die Ähnlichkeitstransformation

$$V^{-1}AV = \begin{pmatrix} \lambda_1 & & \\ & \lambda_2 & \\ & & \ddots \\ & & & \lambda_n \end{pmatrix} = \text{diag } \lambda_i$$

in die Diagonalmatrix  $\text{diag } \lambda_i$  überführt werden, in deren Hauptdiagonale die Eigenwerte  $\lambda_1, \lambda_2, \dots, \lambda_n$  stehen. Die  $(n \times n)$ -Matrix

$$V = (v_1 \ v_2 \ \dots \ v_n)$$

besteht aus den  $n$  Eigenvektoren  $v_i$  von  $A$ , die in derselben Reihenfolge stehen wie die zugehörigen Eigenwerte  $\lambda_i$ .

Mit der Transformationsmatrix  $V$  kann auch die Matrixexponentialfunktion  $e^{At}$  in eine Diagonalform gebracht werden

$$V^{-1}e^{At}V = \begin{pmatrix} e^{\lambda_1 t} & & & \\ & e^{\lambda_2 t} & & \\ & & \ddots & \\ & & & e^{\lambda_n t} \end{pmatrix} = \text{diag } e^{\lambda_i t},$$

wie man sich anhand der Definition (4.4) überlegen kann. Dann erhält die Bewegungsgleichung (4.5) die einfachere Form:

$$\begin{aligned} \mathbf{x}(t) &= e^{At} \mathbf{x}_0 + \int_0^t e^{A(t-\tau)} \mathbf{b} u(\tau) d\tau \\ &= VV^{-1}e^{At}VV^{-1} \mathbf{x}_0 + \int_0^t VV^{-1}e^{A(t-\tau)}VV^{-1} \mathbf{b} u(\tau) d\tau \\ &= V \text{diag } e^{\lambda_i t} \underbrace{V^{-1}\mathbf{x}_0}_{\tilde{\mathbf{x}}_0} + \int_0^t V \text{diag } e^{\lambda_i(t-\tau)} \underbrace{V^{-1}\mathbf{b}}_{\tilde{\mathbf{b}}} u(\tau) d\tau \end{aligned}$$

und

Bewegungsgleichung in kanonischer Darstellung:

$$\mathbf{x}(t) = \sum_{i=1}^n \mathbf{v}_i e^{\lambda_i t} \tilde{x}_{i0} + \sum_{i=1}^n \mathbf{v}_i \int_0^t e^{\lambda_i(t-\tau)} \tilde{b}_i u(\tau) d\tau.$$

(4.19)

Im letzten Umformungsschritt wurden die Vektoren  $\tilde{\mathbf{x}}_0$  und  $\tilde{\mathbf{b}}$  in ihre Komponenten  $\tilde{x}_{i0}$  bzw.  $\tilde{b}_i$  zerlegt. Gleichung (4.19) wird als *Bewegungsgleichung in kanonischer Darstellung* bezeichnet. Die Funktionen  $e^{\lambda_i t}$  werden *Modi* oder *Eigenvorgänge* des Systems genannt. Wie stark diese Funktionen in die einzelnen Komponenten des Zustandsvektors  $\mathbf{x}$  eingehen, wird durch die Vektoren  $\mathbf{v}_i$  bestimmt.

Für die Ausgangsgröße erhält man

$$y(t) = \sum_{i=1}^n \mathbf{c}^T \mathbf{v}_i e^{\lambda_i t} \tilde{x}_{i0} + \sum_{i=1}^n \mathbf{c}^T \mathbf{v}_i \int_0^t e^{\lambda_i(t-\tau)} \tilde{b}_i u(\tau) d\tau + du(t). \quad (4.20)$$

**Diskussion.** Entsprechend den Gln. (4.19) und (4.20) lassen sich die Eigenbewegungen des Systems bezüglich des Zustands  $\mathbf{x}(t)$  und der Ausgangsgröße  $y(t)$  durch die Modi  $e^{\lambda_i t}$  beschreiben, die auch in den Integralen vorkommen, mit denen die erzwungene Bewegung berechnet wird. Wie die folgende Fallunterscheidung zeigt, sind deshalb die Realteile der Eigenwerte  $\lambda_i$  der Matrix  $A$  für das Systemverhalten entscheidend:

- Haben alle Eigenwerte  $\lambda_i$  von  $\mathbf{A}$  einen negativen Realteil

$$\operatorname{Re}\{\lambda_i\} < 0, \quad i = 1, 2, \dots, n, \quad (4.21)$$

so klingt die Eigenbewegung

$$\mathbf{x}_{\text{frei}}(t) = \sum_{i=1}^n \mathbf{v}_i e^{\lambda_i t} \tilde{x}_{i0} \xrightarrow{t \rightarrow \infty} \mathbf{0}$$

ab, d. h., das ungestörte System ( $u(t) = 0$ ) nähert sich asymptotisch seiner Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$ .

- Hat wenigstens ein Eigenwert einen positiven Realteil

$$\operatorname{Re}\{\lambda_i\} > 0,$$

so wächst mindestens eine Zustandsvariable  $\tilde{x}_i(t)$  für  $t \rightarrow \infty$  über alle Grenzen, wenn  $\tilde{x}_{i0} \neq 0$ , ( $i = 1, 2, \dots, n$ ) ist. Das ungestörte System kehrt nicht aus jeder Anfangsauslenkung  $\mathbf{x}_0$  in die Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  zurück.

Diese Fallunterscheidung wird bei der Stabilitätsanalyse im Kap. 6 ausführlich untersucht. Ein System soll jedoch bereits jetzt als *stabil* (bzw. genauer als asymptotisch stabil) bezeichnet werden, wenn es die Bedingung (4.21) erfüllt.

**Zeitkonstante und dominierender Eigenwert.** Die Eigenvorgänge  $e^{\lambda_i t}$ , ( $i = 1, 2, \dots, n$ ) überlagern sich in der in Gl. (4.20) angegebenen Summe. Bei stabilen Systemen klingen alle Eigenvorgänge ab und tragen nur über jeweils eine begrenzte Zeitspanne wesentlich zum Wert der Ausgangsgröße bei. Um diese Zeitspanne abzuschätzen, werden die Eigenvorgänge mit negativen reellen Eigenwerten unter Verwendung der zu dem entsprechenden Eigenwert  $-\lambda$  gehörenden *Zeitkonstanten*

$$T = \frac{1}{\lambda} \quad (4.22)$$

dargestellt, wobei

$$e^{-\lambda t} = e^{-\frac{t}{T}}$$

gilt. Wenn man jetzt den Zeitpunkt  $t$  betrachtet, der dreimal so groß wie die Zeitkonstante  $T$  ist, so gilt

$$e^{-\frac{3T}{T}} = e^{-3} = 0,05,$$

was bedeutet, dass nach Ablauf der dreifachen Zeitkonstante ein Eigenvorgang nur noch 5% der ursprünglichen Amplitude besitzt. Man kann also erwarten, dass ein Eigenvorgang mit der Zeitkonstanten  $T$  für Zeitpunkte  $t > 3T$  keinen wesentlichen Einfluss mehr auf den Systemausgang ausübt.

Bei konjugiert-komplexen Eigenwerten  $\lambda_{1/2} = -\delta \pm j\omega$  mit negativem Realteil ( $\delta > 0$ ) kann man eine ähnliche Überlegung anstellen, indem man die beiden zu diesen Eigenwerten gehörenden Eigenvorgänge zu

$$e^{\lambda_1 t} + e^{\lambda_2 t} = e^{-\delta + j\omega t} + e^{-\delta - j\omega t} = e^{-\delta t} (e^{j\omega t} + e^{-j\omega t})$$



zusammenfasst und die Amplitude entsprechend

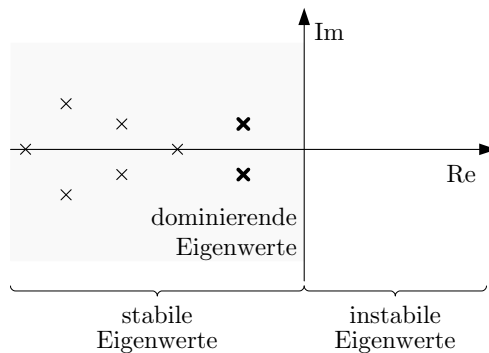
$$|e^{-\delta t} (e^{j\omega t} + e^{-j\omega t})| = e^{-\delta t} \cdot |e^{j\omega t} + e^{-j\omega t}| \leq 2e^{-\delta t}$$

nach oben abschätzt. Dann kann man mit  $\delta$  an Stelle von  $\lambda$  die zuvor gemachten Überlegungen wiederholen und kommt entsprechend

$$\begin{aligned} t = 3T : \quad 2e^{-\frac{3T}{T}} &= e^{-3} = 0,10 \\ t = 3,7T : \quad 2e^{-\frac{3,7T}{T}} &= e^{-3,7} = 0,05 \end{aligned}$$

zu dem Ergebnis, dass auch die zu konjugiert-komplexen Eigenwerten gehörenden Eigenvorgänge nach der dreifachen Zeitkonstante  $T = \frac{1}{\delta}$  im Wesentlichen abgeklungen sind.

Diese Betrachtungen zeigen, dass jeder stabile Eigenvorgang nur über ein begrenztes Zeitintervall einen wesentlichen Einfluss auf den Systemausgang hat und dass dieses Zeitintervall vom Realteil des zugehörigen Eigenwertes abhängt. Die Eigenwerte mit den betragsmäßig kleinsten Realteilen bestimmen das Systemverhalten am längsten. Sie werden auch als die *dominierenden Eigenwerte* bezeichnet, wobei man mit diesem Begriff häufig die Vorstellung verknüpft, dass aus den Eigenwerten eines Systems ein Eigenwert oder ein konjugiert-komplexes Eigenwertpaar insofern herausragt, als dass dieser Eigenwert bzw. dieses Eigenwertpaar einen betragsmäßig wesentlich kleineren Realteil hat als alle anderen Eigenwerte.

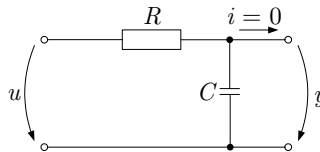


**Abb. 4.11:** Darstellung der Eigenwerte in der komplexen Ebene

Zur grafischen Veranschaulichung dieser Tatsache markiert man die Eigenwerte des betrachteten Systems durch Kreuze in der komplexen Ebene. Entsprechend Gl. (4.21) ist die Imaginärachse die Stabilitätsgrenze, wobei alle Eigenwerte eines stabilen Systems in der linken komplexen Halbebene liegen. Die dominierenden Eigenwerte sind dann die der Imaginärachse am nächsten liegenden Eigenwerte (Abb. 4.11).

**Aufgabe 4.1** Verhalten eines RC-Gliedes

Bei dem in Abb. 4.12 gezeigten RC-Glied ist die an den linken Klemmen anliegende Spannung die Eingangsgröße  $u(t)$  und die Spannung über dem Kondensator die Ausgangsgröße  $y(t)$ . Die rechten Klemmen sind nicht belastet, so dass der in der Abbildung eingetragene Strom  $i(t)$  verschwindet.

**Abb. 4.12:** RC-Glied

1. Stellen Sie das Zustandsraummodell für das RC-Glied auf, wobei Sie die Spannung über dem Kondensator als Zustandsgröße verwenden.
2. Schreiben Sie die Bewegungsgleichung für das RC-Glied auf.
3. Berechnen Sie die Eigenbewegung des RC-Gliedes, wenn der Kondensator zur Zeit  $t = 0$  die Spannung  $u_{C0}$  hat und die linken Klemmen kurzgeschlossen sind ( $u(t) = 0$ ). Skizzieren Sie den Verlauf der Ausgangsgröße.
4. Wie verläuft die Spannung, wenn der zur Zeit  $t = 0$  spannungslose Kondensator aufgeladen wird, wobei die Eingangsspannung auf dem Wert  $u(t) = \bar{u}$  konstant gehalten wird? Wie groß ist die statische Verstärkung des RC-Gliedes?
5. Skizzieren Sie die Ausgangsgröße für den Fall, dass der Kondensator ausgehend von der Spannung  $y(0) = u_{C0}$  bei einer konstanten Eingangsspannung  $u(t) = 2u_{C0}$  weiter aufgeladen wird.
6. Stellen Sie mit Hilfe von MATLAB die Ausgangsgröße für folgende Vorgaben grafisch dar und diskutieren Sie die Abhängigkeit des Verlaufs der Ausgangsgröße von diesen Parametern:

$u_{C0}$	$R$	$C$	$\bar{u}$
0,5 V	10 k $\Omega$	1 $\mu$ F	1 V
0 V	10 k $\Omega$	1 $\mu$ F	1 V
0 V	10 k $\Omega$	1 $\mu$ F	2 V
0 V	8 k $\Omega$	1 $\mu$ F	2 V
0 V	500 $\Omega$	1 $\mu$ F	2 V

□

**Aufgabe 4.2\*** Temperaturverhalten eines Rührkesselreaktors

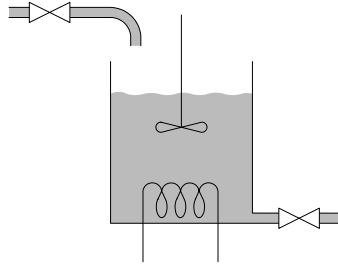
Der in Abb. 4.13 gezeigte Rührkesselreaktor wird in einem Batchprozess genutzt, um eine Flüssigkeit zu erhitzen. Bei geschlossenen Ventilen im Zulauf und im Ablauf wird für das Temperaturverhalten des Reaktors die Differentialgleichung

$$\frac{dT(t)}{dt} = aT(t) + bu(t), \quad T(0) = T_0 \quad (4.23)$$

aufgestellt, deren Parameter  $a$  und  $b$  sich entsprechend

$$a = -\frac{\alpha A}{mc} \quad \text{und} \quad b = \frac{1}{mc}$$

aus der Oberfläche  $A$  des Reaktorgefäßes, dem Wärmeübergangskoeffizient  $\alpha$  zwischen Reaktor und Umgebung, der Masse  $m$  der eingefüllten Flüssigkeit sowie der spezifischen Wärmekapazität  $c$  der Flüssigkeit ergeben. Die Eingangsgröße  $u(t)$  ist die dem Reaktor durch die Heizung pro Zeiteinheit zugeführte Wärmemenge, die Ausgangsgröße die Temperatur  $T(t)$ .



**Abb. 4.13:** Geheizter Rührkesselreaktor

Die Heizleistung des Reaktors wird folgendermaßen festgelegt: Im Intervall  $0 \leq t < t_1$  hat die Eingangsgröße  $u(t)$  den Wert  $u_1$  und im Intervall  $t_1 \leq t < t_2$  den Wert  $u_2$ . Für  $t \geq t_2$  ist die Heizung abgeschaltet.

1. Bestimmen Sie den Temperaturverlauf für die beschriebene Eingangsgröße und den Anfangswert  $T_0 = 0$ .
2. Unter welcher Bedingung bleibt die Temperatur im Zeitintervall  $t_1 \leq t < t_2$  konstant?
3. Zerlegen Sie die Bewegungsgleichung für die Temperatur so, dass Sie den Einfluss der in den beiden Zeitintervallen dem Reaktor zugeführten Wärmemengen auf das Temperaturverhalten im Zeitintervall  $t > t_2$  erkennen können. Interpretieren Sie Ihr Ergebnis.  $\square$

## 4.2.5 Übergangsfunktion

Die Übergangsfunktion beschreibt die Reaktion des Systems auf eine sprungförmige Veränderung der Eingangsgröße, wenn sich das System zur Zeit  $t = 0$  in der Ruhelage  $x_0 = 0$  befindet (Abb. 4.14).

Die sprungförmige Eingangsgröße wird durch die *Sprungfunktion*

$$\sigma(t) = \begin{cases} 0 & \text{für } t < 0 \\ 1 & \text{für } t \geq 0 \end{cases} \quad (4.24)$$

dargestellt:

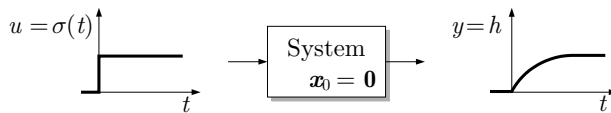
$$u(t) = \sigma(t). \quad (4.25)$$

Die Systemantwort  $y(t)$  erhält man aus Gl. (4.9), wobei auf Grund des verschwindenden Anfangszustands die Eigenbewegung verschwindet. Die Übergangsfunktion wird mit  $h(t)$  bezeichnet:

Übergangsfunktion:  $h(t) = \int_0^t \mathbf{c}^T \mathbf{e}^{\mathbf{A}\tau} \mathbf{b} \, d\tau + d.$

(4.26)

Als zweiter Summand wurde  $+d$  an Stelle von  $+d\sigma(t)$  geschrieben, weil vereinbarungsgemäß alle Signale, also auch  $h(t)$ , für  $t < 0$  verschwinden und Gl. (4.26) den Wert von  $h(t)$  nur für  $t \geq 0$  bestimmt.



**Abb. 4.14:** System mit sprungförmiger Eingangsgröße

Wenn  $\det \mathbf{A} \neq 0$  gilt, kann die Übergangsfunktion weiter ausgerechnet werden. Das Integral führt auf

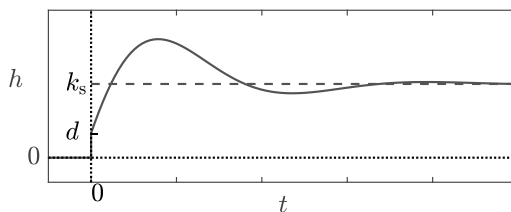
$$\begin{aligned} h(t) &= \mathbf{c}^T \mathbf{A}^{-1} \mathbf{e}^{\mathbf{A}\tau} \mathbf{b} \Big|_{\tau=0}^{\tau=t} + d\sigma(t) \\ &= \mathbf{c}^T \mathbf{A}^{-1} \mathbf{e}^{\mathbf{A}t} \mathbf{b} - \mathbf{c}^T \mathbf{A}^{-1} \mathbf{b} + d\sigma(t) \end{aligned} \quad (4.27)$$

und nach Weglassen von  $\sigma(t)$  auf

$$h(t) = d - \mathbf{c}^T \mathbf{A}^{-1} \mathbf{b} + \mathbf{c}^T \mathbf{A}^{-1} \mathbf{e}^{\mathbf{A}t} \mathbf{b}. \quad (4.28)$$

Die Zeitabhängigkeit der Übergangsfunktion entsteht durch den dritten Summanden. Wie die kanonische Darstellung der Bewegungsgleichung gezeigt hat, ist dieser Summand bei diagonalähnlichen Matrizen  $\mathbf{A}$  eine Summe der  $n$  e-Funktionen  $e^{\lambda_i t}$ , ( $i = 1, 2, \dots, n$ ).

Wenn  $\det \mathbf{A} = 0$  gilt, besitzt die Matrix  $\mathbf{A}$  einen oder mehrere verschwindende Eigenwerte. Dann führt die Lösung des Integrals in Gl. (4.26) auf Terme, in denen außer den Exponentialfunktionen  $e^{\lambda_i t}$ , ( $i = 1, 2, \dots, n$ ) auch Produkte von e-Funktionen mit  $t$ ,  $t^2$ ,  $t^3$  usw. vorkommen.



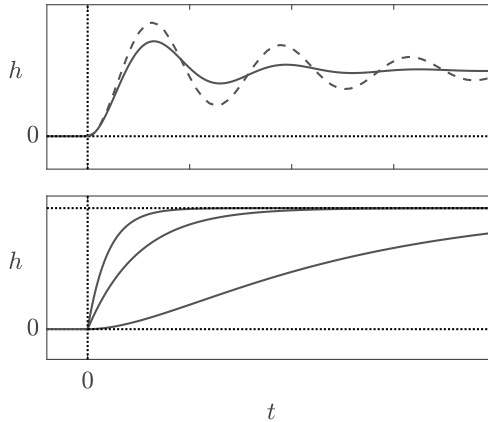
**Abb. 4.15:** Übergangsfunktion eines Systems zweiter Ordnung

Abbildung 4.15 zeigt den typischen Verlauf der Übergangsfunktion stabiler Systeme. Für große Zeit  $t$  nähert sich die Übergangsfunktion dem Endwert  $h(\infty) = k_s$ , der durch die statische Verstärkung  $k_s$  des Systems bestimmt ist. Dieser Wert lässt sich aus Gl. (4.28) ermitteln, wobei man die bereits in Gl. (3.24) auf anderem Wege abgeleitete Beziehung

$$k_s = d - \mathbf{c}^T \mathbf{A}^{-1} \mathbf{b} \quad (4.29)$$

erhält.

Für Systeme, deren Parameter  $d$  in der Ausgangsgleichung nicht verschwindet, springt die Übergangsfunktion zur Zeit  $t = 0$  auf den Wert  $d$ , während für alle Systeme mit  $d = 0$  die Übergangsfunktion beim Wert  $h(0) = 0$  beginnt. Systeme mit  $d \neq 0$  werden deshalb als *sprungfähige Systeme* bezeichnet.



**Abb. 4.16:** Übergangsfunktionen unterschiedlicher Systeme

Abbildung 4.16 zeigt weitere Übergangsfunktionen. Im oberen Teil ist das Verhalten von schwingungsfähigen Systemen dargestellt, wobei die beiden Kurven zu Systemen mit unterschiedlicher Dämpfung gehören. Der untere Teil der Abbildung zeigt Übergangsfunktionen, die sich ohne Überspringen unterschiedlich schnell dem statischen Endwert nähern.

Erhält das System an Stelle der Eingangsgröße (4.25) eine sprungförmige Eingangsgröße mit der Amplitude  $u_0$

$$u(t) = u_0 \sigma(t),$$

so wird auf Grund der Linearität des Systems die Ausgangsgröße auch um den Faktor  $u_0$  verändert:

$$y(t) = u_0 h(t).$$

#### 4.2.6 Berechnung der Matrixexponentialfunktion

Die vorhergehenden Abschnitte haben gezeigt, dass die Matrixexponentialfunktion  $e^{\mathbf{A}t}$  in allen Bewegungsgleichungen vorkommt. Es gibt deshalb verschiedene Verfahren, mit denen man diese Funktion berechnen kann. Hier soll nur das Verfahren von SYLVESTER<sup>1</sup> angegeben werden.

<sup>1</sup> JAMES JOSEPH SYLVESTER (1814 – 1897), englischer Mathematiker

Im Abschn. 4.2.4 wurde gezeigt, dass sich die Eigenbewegung eines linearen Systems aus einer Summe von  $n$  e-Funktionen zusammensetzt, in deren Exponenten die Eigenwerte der Matrix  $\mathbf{A}$  vorkommen. Dementsprechend gibt es eine Darstellung der Matrixexponentialfunktion als Summe dieser  $n$  e-Funktionen:

$$e^{\mathbf{A}t} = \sum_{i=1}^n e^{\lambda_i t} \mathbf{F}_i \quad (4.30)$$

Die Formel von SYLVESTER zeigt, wie man die  $n$  Matrizen  $\mathbf{F}_i$  ausrechnen kann, wenn alle Eigenwerte von  $\mathbf{A}$  einfach sind:

$$\mathbf{F}_i = \prod_{j=1, j \neq i}^n \frac{\mathbf{A} - \lambda_j \mathbf{I}}{\lambda_i - \lambda_j}, \quad i = 1, 2, \dots, n.$$

Das wichtigste Merkmal dieser Formel besteht in der Tatsache, dass sie die Matrixexponentialfunktion als geschlossenen Ausdruck (4.30) angibt. Mit dieser Formel muss man die Funktion  $e^{\mathbf{A}t}$  nicht für jeden Zeitpunkt  $t$  einzeln berechnen wie mit den meisten anderen Berechnungsverfahren, sondern die Matrixexponentialfunktion entsteht in direkter Beziehung zu den Modi  $e^{\lambda_i t}$ , ( $i = 1, 2, \dots, n$ ) des Systems.

Dasselbe erreicht man über die Ähnlichkeitstransformation der Matrix  $\mathbf{A}$ . Wird diese auf alle Glieder der Reihe (4.4) angewendet, so erhält man die Beziehung

$$e^{\mathbf{A}t} = \mathbf{V} \operatorname{diag} e^{\lambda_i t} \mathbf{V}^{-1}, \quad (4.31)$$

die wiederum  $e^{\mathbf{A}t}$  in Abhängigkeit von den Modi des Systems darstellt. Diese Gleichung gilt für alle diagonalähnlichen Matrizen  $\mathbf{A}$ .

## 4.3 Verhalten nichtlinearer Systeme

### 4.3.1 Lösung der nichtlinearen Modellgleichungen

Während es für lineare Differentialgleichungen stets eine Lösung gibt, stellt sich bei nichtlinearen Differentialgleichungen (3.26)

$$\dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t)), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (4.32)$$

zunächst die Frage nach der Existenz einer Lösung. Diese Frage soll hier nicht ausführlich behandelt werden, sondern es wird vorausgesetzt, dass die Funktion  $\mathbf{g}(\mathbf{x}, u)$  eine Lipschitzbedingung erfüllt, dass also für hinreichend groß gewählte Konstanten  $k_x$  und  $k_u$  die Bedingungen

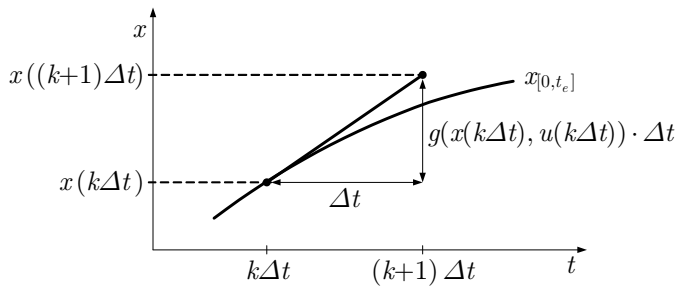
$$\begin{aligned} \|\mathbf{g}(\mathbf{x}_1, u) - \mathbf{g}(\mathbf{x}_2, u)\| &\leq k_x \|\mathbf{x}_1 - \mathbf{x}_2\| \\ \|\mathbf{g}(\mathbf{x}, u_1) - \mathbf{g}(\mathbf{x}, u_2)\| &\leq k_u |u_1 - u_2| \end{aligned}$$

für alle  $\mathbf{x}$ ,  $\mathbf{x}_1$ ,  $\mathbf{x}_2$  und  $u$ ,  $u_1$ ,  $u_2$  erfüllt sind, wobei  $\|\cdot\|$  eine Vektornorm und  $|\cdot|$  den Betrag kennzeichnet. Dadurch ist die Existenz einer Lösung der Differentialgleichung (4.32) gesichert.

Diese Voraussetzung wird auch bei allen späteren Betrachtungen zu nichtlinearen Systemen verwendet, ohne dass sie jedes Mal wiederholt wird.

Nichtlineare Differentialgleichungen können i. Allg. nicht geschlossen gelöst werden. Man ist deshalb bei der Vorhersage des Verhaltens nichtlinearer Systeme darauf angewiesen, die Differentialgleichung für einen Anfangszustand  $x_0$  und eine Eingangsgröße  $u_{[0,t_e]}$  numerisch zu lösen. Dafür sind in der numerischen Mathematik unterschiedliche Integrationsverfahren entwickelt worden, von denen man in Abhängigkeit von den Eigenschaften der Funktion  $g$  ein geeignetes auswählen muss. Hier soll keine Einführung in die numerische Lösung von Anfangswertproblemen gegeben, sondern nur durch eine kurze Wiedergabe des EULER-Verfahrens<sup>2</sup> auf die in der numerischen Mathematik entwickelten Verfahren hingewiesen werden.

**Eulerverfahren zur Lösung nichtlinearer Differentialgleichungen.** Das Eulerverfahren ist ein Einschrittverfahren, mit dem aus dem Zustandsvektor zur Zeit  $t$  der Wert zur Zeit  $t + \Delta t$  berechnet wird, wobei  $\Delta t$  die Integrationsschrittweite darstellt. Das Ergebnis ist eine Wertefolge für  $x(t)$  für die diskreten Zeitpunkte  $t = 0, \Delta t, 2\Delta t, \dots, k_e \Delta t = t_e$ .



**Abb. 4.17:** Eulerintegration einer nichtlinearen Differentialgleichung erster Ordnung

Grundlage dieses Verfahrens ist die Taylorentwicklung der Funktion  $x(t + \Delta t)$

$$x(t + \Delta t) = x(t) + \frac{dx(t)}{dt} \frac{\Delta t}{1!} + \frac{d^2x(t)}{dt^2} \frac{(\Delta t)^2}{2!} + \dots, \quad (4.33)$$

die nach dem zweiten Summanden abgebrochen wird

$$x(t + \Delta t) \approx x(t) + \frac{dx(t)}{dt} \Delta t$$

und durch Einsetzen der Differentialgleichung die Beziehung

$$x(t + \Delta t) \approx x(t) + g(x(t), u(t)) \cdot \Delta t$$

liefert (Abb. 4.17). Für hinreichend kleinen Zeitschritt  $\Delta t$  stimmt das Ergebnis mit guter Näherung, so dass das  $\approx$ -Zeichen durch das Gleichheitszeichen ersetzt wird. Setzt man nun für  $t$  einen beliebigen Zeitpunkt  $k\Delta t$  ein, so erhält man die rekursive Beziehung

<sup>2</sup> LEONHARD EULER (1707 - 1783), schweizer Mathematiker

Eulerintegration:

$$\left. \begin{aligned} \mathbf{x}(0\Delta t) &= \mathbf{x}_0 \\ \mathbf{x}((k+1)\Delta t) &= \mathbf{x}(k\Delta t) + \mathbf{g}(\mathbf{x}(k\Delta t), u(k\Delta t)) \cdot \Delta t \\ y(k\Delta t) &= h(\mathbf{x}(k\Delta t), u(k\Delta t)) \end{aligned} \right\} k = 0, 1, \dots, k_e. \quad (4.34)$$

In der ersten Zeile wird für  $\mathbf{x}(0 \cdot \Delta t)$  der Anfangswert  $\mathbf{x}_0$  eingesetzt. Anschließend wird der Zustand zum Zeitpunkt  $(k+1) \cdot \Delta t$  aus dem vorher berechneten Wert  $\mathbf{x}(k \cdot \Delta t)$  und dem Wert der Eingangsgröße zum Zeitpunkt  $k \cdot \Delta t$  bestimmt. Schließlich erhält man aus dem berechneten Zustand und der Eingangsgröße die gesuchte Ausgangsgröße  $y(k \cdot \Delta t)$ . Der Zähler  $k$  wird von null bis auf  $k_e = \frac{t_e}{\Delta t}$  schrittweise erhöht.

Bei dieser numerischen Lösung wird also nicht die vollständige Funktion  $u_{[0, t_e]}$  verwendet, um den Verlauf  $y_{[0, t_e]}$  zu berechnen, sondern es wird nur mit der diskreten Wertefolge

$$\mathbf{u} = (u(0), u(\Delta t), u(2\Delta t), \dots, u(t_e))$$

für die Eingangsgröße gearbeitet und die diskrete Wertefolge

$$\mathbf{y} = (y(0), y(\Delta t), y(2\Delta t), \dots, y(t_e))$$

der Ausgangsgröße berechnet. In der grafischen Darstellung kann man diese Werte durch Geraden verbinden und erhält eine umso bessere Approximation der Signale  $u_{[0, t_e]}$  bzw.  $y_{[0, t_e]}$ , je kleiner die Integrationsschrittweite  $\Delta t$  gewählt wird.

Dieses Verfahren zeigt die Idee der Integrationsverfahren für nichtlineare Differentialgleichungen. Es ist jedoch numerisch nicht sehr gut, weil bei diesem Verfahren der Verlauf der Funktion  $\mathbf{g}(\mathbf{x}, u)$  in jedem betrachteten Zeitintervall der Breite  $\Delta t$  durch die Tangente an die Kurve  $\mathbf{g}$  ersetzt wird (Abb. 4.17). Deshalb wurden genauere Verfahren entwickelt, wie z. B. das RUNGE-KUTTA-Verfahren<sup>3</sup>, bei dem der Verlauf von  $\mathbf{g}$  im betrachteten Zeitintervall besser approximiert wird.

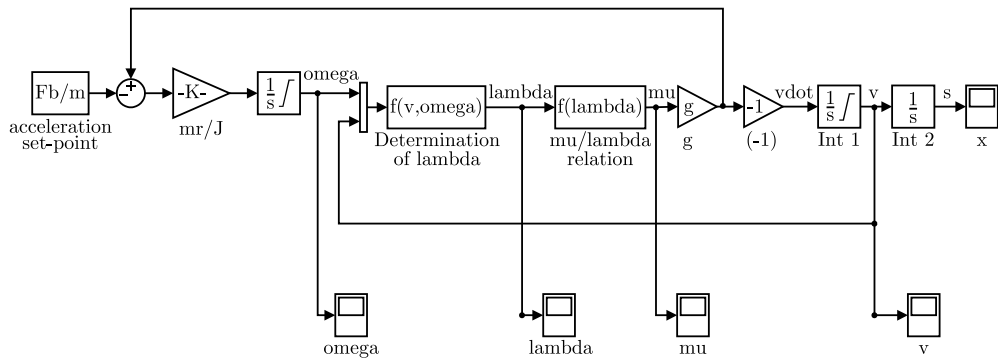
### 4.3.2 Rechnergestützte Analyse nichtlinearer Systeme

In der industriellen Praxis verwendet man kommerziell verfügbare Programmpakete, um die Systemgleichungen zu lösen. Bei einigen von ihnen muss man das Modell nicht in die Standardform (4.32) überführen, sondern man kann ausgehend vom Blockschaltbild die Gleichungen der einzelnen Komponenten getrennt voneinander eingeben. Das hat den Vorteil, dass man die Systemstruktur sehr einfach ändern kann, ohne jedesmal die aufwändigen Umformungen in die Modellform (4.32) wiederholen zu müssen.

Bei dem zu MATLAB gehörigen Programmpaket Simulink kann das Blockschaltbild direkt zur Modelleingabe verwendet werden. Abbildung 4.18 zeigt, dass man das Blockschaltbild aus Abb. 3.16 auf S. 109 dabei nur geringfügig umformen muss. Hinzugekommen sind lediglich die unteren Blöcke, die eine grafische Ausgabe der betreffenden Signale auf dem Bildschirm

<sup>3</sup> CARL RUNGE (1856 – 1927) und WILHELM KUTTA (1867 – 1944), deutsche Mathematiker





**Abb. 4.18:** Simulink-Blockschaltbild des Modells für den Abbremsvorgang eines Fahrzeugs

bewirken. Die Bedeutung der anderen Simulink-Blöcke kann man sich aus dem Vergleich mit dem Blockschaltbild erschließen.

Diese Art der Modelleingabe hat einen weiteren Vorteil bezüglich der Parametrierung. In einer Zustandsgleichung der Form (4.32) erscheinen die physikalischen Parameter untereinander verknüpft als Koeffizienten der nichtlinearen Vektorfunktion  $g$ . Im Blockschaltbild dagegen sind die physikalischen Parameter weiterhin dem entsprechenden Block zugeordnet, so dass sie dort gespeichert werden und gegebenenfalls unabhängig voneinander verändert werden können.

#### Beispiel 4.2 Simulation des Abbremsvorgangs eines Fahrzeugs

Um zu zeigen, wie man mit rechnergestützten Verfahren nichtlineare Systeme analysiert, wurde das im Beispiel 3.9 angegebene nichtlineare Modell, das den Abbremsvorgang eines Kraftfahrzeugs beschreibt, in Simulink implementiert. Das heißt, es wurde das in Abb. 4.18 gezeigte Blockschaltbild mit den im Grafikeditor von Simulink verfügbaren Blöcken zusammengestellt und es wurden den Blöcken die entsprechenden mathematischen Beschreibungen zugeordnet. Dreiecke beschreiben statische Übertragungsglieder, deren Ausgang sich aus dem Produkt des Eingangs mit dem Verstärkungsfaktor des Blocks ergibt. Blöcke mit Integrationszeichen führen eine Integration der Eingangsgröße durch. Die beiden mit  $f(v, \omega)$  bzw.  $f(\lambda)$  gekennzeichneten Blöcke sind nichtlineare statische Übertragungsglieder, deren Ausgang sich aus einer nichtlinearen Funktion des Eingangs ergibt. Da diese Simulink-Blöcke nur einen Eingang haben, müssen die beiden Eingänge  $\omega(t)$  und  $v(t)$  des linken Blocks vorher zu einem Vektor zusammengefasst werden, was durch den schmalen Balken verdeutlicht wird. Der ganz linke Block ist ein Signalgenerator, der die konstante Größe  $F_B/m$  vorgibt. Die nach unten herausgezogenen Blöcke dienen zur grafischen Darstellung der berechneten Signale.

Bevor das Systemverhalten simuliert werden kann, müssen die Systemparameter vorgegeben werden. Dies geschieht durch Eintragen der Parameterwerte in die Blöcke oder durch Wertezuordnung von der Arbeitsoberfläche (*command window*) von MATLAB aus. Es wurden folgende Parameter verwendet:

Parameter	Symbol und Wert	Maßeinheit
Erdbeschleunigung	$g = 9,81$	$\frac{\text{m}}{\text{s}^2}$
$\frac{1}{4}$ der Fahrzeugmasse	$m = 250$	kg
Radradius	$r = 0,406$	m
Trägheitsmoment des Rades	$J = 12,73$	$\text{kgm}^2$

Für die nichtlineare  $\mu(\lambda)$ -Kennlinie (3.38) gibt es die analytische Beschreibung

$$\mu = c_1 \left( 1 - e^{-c_2 \lambda} \right) - c_3 \lambda, \quad (4.35)$$

zu der beispielsweise folgende Parameter gehören:

Straßenbelag	Parameter in Gl. (4.35)		
	$c_1$	$c_2$	$c_3$
Trockener Asphalt	1,2801	23,99	0,52
Nasser Asphalt	0,857	33,822	0,347
Eis	0,1946	94,129	0,0646

Außerdem ist der Anfangszustand des Systems (also die Anfangswerte der Integratoren) festzulegen. Hier wird mit  $v_0 = 50 \frac{\text{km}}{\text{h}}$  und der dazu gehörenden Winkelgeschwindigkeit  $\omega_0 = \frac{v_0}{r}$  gearbeitet. Der zurückgelegte Weg wurde von  $s_0 = 0 \text{ m}$  an gemessen.

Für die Simulation des Systemverhaltens muss schließlich das Zeitintervall  $[0, t_e]$  festgelegt und die Eingangsgröße ausgewählt werden. Es wird mit einer konstanten Bremskraft

$$F_B(t) = m \frac{v_0}{2} = 1736 \text{ N}$$

gearbeitet, bei der man erwarten kann, dass das Fahrzeug mit einer Bremsbeschleunigung von

$$a_{\text{Soll}}(t) = \frac{F_B}{m} = \frac{1736 \text{ N}}{250 \text{ kg}} = 6,95 \frac{\text{m}}{\text{s}^2}$$

in etwa 2 Sekunden zum Stillstand kommt.

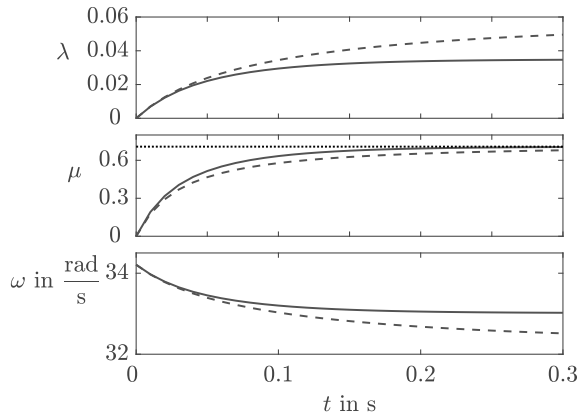
Zunächst wird untersucht, wie sich der Schlupf nach Eingreifen der Bremskraft aufbaut. Da dieser Vorgang wesentlich schneller ist als der Abbremsvorgang des Fahrzeugs, kann dabei mit einer konstanten Geschwindigkeit gearbeitet werden. Im Blockschaltbild wird deshalb die Rückführung von der Geschwindigkeit  $v$  in den linken nichtlinearen Block aufgetrennt und statt dessen die konstante Geschwindigkeit  $v = v_0$  an den Block angelegt. Damit ist die Simulationaufgabe vollständig beschrieben, so dass der Lösungsalgorithmus für die durch das Blockschaltbild dargestellte nichtlineare Differentialgleichung alle Signalverläufe berechnen kann.

Das in Abb. 4.19 für das Abbremsen auf Asphalt dargestellte Simulationsergebnis zeigt, dass sich bei trockenem Asphalt ein Schlupf von etwa 0,034 aufbaut, der einer Kraftschlussbeanspruchung von  $\mu = 0,707$  entspricht. Das Fahrzeug erhält also eine Bremsbeschleunigung von

$$a_{\text{Ist}} = \mu g = 0,707 \cdot 9,81 \frac{\text{m}}{\text{s}^2} = 6,95 \frac{\text{m}}{\text{s}^2},$$

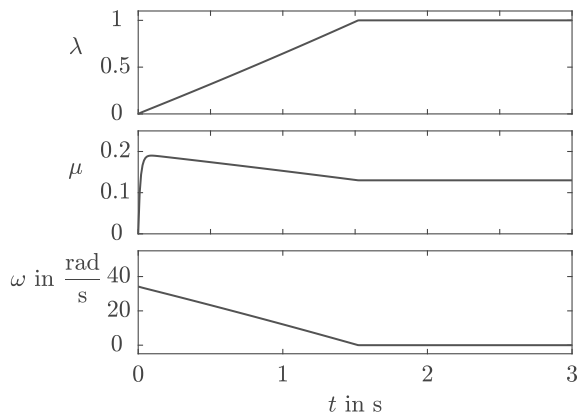
die erwartungsgemäß gleich der Sollbeschleunigung  $a_{\text{Soll}}$  ist. Wenn das System in eine Ruhelage einschwingt, muss die Summationsstelle im linken Teil des Blockschaltbilds ein verschwindendes Signal liefern, woraus man diesen Endwert der Bremsbeschleunigung ablesen kann.

Um das Abbremsen auf nassem Asphalt simulieren zu können, müssen im Modell lediglich die Koeffizienten  $c_1$ ,  $c_2$  und  $c_3$  der  $\mu(\lambda)$ -Kennlinie ausgetauscht werden. Wie die gestrichelten Kurven



**Abb. 4.19:** Schlupf, Kraftschlussbeanspruchung und Winkelgeschwindigkeit beim Abbremsen auf trockenem (—) und nassem (---) Asphalt

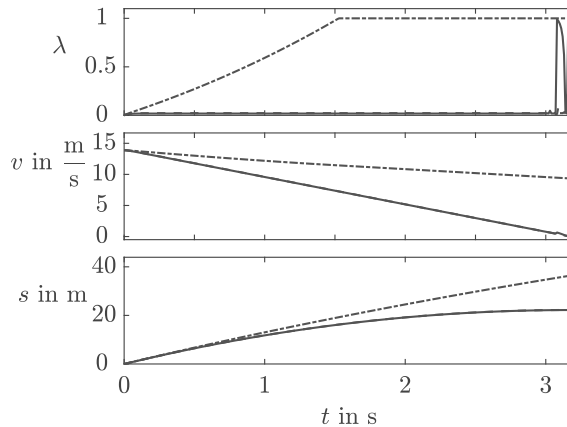
zeigen, stellt sich ein höherer Schlupf ein. Das erklärt sich daraus, dass nicht der Schlupf, sondern die Kraftschlussbeanspruchung denselben statischen Endwert haben muss, damit sich die Signale an der bereits genannten Summationsstelle aufheben und das System eine Ruhelage einnimmt. Auf nassem Asphalt ist jedoch ein größerer Schlupf notwendig, um dieselbe Kraftschlussbeanspruchung wie vorher zu erhalten (vgl. Abb. 3.15). Die Drehgeschwindigkeit des Rades ist deshalb etwas niedriger als bei trockenem Asphalt.



**Abb. 4.20:** Schlupf, Kraftschlussbeanspruchung und Winkelgeschwindigkeit beim Abbremsen auf vereister Straße mit blockierenden Rädern

Für das Abbremsen auf vereister Straße ist das Simulationsergebnis in Abb. 4.20 zu sehen. Erwartungsgemäß blockiert das Rad, weil die verwendete Bremskraft eine Kraftschlussbeanspruchung von  $\mu = 0,707$  erfordert, die über dem Haftbeiwert  $\mu_H = 0,2$  von Eis liegt (vgl. Abb. 3.15). Deshalb steigt

der Schlupf bis auf seinen Maximalwert 1 an, bei dem das Rad blockiert. Das Rad bremst nur noch mit dem Gleitbeiwert  $\mu = \mu_G = 0,15$  von Eis.



**Abb. 4.21:** Abbremsmanöver auf Asphaltstraße (—) und auf vereister Straße (---)

Als zweites wird das Abbremsmanöver als Ganzes simuliert, wofür mit dem vollständigen Blockschaltbild aus Abb. 4.18 gearbeitet wird. Das Ergebnis ist in Abb. 4.21 dargestellt. Auf der Asphaltstraße wird das Fahrzeug in etwa drei Sekunden zum Stillstand gebracht, wobei es etwa 20 Meter zurücklegt. Auf vereister Straße ist der Bremsweg wesentlich länger. Nach etwa drei Sekunden ist die Geschwindigkeit noch nicht einmal auf die Hälfte der Anfangsgeschwindigkeit gefallen.

**Diskussion.** Die offensichtlich falschen Simulationsergebnisse im oberen Teil von Abb. 4.21 weisen auf eine Gefahr bei der Verwendung von rechnergestützten Werkzeugen hin. So bequem das Programmieren der einzelnen Blöcke und die Verwendung vorgefertigter Lösungsalgorithmen auch ist, es kann schnell passieren, dass man bei großen Systemen die Übersicht über die zu erwartenden Ergebnisse verliert und möglicherweise numerische Fehler als Systemeigenschaften interpretiert. Die hier gezeigten offensichtlich falschen Schlupfwerte beim Abbremsen auf der Asphaltstraße sind dadurch begründet, dass sich das Fahrzeug nach etwa drei Sekunden dem Stillstand nähert, wodurch der Nenner in der Definitionsgleichung (3.37) des Schlupfes verschwindet. Da das Simulationssystem diese numerische Schwierigkeit nicht von allein umgehen kann, muss der nichtlineare Block so programmiert werden, dass er den Schlupf auch bei sehr kleinen Geschwindigkeiten numerisch stabil berechnet, was hier zur Demonstration dieser Fehlerquelle nicht getan wurde. □

#### Aufgabe 4.3 Simulation des Abbremsvorgangs eines Fahrzeugs



Von der Homepage zu diesem Buch [www.atp.rub.de/Buch/AT](http://www.atp.rub.de/Buch/AT) können Sie das Simulink-Blockschaltbild `carbrakevel.mdl` aus Abb. 4.18 für die folgenden Untersuchungen herunterladen:

1. Wiederholen Sie die im Beispiel 4.2 angegebenen Berechnungen. Fügen Sie zusätzliche Ausgabeblocke („scopes“) ein und untersuchen Sie die Ausgangsgrößen der einzelnen Blöcke.
2. Simulieren Sie das Verhalten des Fahrzeugs für unterschiedliche Anfangszustände, wobei Sie Zustände auf beiden Seiten der Schlupfkennlinie in Abb. 3.15 auswählen. Worin unterscheiden sich die erhaltenen Kurven prinzipiell?

3. Berechnen Sie die freie Bewegung, die erzwungene Bewegung und die Überlagerung beider Bewegungen und untersuchen Sie, ob trotz des nichtlinearen Systemverhaltens Gl. (4.10) wenigstens näherungsweise gilt.  $\square$

## 4.4 Kennwertermittlung

Im Kap. 3 wurde das Modell durch Aufstellung der physikalischen Grundbeziehungen erhalten, die die im System wirkenden physikalischen Vorgänge beschreiben. Man spricht bei dieser Art der Modellbildung auch von einer *theoretischen Modellbildung* oder theoretischen Prozessanalyse. Sie ist immer dann erfolgreich, wenn die sich in dem System abspielenden Vorgänge im Einzelnen bekannt sind und wenn die Werte aller Parameter, die in den Beschreibungen dieser Vorgänge auftreten, bestimmt werden können.

Bei vielen praktischen Aufgabenstellungen will man für ein gegebenes System ein einfaches, möglicherweise nur näherungsweise richtiges Modell angeben, ohne den relativ aufwändigen Weg der theoretischen Modellbildung zu gehen. Da das Verhalten eines Systems in gemessenen Verläufen der Eingangs- und Ausgangsgrößen erkennbar ist, bietet sich ein alternativer Modellbildungsweg an, den man mit *Prozessidentifikation* bezeichnet. Aus den zahlreichen dafür entwickelten Methoden soll hier nur eine sehr einfache Methode herausgegriffen werden, mit der man das Zustandsraummodell eines Systems erster Ordnung anhand einer gemessenen Übergangsfunktion bestimmt.

Betrachtet wird ein System erster Ordnung, also ein System mit einer skalaren Zustandsgleichung

$$\dot{x}(t) = ax(t) + bu(t), \quad x(0) = x_0.$$

Bei diesem System ist es üblich, die beiden Parameter  $a$  und  $b$  in Abhängigkeit von einer Zeitkonstanten

$$T = -\frac{1}{a} \quad (4.36)$$

auszudrücken und die statische Verstärkung  $k_s$  in die Ausgangsgleichung zu schreiben:

$$\text{System erster Ordnung : } \begin{cases} \dot{x}(t) = -\frac{1}{T}x(t) + \frac{1}{T}u(t), & x(0) = x_0 \\ y(t) = k_s x(t). \end{cases} \quad (4.37)$$

Die Übergangsfunktion

$$h(t) = k_s \left(1 - e^{-\frac{t}{T}}\right) \quad (4.38)$$

dieses Systems hat den in Abb. 4.22 angegebenen Verlauf. Gesucht sind die Parameter  $T$  und  $k_s$ .

Wie die folgenden Überlegungen zeigen, kann man die beiden gesuchten Parameter aus der grafischen Darstellung von  $h(t)$  ablesen. Die statische Verstärkung  $k_s$  erhält man als Endwert der Übergangsfunktion:

$$k_s = h(\infty).$$

Für die Festlegung der Zeitkonstanten  $T$  wird die Ableitung der Übergangsfunktion gebildet

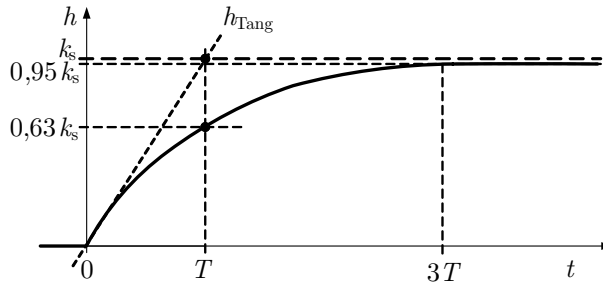


Abb. 4.22: Übergangsfunktion eines Systems erster Ordnung

$$\dot{h}(t) = k_s \frac{1}{T} e^{-\frac{t}{T}},$$

die für  $t = 0$  den Wert

$$\dot{h}(0) = \frac{k_s}{T}$$

hat. Dieser Wert gibt die Steigung der Tangente an die Übergangsfunktion zur Zeit  $t = 0$  an, die durch

$$h_{\text{Tang}}(t) = \dot{h}(0) t = k_s \frac{t}{T}$$

beschrieben ist. Sie erreicht den Wert  $k_s$  zur Zeit  $t = T$ . Folglich erhält man  $T$  aus der grafischen Darstellung von  $h(t)$ , indem man für  $t = 0$  die Tangente an die Kurve legt, den Schnittpunkt dieser Tangente mit der Parallelen zur Zeitachse mit dem Abstand  $k_s$  konstruiert und am Schnittpunkt die Zeitkonstante  $T$  abliest (Abb. 4.22).

Andere Möglichkeiten zur Bestimmung von  $T$  erhält man, wenn man die Übergangsfunktion für die Zeitpunkte  $t = T$  bzw.  $t = 3T$  berechnet:

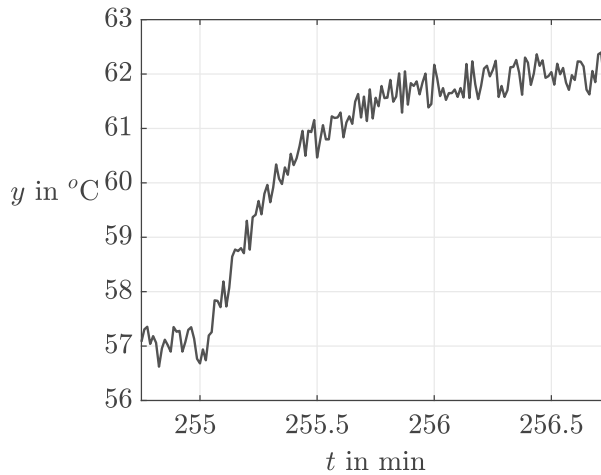
$$h(T) = k_s(1 - e^{-1}) \approx 0,63 k_s \quad (4.39)$$

$$h(3T) = k_s(1 - e^{-3}) \approx 0,95 k_s. \quad (4.40)$$

Die erste Gleichung zeigt, dass man die Zeitkonstante  $T$  als den Zeitpunkt bestimmen kann, bei dem der Wert der gemessenen Übergangsfunktion 63% des Endwertes beträgt („63%-Regel“). Die zweite Gleichung besagt, dass die Zeitkonstante ein Drittel der Zeit ist, die bis zum Erreichen von 95% des Endwertes vergeht („95%-Regel“).

#### Aufgabe 4.4\* Kennwertermittlung eines Wärmeübertragers

Für den in Abb. 3.20 auf S. 115 dargestellten Wärmeübertrager soll ein Modell der Form (4.37) aufgestellt werden. Die beiden Flüssigkeiten fließen mit dem konstanten Volumenstrom  $w_1$  bzw.  $w_2$  in den Wärmeübertrager und haben dabei die Temperaturen  $T_{10}$  bzw.  $T_{20}$ . Die wärmere Flüssigkeit gibt Wärme an die kältere ab, so dass sich die Temperaturen auf die Werte  $T_1(t)$  bzw.  $T_2(t)$  verändern. Im Folgenden wird angenommen, dass die Temperatur  $T_{20}(t)$  des Kühlmittels die durch eine Steuerung beeinflussbare Eingangsgröße  $u(t)$  und die Austrittstemperatur  $T_1(t)$  der anderen Flüssigkeit als Ausgangsgröße  $y(t)$  messbar sind.



**Abb. 4.23:** Experimentell bestimmte Übergangsfunktion eines Wärmeübertragers

Abbildung 4.23 zeigt einen Ausschnitt aus dem Temperaturverlauf  $y(t) = T_1(t)$ , wobei die Zeit in Minuten und die Ausgangsgröße in Grad Celsius gemessen wurden. Zur Zeit  $t = 255$  min wurde die Temperatur  $T_{20}(t)$  um 10 Kelvin erhöht. Das gesuchte lineare Modell soll das Verhalten des Wärmeübertragers um den Arbeitspunkt beschreiben.

1. Wie müssen die Skalen der Abb. 4.23 verändert werden, damit sie der in Abb. 4.22 gezeigten Übergangsfunktion entsprechen?
2. Bestimmen Sie die Modellparameter mit den drei angegebenen Methoden und vergleichen Sie die Ergebnisse. Welche physikalische Bedeutung haben die von Ihnen ermittelten Parameter  $k_s$  und  $T$  für den Wärmeübertrager? Welche Maßeinheiten haben sie?
3. Vergleichen Sie Ihr Modell mit dem in Aufgabe 3.7 erhaltenen Modell (17.2) auf S. 591.
4. Warum gibt das durch Kennwertermittlung aufgestellte Modell das Verhalten des Wärmeübertragers sehr gut wiedergeben? □

## Literaturhinweise

Für die Kennwertermittlung wurden vielfältige grafische Methoden erarbeitet, mit denen auch Systeme höherer Ordnung anhand von Messkurven identifiziert werden können. Diese Verfahren haben heute insofern noch Bedeutung, als dass man mit ihnen ohne großen rechentechnischen Aufwand eine grobe Beschreibung dynamischer Systeme erhalten kann und damit einen Einblick in die wichtigsten Systemeigenschaften bekommt. Sie sind z. B. in [84] beschrieben.





# 5

## Steuerbarkeit und Beobachtbarkeit

*Steuerbarkeit und Beobachtbarkeit sind zwei wichtige Eigenschaften dynamischer Systeme, die die Lösbarkeit von Überwachungs- und Steuerungsaufgaben maßgebend beeinflussen. Dieses Kapitel behandelt Kriterien für die diesbezügliche Systemanalyse und zeigt, dass diese Eigenschaften im Wesentlichen von der Systemstruktur abhängen.*

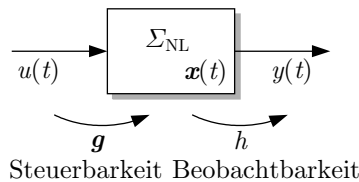
### 5.1 Definition der Steuerbarkeit und der Beobachtbarkeit

Das Zustandsraummodell

$$\Sigma_{\text{NL}} : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = h(\mathbf{x}(t)) \end{cases} \quad (5.1)$$

stellt den Zusammenhang zwischen der Eingangsgröße  $u(t)$  und der Ausgangsgröße  $y(t)$  dar, wobei in der ersten Gleichung der Einfluss des Eingangs auf den Zustand  $\mathbf{x}(t)$  beschrieben und mit der zweiten Gleichung der Ausgang aus dem Zustand berechnet wird (Abb. 5.1). Für viele Automatisierungsaufgaben ist es wichtig zu wissen, ob es einerseits durch eine geeignete Wahl der Eingangsgröße  $u(t)$  gelingen kann, das System  $\Sigma_{\text{NL}}$  in einen beliebig vorgegebenen Zielzustand  $\mathbf{x}_e$  zu steuern (Steuerbarkeit) und ob es andererseits möglich ist, aus der gemessenen Ausgangsgröße  $y(t)$  den Zustand des Systems zu rekonstruieren (Beobachtbarkeit). Diese Fragen stehen im Mittelpunkt dieses Kapitels.

Die folgenden Definitionen zeigen, dass die beiden Fragen nur für vollständig steuerbare und vollständig beobachtbare Systeme mit ja zu beantworten sind. In den Abschnitten 5.2 und



**Abb. 5.1:** Bezug der Steuerbarkeit und Beobachtbarkeit zu den Systemgleichungen

5.3 werden Kriterien angegeben, unter denen lineare Systeme diese Eigenschaften besitzen. Dann wird im Abschn. 5.4 gezeigt, dass diese Eigenschaften im Wesentlichen durch die Struktur des Systems bestimmt sind und deshalb aus dem Strukturgraphen abgelesen werden können. Wenn ein System diese Eigenschaften nicht besitzt, so kann es, wie im Abschn. 5.5 erläutert wird, in mehrere Teilsysteme zerlegt werden, von denen nur eines vollständig steuerbar und vollständig beobachtbar ist.

**Steuerbarkeit.** Für die Steuerbarkeitsuntersuchung ist nur die Zustandsgleichung (5.1) maßgebend. Das System ist zur Zeit  $t = 0$  im Zustand  $x_0$  und soll in einem endlichen Zeitintervall  $[0, t_e]$  in einen vorgegebenen Zielzustand  $x_e$  überführt werden. Dafür muss die Eingangsgröße  $u_{[0, t_e]}$  geeignet gewählt werden. Ein System heißt vollständig steuerbar, wenn dies für beliebige Anfangs- und Endzustände  $x_0$  und  $x_e$  möglich ist:

**Definition 5.1 (Steuerbarkeit)**

*Das System (5.1) heißt vollständig steuerbar, wenn es von jedem Anfangszustand  $x_0$  in endlicher Zeit  $t_e$  durch eine geeignet gewählte Eingangsgröße  $u_{[0, t_e]}$  in einen beliebig vorgegebenen Endzustand  $x(t_e) = x_e$  überführt werden kann.*

Natürlich hängt der Verlauf  $u_{[0, t_e]}$  der Eingangsgröße von  $x_0$ ,  $x_e$  und  $t_e$  ab. Die Definition lässt zu, dass die Endzeit  $t_e$  in Abhängigkeit vom vorgegebenen Zielzustand  $x_e$  gewählt wird, dass also bei „weit entfernten“ Zuständen mit einer größeren und bei „näher gelegenen“ Zuständen  $x_e$  mit einer kürzeren Zeit  $t_e$  gearbeitet wird. Wichtig ist nur, dass diese Zeit endlich sein muss. Das Attribut „vollständig“ bezieht sich darauf, dass die Zustandsüberführung zwischen beliebigen Zuständen  $x_0$  und  $x_e$  möglich sein soll.

**Beobachtbarkeit.** Bei der Beobachtbarkeit wird untersucht, ob man aus der Kenntnis des Eingangs  $u$  und des Ausgangs  $y$  den Anfangszustand  $x_0$  oder den aktuellen Zustand  $x(t_e)$  rekonstruieren kann.

Wie man sich leicht überlegen kann, reicht ein Messwertpaar  $(u(t), y(t))$ , das die Eingangsgröße und die Ausgangsgröße zu einem Zeitpunkt  $t$  beschreibt, nicht aus, um den Zustand  $x(t)$  zu bestimmen. Das geht schon aus Dimensionsgründen nicht: Das System hat eine einzige Messgröße  $y(t)$ , aber  $n$  Zustandsvariablen  $x_i(t)$ . Es sind also mindestens  $n$  Messungen notwendig.

Man stellt das Beobachtungsproblem deshalb allgemeiner und will den Zustand  $x(t)$  aus dem bekannten Verlauf  $u_{[0,t_e]}$  und  $y_{[0,t_e]}$  des Eingangs und des Ausgangs unter Verwendung des Modells (5.1) rekonstruieren. Wenn dies möglich ist, wird das System beobachtbar genannt:

**Definition 5.2 (Beobachtbarkeit)**

*Das System (5.1) heißt vollständig beobachtbar, wenn der Anfangszustand  $x_0$  aus dem für ein endliches Intervall  $[0, t_e]$  bekannten Verlauf der Eingangsgröße  $u_{[0,t_e]}$  und der Ausgangsgröße  $y_{[0,t_e]}$  bestimmt werden kann.*

Wenn man  $x_0$  berechnet hat, so kann man mit Hilfe des Modells und des Eingangsgrößenverlaufs  $u_{[0,t_e]}$  auch den aktuellen Zustand  $x(t_e)$  ermitteln.

## 5.2 Steuerbarkeit linearer Systeme

### 5.2.1 Steuerbarkeitskriterium

Die Steuerbarkeit wird im Folgenden für lineare Systeme

$$\Sigma: \dot{x}(t) = Ax(t) + bu(t), \quad x(0) = x_0 \quad (5.2)$$

untersucht. Das von KALMAN<sup>1</sup> bewiesene Kriterium bezieht sich auf die Steuerbarkeitsmatrix

$$S_S = (b \quad Ab \quad A^2b \quad \dots \quad A^{n-1}b), \quad (5.3)$$

die aus den nebeneinander geschriebenen Spaltenvektoren  $b, Ab, \dots, A^{n-1}b$  besteht.

**Satz 5.1 (Steuerbarkeitskriterium)**

*Das System (5.2) ist genau dann vollständig steuerbar, wenn die Steuerbarkeitsmatrix  $S_S$  den Rang  $n$  hat:*

$$\text{Rang } S_S = n. \quad (5.4)$$

Für die hier betrachteten Systeme mit einer Eingangsgröße und einer Ausgangsgröße ist die Bedingung (5.4) äquivalent zur Forderung

$$\det S_S \neq 0. \quad (5.5)$$

An Stelle eines Beweises wird im Folgenden gezeigt, dass man für Systeme, die dieses Kriterium erfüllen, eine Steuerung  $u_{[0,t_e]}$  finden kann, die die gewünschte Zustandsüberführung vornimmt. Dazu wird zunächst die Matrix

<sup>1</sup> RUDOLF EMIL KÁLMÁN (1930 – 2016), amerikanischer Mathematiker ungarischer Herkunft

$$\mathbf{W}_S = \int_0^{t_e} e^{\mathbf{A}t} \mathbf{b} \mathbf{b}^T e^{\mathbf{A}^T t} dt \quad (5.6)$$

gebildet, die als *gramsche Steuerbarkeitsmatrix* bezeichnet wird. Diese  $(n \times n)$ -Matrix ist regulär, wenn die Bedingung (5.4) erfüllt, das System  $\Sigma$  also vollständig steuerbar ist. Die gewünschte Zustandsüberführung von  $\mathbf{x}_0$  nach  $\mathbf{x}_e$  kann dann mit der Eingangsgröße

$$u_{[0,t_e]} = -\mathbf{b}^T e^{\mathbf{A}^T(t_e - t)} \mathbf{W}_S^{-1} \left( e^{\mathbf{A}t_e} \mathbf{x}_0 - \mathbf{x}_e \right), \quad 0 \leq t \leq t_e \quad (5.7)$$

erfolgen. Dass dies so ist, erkennt man, wenn man diese Steuerung in die Bewegungsgleichung (4.5) einsetzt und  $\mathbf{x}(t_e)$  ausrechnet.

Die angegebene Steuerung  $u_{[0,t_e]}$  beschreibt keinesfalls den einzigen Verlauf der Eingangsgröße, durch den die gewünschte Umsteuerung erfolgt. Theoretisch gibt es unendlich viele andere Eingangssignale, die dasselbe leisten. Die hier angegebene Steuerung hat die interessante Eigenschaft, dass sie im Mittel die kleinste Amplitude hat, also die „Energie“, mit der die Umsteuerung erfolgt, minimal ist.

### 5.2.2 Eigenschaften vollständig steuerbarer Systeme

Aus der Steuerbarkeitsdefinition sowie der angegebenen Steuerung (5.7) ergeben sich folgende Merkmale vollständig steuerbarer Systeme:

- Bei der angegebenen Steuerung kann die Endzeit  $t_e$  beliebig gewählt werden. Das heißt, dass lineare Systeme in beliebig kurzer Zeit zwischen beliebigen Zuständen umgesteuert werden können, wenn sie vollständig steuerbar sind. Der Preis für kurze Umsteuerzeiten liegt i. Allg. in großen Stellamplituden. Je kleiner  $t_e$  ist, umso kleiner sind die Elemente von  $\mathbf{W}_S$  und umso größer die Elemente von  $\mathbf{W}_S^{-1}$  und  $u(t)$  nach Gl. (5.7).
- Dass ein System in einen beliebigen Zustand  $\mathbf{x}(t_e) = \mathbf{x}_e$  gesteuert werden kann, heißt nicht, dass es möglich ist, das System durch eine geeignete Wahl von  $u(t)$  für  $t > t_e$  im Zustand  $\mathbf{x}_e$  zu halten. Zustände, in denen das System auch für  $t > t_e$  verharren kann, erhält man aus Gl. (5.2) und der Beziehung  $\dot{\mathbf{x}} = \mathbf{0}$ , die auf die Bedingung

$$\mathbf{x}_e = \mathbf{k}_{sx} \bar{u} \quad (5.8)$$

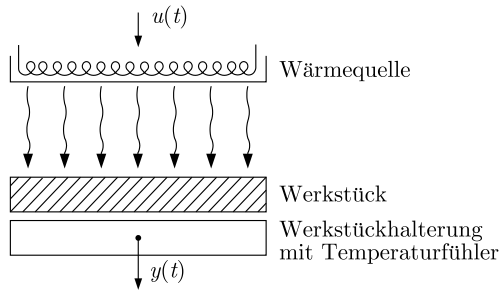
mit dem Vektor

$$\mathbf{k}_{sx} = -\mathbf{A}^{-1} \mathbf{b} \quad (5.9)$$

führt. Diese Zustände liegen auf der Linie im Zustandsraum, die durch Gl. (5.8) für  $\bar{u} = -\infty \dots + \infty$  beschrieben wird.

#### Beispiel 5.1 Steuerbarkeit eines Industrieofens

Es wird ein Industrieofen betrachtet, in dem ein Werkstück auf eine vorgegebene Temperatur erhitzt wird. Das Werkstück ist auf einem Werkstückhalter montiert, dessen Temperatur gemessen wird. Hier



**Abb. 5.2:** Erwärmung eines Werkstücks in einem Industrieofen

wird untersucht, ob es möglich ist, das Werkstück durch entsprechende Vorgabe der die Wärmequelle steuernden Stellgröße  $u(t)$  auf eine beliebige Solltemperatur  $T_e$  zu bringen (Abb. 5.2).

**Modellbildung.** Bei der Modellierung des Erwärmungsprozesses wird von einer homogenen Energieverteilung innerhalb des Werkstücks und des Werkstückhalters ausgegangen, so dass dort an allen Stellen die Temperatur  $T_1(t)$  bzw.  $T_2(t)$  herrscht.  $T_u$  ist die Umgebungstemperatur, die als konstant angenommen wird. Der Wärmestrom  $\dot{Q}(t)$  bezeichnet die pro Zeiteinheit vom Werkstück aufgenommene Wärmemenge, wobei angenommen wird, dass  $\dot{Q}(t)$  direkt durch die Stellgröße  $u(t)$  vorgegeben wird ( $u(t) = \dot{Q}(t)$ ).

Unter den genannten Modellannahmen wird der Erwärmungsprozess mit Hilfe von drei Wärmeströmen beschrieben, die in Abb. 5.3 mit  $\dot{Q}(t)$ ,  $\dot{Q}_{21}(t)$  und  $\dot{Q}_{u2}(t)$  bezeichnet sind und die von den Temperaturdifferenzen abhängen. Für  $\dot{Q}_{21}(t)$  gilt

$$\dot{Q}_{21}(t) = A k_{21} (T_1(t) - T_2(t)),$$

wobei  $A$  die Fläche des Werkstücks und des Werkstückhalters und  $k_{21}$  eine Konstante ist, die von den Schichtdicken  $d_1$  und  $d_2$  sowie von den Wärmeleitfähigkeiten  $\lambda_1$  und  $\lambda_2$  beider Schichten abhängt. Wenn man annimmt, dass die Temperaturen  $T_1(t)$  und  $T_2(t)$  in der Mitte jeder Schicht wirken, muss der Wärmestrom  $\dot{Q}_{21}(t)$  über die Länge  $\frac{d_1}{2}$  das Werkstück mit der Wärmeleitfähigkeit  $\lambda_1$  und dann über die Länge  $\frac{d_2}{2}$  den Werkstückhalter mit der Wärmeleitfähigkeit  $\lambda_2$  durchströmen. Der Gesamtwiderstand

$$\frac{d_1}{2\lambda_1} + \frac{d_2}{2\lambda_2}$$

setzt sich aus der Summe der beiden Widerstände zusammen, so dass man für die Konstante  $k_{21}$  die Beziehung

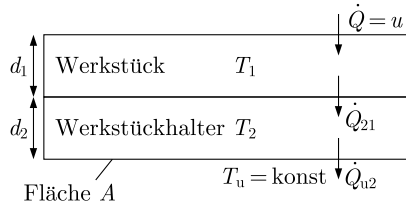
$$k_{21} = \frac{1}{\frac{d_1}{2\lambda_1} + \frac{d_2}{2\lambda_2}}$$

erhält.

Der Wärmestrom in einen festen Körper und die Temperaturänderung des Körpers stehen über die Beziehung

$$\dot{Q}(t) = c m \dot{T}(t)$$

miteinander in Verbindung, in der  $c$  die spezifische Wärmekapazität und  $m$  die Masse darstellt. Für die Masse gilt  $m = \rho d A$  mit  $\rho$  als Dichte des Stoffes. Für den Wärmestrom  $\dot{Q}(t)$  muss die pro Zeiteinheit in der betreffenden Materialsicht verbleibende Wärme eingesetzt werden. Deshalb gilt für das Werkstück



**Abb. 5.3:** Modellansatz zur Beschreibung des Industrieofens

$$c_1 \rho_1 d_1 A \dot{T}_1(t) = \dot{Q}(t) - \dot{Q}_{21}(t) = \dot{Q}(t) - A k_{21} (T_1(t) - T_2(t)).$$

Durch Umstellen der Gleichung erhält man daraus

$$\dot{T}_1(t) = -\frac{k_{21}}{c_1 \rho_1 d_1} T_1(t) + \frac{k_{21}}{c_1 \rho_1 d_1} T_2(t) + \frac{1}{c_1 \rho_1 d_1 A} u(t),$$

wobei für  $\dot{Q}(t)$  die Stellgröße  $u(t)$  eingesetzt wurde. Für die Temperatur des Werkstückhalters gilt dementsprechend

$$\dot{T}_2(t) = \frac{k_{21}}{c_2 \rho_2 d_2} T_1(t) - \frac{k_{u2} + k_{21}}{c_2 \rho_2 d_2} T_2(t) + \frac{k_{u2}}{c_2 \rho_2 d_2} T_u$$

mit

$$k_{u2} = \frac{1}{\frac{d_2}{2\lambda_2}}.$$

Zusammengefasst ergibt sich ein affines Zustandsraummodell, das nach Linearisierung um den von der Eingangsgröße  $\bar{u}$  und der Umgebungstemperatur  $T_u$  abhängigen Arbeitspunkt  $\bar{T}_1, \bar{T}_2$  auf das lineare Zustandsraummodell

$$\text{Ofen : } \begin{cases} \begin{pmatrix} \dot{T}_1(t) \\ \dot{T}_2(t) \end{pmatrix} = \begin{pmatrix} -\frac{k_{21}}{c_1 \rho_1 d_1} & \frac{k_{21}}{c_1 \rho_1 d_1} \\ \frac{k_{21}}{c_2 \rho_2 d_2} & -\frac{k_{u2} + k_{21}}{c_2 \rho_2 d_2} \end{pmatrix} \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{c_1 \rho_1 d_1 A} \\ 0 \end{pmatrix} u(t) \\ y(t) = (0 \ 1) \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix} \end{cases} \quad (5.10)$$

führt, bei dem die Temperaturen die Abweichungen von den Arbeitspunktwerten beschreiben (für die Linearisierung affiner Modelle siehe Aufg. 3.7 auf S. 114).

**Steuerbarkeitsanalyse.** Aus dem Modell (5.10) erhält man die Steuerbarkeitsmatrix

$$S_S = \begin{pmatrix} \frac{1}{c_1 \rho_1 d_1 A} & -\frac{k_{21}}{c_1^2 \rho_1^2 d_1^2 A} \\ 0 & \frac{k_{21}}{c_1 c_2 \rho_1 \rho_2 d_1 d_2 A} \end{pmatrix}.$$

Für alle sinnvollen physikalischen Parameter hat diese Matrix den vollen Rang. Der Industrieofen ist also vollständig steuerbar.

Im Weiteren wird gezeigt, wie die Umsteuerung von einem gegebenen Anfangszustand  $x_0$  zu einem gegebenen Endzustand  $x_e$  aussehen kann, wobei mit folgenden Parametern gerechnet wird [9]:

Werkstück aus Aluminium:

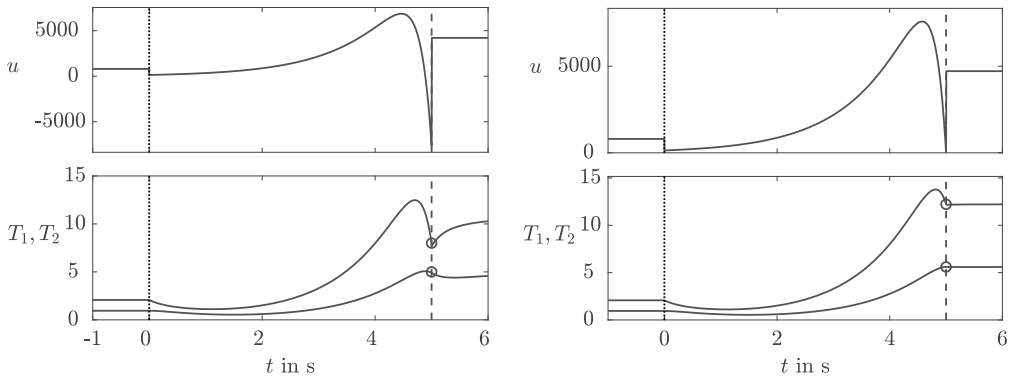
$\lambda_1 = 221 \frac{\text{W}}{\text{mK}}$	Wärmeleitfähigkeit von Aluminium
$\rho_1 = 2,7 \frac{\text{kg}}{\text{dm}^3}$	Dichte von Aluminium
$c_1 = 0,921 \frac{\text{kJ}}{\text{kg K}}$	spezifische Wärmekapazität von Aluminium
$d_1 = 3 \text{ mm}$	Werkstückdicke

Werkstückhalter aus Eisen:

$\lambda_2 = 67 \frac{\text{W}}{\text{mK}}$	Wärmeleitfähigkeit von Eisen
$\rho_2 = 7,87 \frac{\text{kg}}{\text{dm}^3}$	Dichte von Eisen
$c_2 = 0,465 \frac{\text{kJ}}{\text{kg K}}$	spezifische Wärmekapazität von Eisen
$d_2 = 5 \text{ mm}$	Dicke des Werkstückhalters
$A = 3,14 \text{ dm}^2$	Fläche von Werkstück und Werkstückhalter.

Der Wärmestrom  $u(t) = \dot{Q}(t)$  wird in Watt und die Zeit in Sekunden gemessen. Mit diesen Parametern erhält man das Zustandsraummodell

$$\text{Ofen : } \begin{cases} \begin{pmatrix} \dot{T}_1(t) \\ \dot{T}_2(t) \end{pmatrix} = \begin{pmatrix} -3,04 & 3,04 \\ 1,24 & -2,70 \end{pmatrix} \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix} + \begin{pmatrix} 0,0043 \\ 0 \end{pmatrix} u(t) \\ y(t) = (0 \ 1) \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix}. \end{cases} \quad (5.11)$$



**Abb. 5.4:** Steuerung des Industrieofens von  $\mathbf{x}_0 = (2,07 \ 0,95)^T$  nach  $\mathbf{x}_e = (8 \ 5)^T$  (links) bzw.  $\mathbf{x}_e = (12,22 \ 5,6)^T$  (rechts);  $u$  in W,  $T_1$  und  $T_2$  in K

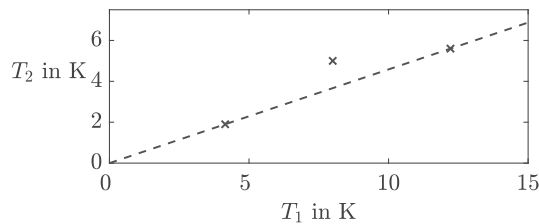
Der linke Teil von Abb. 5.4 zeigt die Situation, dass der Ofen für  $t < 0$  mit dem konstanten Wärmestrom von 800 W betrieben wurde, wofür sich der Anfangszustand

$$\mathbf{x}_0 = \begin{pmatrix} T_1(0) \\ T_2(0) \end{pmatrix} = \begin{pmatrix} 2,07 \\ 0,95 \end{pmatrix}$$

eingestellt hat (alle Größen sind Abweichungen von den Arbeitspunktwerten). Durch Vorgabe einer geeigneten Steuerung  $u_{[0,5]}$  soll der Zustand

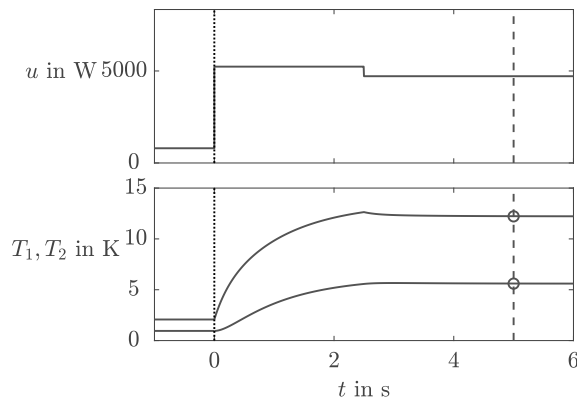
$$\mathbf{x}_e = \begin{pmatrix} T_1(5) \\ T_2(5) \end{pmatrix} = \begin{pmatrix} 8 \\ 5 \end{pmatrix}$$

erreicht werden. Die Stellgröße wurde entsprechend Gl. (5.7) bestimmt. Ihr Verlauf ist im linken oberen Teil von Abb. 5.4 zu sehen. Darunter ist der Verlauf der beiden Temperaturen gezeigt. Erwartungsgemäß stellen sich zur Zeit  $t_e = 5$  die beiden vorgegebenen Temperaturen ein, die in der Abbildung durch die beiden Kreise gekennzeichnet sind.



**Abb. 5.5:** Zustände, in denen der Ofen verharren kann

Der Ofen verlässt den Zustand  $\mathbf{x}_e$  jedoch sofort wieder, obwohl die Stellgröße  $u(t)$  auf den konstanten Wert gesetzt wird, bei dem die gemessene Temperatur (nach dem Einschwingen in den stationären Zustand) den Wert  $T_2(\infty) = 5$  hat. Der Grund dafür liegt darin, dass diejenigen Zustände, in denen der Ofen bei einer geeigneten Wahl der Heizleistung verbleiben kann, entsprechend Gl. (5.8) auf einer Linie im zweidimensionalen Zustandsraum liegen (Abb. 5.5). Der in Abb. 5.4 verwendete Zustand  $\mathbf{x}_e$  ist durch das nicht auf der Linie liegende Kreuz gekennzeichnet. Deshalb hat der Ofen diesen Zielzustand sofort wieder verlassen.



**Abb. 5.6:** Zweiter Stellgrößenverlauf zur Steuerung des Industrieofens  
von  $\mathbf{x}_0 = (2,07 \ 0,95)^T$  in  $\mathbf{x}_e = (12,22 \ 5,6)^T$

Wenn man einen Zielzustand wählt, der die Gl. (5.8) für ein geeignet gewähltes  $\bar{u}$  erfüllt und folglich auf der in Abb. 5.5 gezeigten Linie liegt, so kann man den Ofen in den Zielzustand steuern und ihn dort



halten. Der in Abb. 5.4 rechts dargestellte Stellgrößenverlauf wurde wieder mit Hilfe von Gl. (5.7) bestimmt und  $u(t) = \bar{u}$  für  $t > 5$  entsprechend Gl. (5.8) so festgelegt, dass der Endzustand beibehalten wird.

Abbildung 5.6 zeigt, dass dies nicht der einzige Stellgrößenverlauf ist, sondern dass man auch mit stückweise konstanten Stellgrößen die vorgegebene Umsteuerung realisieren kann. In diesem Beispiel erhält man dabei sogar einen wesentlich ruhigeren Verlauf für die Stellgröße und die beiden Temperaturen.  $\square$

## 5.3 Beobachtbarkeit linearer Systeme

### 5.3.1 Beobachtbarkeitskriterium

Die Beobachtbarkeit dynamischer Systeme wird in diesem Abschnitt für lineare Systeme

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) \end{cases} \quad (5.12)$$

untersucht. Das KALMAN-Kriterium bezieht sich auf die Beobachtbarkeitsmatrix

$$\mathbf{S}_B = \begin{pmatrix} \mathbf{c}^T \\ \mathbf{c}^T \mathbf{A} \\ \mathbf{c}^T \mathbf{A}^2 \\ \vdots \\ \mathbf{c}^T \mathbf{A}^{n-1} \end{pmatrix}, \quad (5.13)$$

in der die Zeilenvektoren  $\mathbf{c}^T, \mathbf{c}^T \mathbf{A}, \dots, \mathbf{c}^T \mathbf{A}^{n-1}$  untereinander geschrieben sind.

#### **Satz 5.2 (Beobachtbarkeitskriterium)**

*Das System (5.12) ist genau dann vollständig beobachtbar, wenn die Beobachtbarkeitsmatrix  $\mathbf{S}_B$  den Rang  $n$  hat:*

$$\text{Rang } \mathbf{S}_B = n. \quad (5.14)$$

Um zu zeigen, dass man unter der angegebenen Beobachtbarkeitsbedingung tatsächlich den Zustand berechnen kann, wird aus dem Modell (5.12) das folgende Gleichungssystem abgeleitet:

$$\begin{aligned} y(t) &= \mathbf{c}^T \mathbf{x}(t) \\ \dot{y}(t) &= \mathbf{c}^T \dot{\mathbf{x}}(t) = \mathbf{c}^T \mathbf{A} \mathbf{x}(t) + \mathbf{c}^T \mathbf{b} u(t) \\ \ddot{y}(t) &= \mathbf{c}^T \ddot{\mathbf{x}}(t) = \mathbf{c}^T \mathbf{A}^2 \mathbf{x}(t) + \mathbf{c}^T \mathbf{A} \mathbf{b} u(t) + \mathbf{c}^T \mathbf{b} \dot{u}(t) \\ &\vdots \\ \frac{d^{n-1}y(t)}{dt^{n-1}} &= \mathbf{c}^T \mathbf{A}^{n-1} \mathbf{x}(t) + \mathbf{c}^T \mathbf{A}^{n-2} \mathbf{b} u(t) + \dots + \mathbf{c}^T \mathbf{b} \frac{d^{n-2}u(t)}{dt^{n-2}}. \end{aligned}$$

Es kann als

$$\begin{pmatrix} y(t) \\ \dot{y}(t) \\ \ddot{y}(t) \\ \vdots \\ \frac{d^{n-1}y(t)}{dt^{n-1}} \end{pmatrix} = \begin{pmatrix} \mathbf{c}^T \\ \mathbf{c}^T \mathbf{A} \\ \mathbf{c}^T \mathbf{A}^2 \\ \vdots \\ \mathbf{c}^T \mathbf{A}^{n-1} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ \mathbf{c}^T \mathbf{b} & 0 & 0 & \cdots & 0 \\ \mathbf{c}^T \mathbf{A} \mathbf{b} & \mathbf{c}^T \mathbf{b} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}^T \mathbf{A}^{n-2} \mathbf{b} & \mathbf{c}^T \mathbf{A}^{n-3} \mathbf{b} & \mathbf{c}^T \mathbf{A}^{n-4} \mathbf{b} & \cdots & \mathbf{c}^T \mathbf{b} \end{pmatrix} \begin{pmatrix} u(t) \\ \dot{u}(t) \\ \ddot{u}(t) \\ \vdots \\ \frac{d^{n-2}u(t)}{dt^{n-2}} \end{pmatrix}$$

zusammengefasst und durch

$$\tilde{\mathbf{y}}(t) = \mathbf{S}_B \mathbf{x}(t) + \mathbf{H} \tilde{\mathbf{u}}(t)$$

abgekürzt werden. Dabei geben die beiden Vektoren  $\tilde{\mathbf{y}}(t)$  und  $\tilde{\mathbf{u}}(t)$  den Verlauf der Ausgangs- bzw. Eingangsgröße wieder, indem Sie den Wert sowie die Ableitungen bis zur Ordnung  $n - 1$  bzw.  $n - 2$  beider Größen zur Zeit  $t$  enthalten.

Wenn das System vollständig beobachtbar ist, so ist die Beobachtbarkeitsmatrix  $\mathbf{S}_B$  invertierbar und man erhält die Beziehung

$$\mathbf{x}(t) = \mathbf{S}_B^{-1} (\tilde{\mathbf{y}}(t) - \mathbf{H} \tilde{\mathbf{u}}(t)). \quad (5.15)$$

Diese Gleichung zeigt, dass es unter der Bedingung (5.14) möglich ist, aus den Funktionen  $y(t)$  und  $u(t)$  den Zustand  $\mathbf{x}(t)$  zu berechnen.

Technisch ist die hier beschriebene Vorgehensweise freilich nicht anwendbar, denn für die Bildung des Messvektors  $\tilde{\mathbf{y}}(t)$  muss der Verlauf der Ausgangsgröße in einem Intervall  $[0, t_e]$  um den Zeitpunkt  $t$  so genau gemessen werden, dass die Werte der ersten  $n - 1$  Ableitungen von  $y(t)$  zum Zeitpunkt  $t$  bekannt sind. Bei jedem Differenzieren der Messkurven wird auch das dem Nutzsignal  $y(t)$  überlagerte Messrauschen differenziert, wobei unter typischen praktischen Randbedingungen spätestens ab der zweiten Ableitung das erhaltene Ergebnis stärker vom Messrauschen als von der Messgröße abhängt. Im Kap. 9 wird deshalb ein anderer Weg angegeben, bei dem die Rekonstruktion von  $\mathbf{x}(t)$  aus  $y(t)$  mit einem Beobachter vorgenommen wird, der diese Differentiation vermeidet. Im Folgenden werden zwei andere Methoden aufgezeigt, um den Zustand auf eine auch numerisch günstige Weise zu berechnen.

### 5.3.2 Berechnung des Anfangszustands aus $n$ Messwerten

Die erste Methode zur Berechnung von  $\mathbf{x}_0$  verwendet an Stelle der Ableitungen der Ausgangsgröße  $n$  Messpunkte, die zu den im Intervall  $[0, t_e]$  liegenden Zeitpunkten  $t_1, t_2, \dots, t_n$  ermittelt werden. Es wird zunächst angenommen, dass das System eine verschwindende Eingangsgröße

hat ( $u = 0$ ) und man folglich die freie Bewegung  $y_{\text{frei}}(t)$  misst. Schreibt man die Ausgangsgleichung für  $n$  Messwerte untereinander, so erhält man das Gleichungssystem

$$\begin{aligned} y_{\text{frei}}(t_1) &= \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_1} \mathbf{x}_0 \\ y_{\text{frei}}(t_2) &= \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_2} \mathbf{x}_0 \\ &\vdots \\ y_{\text{frei}}(t_n) &= \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_n} \mathbf{x}_0. \end{aligned}$$

Die linke Seite kann zu einem Vektor zusammengefasst werden, wodurch die Beziehung

$$\begin{pmatrix} y_{\text{frei}}(t_1) \\ y_{\text{frei}}(t_2) \\ \vdots \\ y_{\text{frei}}(t_n) \end{pmatrix} = \begin{pmatrix} \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_1} \\ \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_2} \\ \vdots \\ \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_n} \end{pmatrix} \mathbf{x}_0 \quad (5.16)$$

entsteht, auf deren rechter Seite die  $(n \times n)$ -Matrix

$$\mathbf{M} = \begin{pmatrix} \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_1} \\ \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_2} \\ \vdots \\ \mathbf{c}^T \mathbf{e}^{\mathbf{A}t_n} \end{pmatrix} \quad (5.17)$$

steht. Gleichung (5.16) kann nach  $\mathbf{x}_0$  aufgelöst werden, wenn die Matrix  $\mathbf{M}$  regulär ist

$$\mathbf{x}_0 = \mathbf{M}^{-1} \begin{pmatrix} y_{\text{frei}}(t_1) \\ y_{\text{frei}}(t_2) \\ \vdots \\ y_{\text{frei}}(t_n) \end{pmatrix}, \quad (5.18)$$

wodurch man eine Berechnungsvorschrift für  $\mathbf{x}_0$  erhält.

Wenn die Beobachtbarkeitsbedingung (5.14) erfüllt ist, können  $n$  Zeitpunkte  $t_i$  so festgelegt werden, dass die Matrix  $\mathbf{M}$  invertierbar ist. In den meisten Anwendungsfällen erhält man bei willkürlich ausgewählten Zeitpunkten  $t_i$  eine reguläre Matrix. Sollten  $n$  zufällig ausgewählte Zeitpunkte zu einer singulären Matrix  $\mathbf{M}$  führen, so muss man lediglich einige Messzeitpunkte verschieben, um auf eine reguläre Matrix zu kommen.

Um diese Methode für Systeme mit Anregung ( $u \neq 0$ ) anwenden zu können, muss man mit Hilfe des Modells aus der gemessenen Ausgangsgröße  $y(t_i)$  die freie Bewegung ermitteln. Durch Umstellung der Bewegungsgleichung (4.9) erhält man

$$y_{\text{frei}}(t_i) = y(t_i) - \int_0^{t_i} \mathbf{c}^T \mathbf{e}^{\mathbf{A}(t-\tau)} \mathbf{b} u(\tau) \, d\tau. \quad (5.19)$$

Bei bekanntem Verlauf  $u_{[0,t_e]}$  kann man das rechts stehende Integral ausrechnen und dann mit Gl. (5.18) den gesuchten Anfangszustand  $\mathbf{x}_0$  bestimmen.

### 5.3.3 Bestimmung des Anfangszustands mit Hilfe der gramschen Beobachtbarkeitsmatrix

Die zweite Methode beruht auf einer Beziehung, die ähnlich zu Gl. (5.7) ist. In Analogie zur Matrix  $\mathbf{W}_S$  in Gl. (5.6) wird die *gramsche Beobachtbarkeitsmatrix* definiert:

$$\mathbf{W}_B = \int_0^{t_e} e^{\mathbf{A}^T t} \mathbf{c} \mathbf{c}^T e^{\mathbf{A} t} dt. \quad (5.20)$$

Diese Matrix ist für vollständig beobachtbare Systeme für eine beliebige Endzeit  $t_e$  positiv definit und deshalb invertierbar. Der gesuchte Anfangszustand  $\mathbf{x}_0$  kann dann aus der Beziehung

$$\mathbf{x}_0 = \mathbf{W}_B^{-1} \int_0^{t_e} e^{\mathbf{A}^T t} \mathbf{c} y_{\text{frei}}(t) dt \quad (5.21)$$

bestimmt werden, wovon man sich durch Einsetzen der Eigenbewegung

$$y_{\text{frei}}(t) = \mathbf{c}^T e^{\mathbf{A} t} \mathbf{x}_0$$

in Gl. (5.21) überzeugen kann.  $y_{\text{frei}}(t)$  wird entsprechend Gl. (5.19) mit  $t$  anstelle von  $t_i$  aus  $u_{[0, t_e]}$  und  $y_{[0, t_e]}$  berechnet.

Diese Lösung des Beobachtungsproblems zeigt, dass der Anfangszustand  $\mathbf{x}_0$  aus dem Verlauf  $u_{[0, t_e]}$  und  $y_{[0, t_e]}$  der Eingangsgröße und der Ausgangsgröße berechnet werden kann, wobei das Zeitintervall  $[0, t_e]$  beliebig kurz sein kann. Jedoch wird aus numerischen Gründen die Genauigkeit umso schlechter, je kleiner der Horizont  $t_e$  ist.

#### Beispiel 5.2 Beobachtbarkeit eines Industrieofens

Bei dem im Beispiel 5.1 betrachteten Industrieofen kann nur die Temperatur des Werkstückhalters gemessen werden, weil das Anbringen eines Temperatursensors am Werkstück zu aufwändig wäre. Es ergibt sich deshalb die Frage, ob die Werkstücktemperatur  $T_2(t)$  aus der Messgröße  $T_1(t)$  rekonstruiert werden kann.

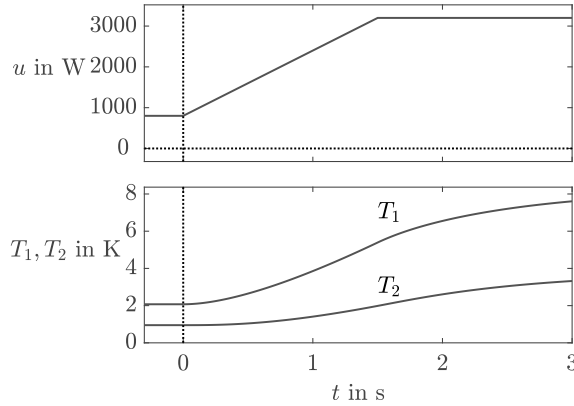
Um die praktische Bedeutung dieser Aufgabe hervorzuheben, sei zunächst angemerkt, dass auch im statischen Zustand die Temperaturen  $\bar{T}_1$  und  $\bar{T}_2$  von Werkstück und Werkstückhalter nicht übereinstimmen, weil auf Grund des Wärmestroms vom Werkstück zur Umgebung ein Temperaturgefälle auftritt, so dass  $\bar{T}_2 < \bar{T}_1$  gilt. Dies ist z. B. in Abb. 5.4 auf S. 161 zu sehen. Dort befindet sich der Ofen für  $t < 0$  im statischen Zustand, in dem  $\bar{T}_2$  erheblich von  $\bar{T}_1$  abweicht. Man könnte jedoch  $\bar{T}_1$  aus  $\bar{T}_2$  unter Verwendung des in Gl. (5.9) eingeführten Vektors  $\mathbf{k}_{sx} = (k_{sx1} \ k_{sx2})^T$  bestimmen:

$$\bar{T}_2 = \bar{T}_1 \frac{k_{sx2}}{k_{sx1}}.$$

Diese Beziehung gilt jedoch nur dann, wenn die Heizleistung konstant und der Ofen in den dazugehörigen statischen Endzustand eingeschwungen ist.

Die Beobachtbarkeit des Ofens macht es möglich, die Temperatur  $T_2(t)$  auch während dynamischer Übergangsvorgänge zu berechnen. Die Beobachtbarkeitsmatrix

$$\mathbf{S}_B = \begin{pmatrix} 0 & 1 \\ \frac{k_{21}}{c_2 \rho_2 d_2} & -\frac{k_{12} + k_{21}}{c_2 \rho_2 d_2} \end{pmatrix}$$



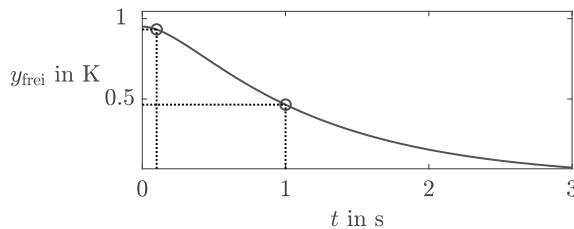
**Abb. 5.7:** Verhalten des Industrieofens bei ansteigender Heizleistung

hat für physikalisch sinnvolle Parameterwerte den vollen Rang, womit die vollständige Beobachtbarkeit bewiesen ist.

Um zu zeigen, dass man während eines Übergangsvorgangs den Anfangszustand  $x_0$  bestimmen kann, wird der Ofen für die in Abb. 5.7 oben dargestellte Eingangsgröße

$$u(t) = \begin{cases} 800 & t \leq 0 \\ 800 + 1600t & 0 < t \leq 1,5 \\ 3200 & t > 1,5 \end{cases}$$

betrachtet, für die sich der im unteren Teil der Abbildung gezeigte Temperaturverlauf einstellt. Gemessen werden kann nur die niedrigere der beiden Temperaturen:  $y(t) = T_2(t)$ .



**Abb. 5.8:** Eigenbewegung des Industrieofens

Mit Hilfe des Modells (5.11) kann aus dem Verlauf der Eingangsgröße  $u(t)$  die erzwungene Bewegung

$$y_{\text{erzw}}(t) = \int_0^t \begin{pmatrix} 0 & 1 \end{pmatrix} e^{\begin{pmatrix} -3,04 & 3,04 \\ 1,24 & -2,70 \end{pmatrix} (t-\tau)} \begin{pmatrix} 0,0043 \\ 0 \end{pmatrix} u(\tau) d\tau$$

und mit der Messgröße  $y(t)$  dann die Eigenbewegung  $y_{\text{frei}}(t) = y(t) - y_{\text{erzw}}(t)$  berechnet werden, die in Abb. 5.8 zu sehen ist. Aus dieser Kurve werden zwei willkürlich gewählte Punkte

$$y_{\text{frei}}(0,1) = 0,933 \quad \text{und} \quad y_{\text{frei}}(1) = 0,465$$

ausgewählt, womit man entsprechend Gl. (5.18) den Anfangszustand aus der Beziehung

$$\begin{pmatrix} T_1(0) \\ T_2(0) \end{pmatrix} = \begin{pmatrix} 0,097 & 0,768 \\ 0,123 & 0,238 \end{pmatrix}^{-1} \begin{pmatrix} 0,933 \\ 0,465 \end{pmatrix} = \begin{pmatrix} 2,09 \\ 0,95 \end{pmatrix}$$

erhält, was mit dem tatsächlich verwendeten Anfangszustand  $(2,073 \ 0,950)^T$  in guter Übereinstimmung ist.

Man kann auch Gl. (5.21) anwenden, wenn man zuvor die Matrix  $\mathbf{W}_B$  mit Hilfe von Gl. (5.20) für die hier verwendete Endzeit  $t_e = 3$  berechnet hat. Bei der numerischen Integration muss man erfahrungsgemäß eine sehr kleine Schrittweite verwenden, was bedeutet, dass man für die Temperatur des Werkstückhalters sehr viele Messwerte braucht. Für das Beispiel erhält man ein ähnlich gutes Ergebnis für die Anfangstemperaturen wie mit der ersten Methode. Aus den Anfangstemperaturen  $T_1(0)$  und  $T_2(0)$  sowie dem Verlauf  $u_{[0,3]}$  der Eingangsgröße kann man die gesuchte aktuelle Temperatur  $T_1(3)$  berechnen.  $\square$

**Aufgabe 5.1\*** Berechnung des aktuellen Zustands aus  $n$  Messwerten

Im Abschn. 5.3.2 wurde Gl. (5.18) zur Bestimmung des Anfangszustands  $\mathbf{x}_0$  aus  $n$  Messwerten abgeleitet. Bei vielen Anwendungen ist aber nicht der Anfangszustand, sondern der zur Zeit  $t_e$  vorhandene Systemzustand interessant. Wie kann man den Zustand  $\mathbf{x}(t_e)$  aus dem Verlauf  $u_{[0,t_e]}$  der Eingangsgröße und dem Verlauf  $y_{[0,t_e]}$  der Ausgangsgröße berechnen?  $\square$

## 5.4 Strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit

### 5.4.1 Strukturgraph

Betrachtet man die Steuerbarkeit und die Beobachtbarkeit von einem etwas abstrakteren Standpunkt, so kann man Folgendes vermuten:

**Vermutung:** Ein System ist vollständig steuerbar, wenn es Signalwege vom Eingang zu allen Zustandsvariablen gibt, so dass alle Zustandsgrößen vom Eingang beeinflusst werden können. Ein System ist vollständig beobachtbar, wenn von allen Zustandsvariablen ein Signalweg zum Ausgang führt, so dass die Ausgangsgröße von allen Zustandsgrößen abhängt.

Die folgenden Untersuchungen werden zeigen, dass diese Vermutung (fast) richtig ist.

Die Vermutung wurde aufgrund einer Betrachtungsweise des Systems formuliert, bei der nur zwischen vorhandenen und nicht vorhandenen Signalkopplungen unterschieden wird. Es ist deshalb naheliegend, für die folgenden Steuerbarkeits- und Beobachtbarkeitsuntersuchung den Signalflussgraphen heranzuziehen. Dort sind alle Signale durch Knoten repräsentiert und genau dann durch gerichtete Kanten miteinander verbunden, wenn sie einen direkten Einfluss aufeinander ausüben. Für den durch das Zustandsraummodell (5.10) auf S. 160 beschriebenen Industrieofen ist der Signalflussgraph in Abb. 5.9 dargestellt.

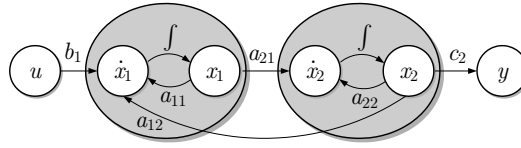


Abb. 5.9: Signalflussgraph des Industrieofens

Da die abgeleiteten Größen  $\dot{x}_i$  stets mit den nicht abgeleiteten Größen  $x_i$  durch die Integration verbunden sind und damit stets ein Signalweg zwischen beiden Größen vorhanden ist, können sie im Graphen zusammengefasst werden. Außerdem ist es für die nachfolgenden Betrachtungen unerheblich, mit welchen Parametern die Kanten des Graphen gewichtet sind, denn es ist nur wichtig zu wissen, ob diese Wichtungen von null verschieden sind oder nicht. Die Zusammenfassung der Knoten und das Weglassen der Wichtungen ergibt einen neuen Graphen, der *Strukturgraph* des Systems genannt wird und der für den Industrieofen in Abb. 5.10 dargestellt ist.

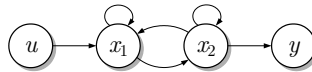


Abb. 5.10: Strukturgraph des Industrieofens

Sieht man sich den Strukturgraphen genauer an, so enthält er genau dann Kanten, wenn die entsprechenden Elemente in der Matrix  $\mathbf{A}$ , im Vektor  $\mathbf{b}$  oder im Vektor  $\mathbf{c}^T$  von null verschieden sind. Man bildet deshalb von den im Zustandsraummodell vorkommenden Matrizen und Vektoren Strukturmatrizen, in denen nur zwischen verschwindenden und nicht verschwindenden Elementen unterschieden wird und deren Elemente deshalb durch eine Null oder einen Stern dargestellt sind. Ein Stern bedeutet, dass das betreffende Element nicht für alle in Frage kommenden Parameterwerte verschwindet. Für das Modell (5.10) des Industrieofens erhält man dabei folgende Strukturmatrizen, wobei die eckigen Klammern die Bildung der \*-Elemente anzeigen:

$$[\mathbf{A}] = \begin{pmatrix} * & * \\ * & * \end{pmatrix}, \quad [\mathbf{b}] = \begin{pmatrix} * \\ 0 \end{pmatrix}, \quad [\mathbf{c}^T] = (0 \quad *).$$

Beispielsweise bedeutet das als Stern geschriebene Element  $a_{21}$ , dass es physikalisch sinnvolle Parameter des Industrieofens gibt, für die  $a_{21} \neq 0$  ist. Deshalb gibt es im Strukturgraphen eine Kante von  $x_1$  nach  $x_2$ . Die Null in der Position von  $b_2$  besagt, dass dieses Element des Vektors  $\mathbf{b}$  für alle physikalisch sinnvollen Parameterwerte des Ofens verschwindet und es im Strukturgraphen deshalb keine Kante von  $u$  nach  $x_2$  gibt.

Dass die Struktur und nicht die konkreten Parameterwerte eines Systems für die Steuerbarkeits- und Beobachtbarkeitseigenschaften maßgebend ist, erkennt man auch an den Sätzen 5.1 auf S. 157 und 5.2. Schreibt man die Bedingung (5.5) für ein System zweiter Ordnung in Abhängigkeit von den Elementen  $s_{ij}$  der Steuerbarkeitsmatrix

$$\det \mathbf{S}_S = s_{11}s_{22} - s_{12}s_{21} \neq 0,$$

dann sieht man, dass die Steuerbarkeitsbedingung beispielsweise für

$$s_{11} = 0, \quad s_{12} \neq 0 \quad \text{und} \quad s_{21} \neq 0$$

erfüllt ist und zwar unabhängig von den konkreten Werten von  $s_{12}$  und  $s_{21}$ . Es reicht also zu wissen, dass die Matrix  $S_S$  die folgende Struktur besitzt:

$$[S_S] = \begin{pmatrix} 0 & * \\ * & * \end{pmatrix}.$$

#### 5.4.2 Definition und Kriterien für die strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit

Mit dem Strukturgraphen kann man die Steuerbarkeit und Beobachtbarkeit aller derjenigen Systeme gleichzeitig untersuchen, die denselben Strukturgraphen haben. Man definiert deshalb die strukturelle Steuerbarkeit und die strukturelle Beobachtbarkeit als Eigenschaften der Klasse

$$\mathcal{S}(S_A, S_b, S_{c^T}) = \{(A, b, c^T) \mid [A] = S_A, [b] = S_b, [c^T] = S_{c^T}\}$$

aller Systeme, deren Parameter  $A$ ,  $b$  und  $c^T$  zu denselben Strukturmatrizen  $S_A$ ,  $S_b$  und  $S_{c^T}$  passen.

##### **Definition 5.3 (Strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit)**

*Eine Klasse von Systemen  $\mathcal{S}(S_A, S_b, S_{c^T})$  heißt strukturell steuerbar bzw. strukturell beobachtbar, wenn es mindestens ein System (5.12) mit  $[A] = S_A$ ,  $[b] = S_b$ ,  $[c^T] = S_{c^T}$  gibt, das vollständig steuerbar bzw. vollständig beobachtbar ist.*

Diese Definition hat die wichtige Konsequenz:

Die strukturelle Steuerbarkeit ist eine *notwendige* Bedingung für die (numerische) Steuerbarkeit und die strukturelle Beobachtbarkeit eine notwendige Bedingung für die (numerische) Beobachtbarkeit.

Wenn man also mit Hilfe der im Folgenden angegebenen Verfahren feststellt, dass die betrachtete Systemklasse nicht strukturell steuerbar oder nicht strukturell beobachtbar ist, dann bedeutet das, dass *kein* System (5.12), das zu dieser Systemklasse gehört, vollständig steuerbar bzw. vollständig beobachtbar ist.

Andererseits folgt aus den strukturellen Eigenschaften nicht, dass auch die entsprechenden numerischen Eigenschaften gelten. Die Elemente von  $A$ ,  $b$  und  $c^T$  können nämlich Werte annehmen, für die die Steuerbarkeitsmatrix oder die Beobachtbarkeitsmatrix einen Rangabfall hat, obwohl die im Folgenden behandelten strukturellen Bedingungen erfüllt sind.



Auf den ersten Blick möchte man bezweifeln, dass die strukturellen Eigenschaften eine große Aussagekraft für die entsprechenden numerischen Eigenschaften haben, denn in der Definition 5.3 wird nur die Existenz eines einzigen vollständig steuerbaren bzw. vollständig beobachtbaren Systems in der betrachteten Klasse gefordert, damit die strukturelle Eigenschaft gilt. Diese Vermutung ist jedoch falsch, denn man kann den folgenden Sachverhalt zeigen:

|| In einer Klasse  $\mathcal{S}$  strukturell steuerbarer und beobachtbarer Systeme sind „fast alle“ Systeme  $(\mathbf{A}, \mathbf{b}, \mathbf{c}^T) \in \mathcal{S}$  vollständig steuerbar und vollständig beobachtbar.

Das heißt, dass sich die strukturellen und die numerischen Eigenschaften nur für ausgewählte Systemparameter unterscheiden können. Mathematisch gesprochen heißt das Attribut „fast alle“, dass diejenigen Systeme, für die beide Eigenschaften nicht zusammenfallen, auf einer Hyperebene im Parameterraum liegen (vgl. die Diskussion im Beispiel 5.3). Diese Tatsache zeigt die Bedeutung der strukturellen Analyse.

**Kriterien für die strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit.** Um für eine gegebene Systemklasse zu überprüfen, ob sie die Eigenschaften der strukturellen Steuerbarkeit und Beobachtbarkeit besitzt, zeichnet man als erstes den Strukturgraphen und überprüft, ob jeder Zustandsknoten auf einem Pfad vom Eingangsknoten erreichbar ist und ob es von jedem Zustandsknoten einen Pfad zum Ausgangsknoten gibt. Damit wird der eingangs aufgestellten Vermutung entsprechend überprüft, ob es Signalwege vom Eingang zu allen Zustandsvariablen bzw. von allen Zustandsvariablen zum Ausgang gibt. Wenn diese Pfade existieren, nennt man die durch den Strukturgraphen dargestellte Systemklasse  $\mathcal{S}$  *eingangsverbunden* und *ausgangsverbunden*.

Die Systemklasse muss jedoch noch eine zweite Bedingung erfüllen, um strukturell steuerbar und strukturell beobachtbar zu sein. Man muss prüfen, dass der strukturelle Rang der Matrizen

$$(\mathbf{S}_A \ \mathbf{S}_b) \quad \text{und} \quad \begin{pmatrix} \mathbf{S}_A \\ \mathbf{S}_{c^T} \end{pmatrix}$$

gleich  $n$  ist. In diesen Matrizen steht neben bzw. unter der Strukturmatrix  $\mathbf{S}_A$  der Vektor  $\mathbf{S}_b$  bzw. der Vektor  $\mathbf{S}_{c^T}$ . Der strukturelle Rang einer Strukturmatrix ist die maximale Anzahl von \*-Elementen, die man so auswählen kann, dass sie in getrennten Zeilen und Spalten stehen. Wenn der strukturelle Rang einer Strukturmatrix gleich  $n$  ist, so haben fast alle numerischen Matrizen, die auf die betrachtete Strukturmatrix führen, den Rang  $n$ . Der strukturelle Rang wird mit „s-Rang“ abgekürzt.

**Satz 5.3 (Kriterium für strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit)**

Eine Klasse  $\mathcal{S}(\mathbf{S}_A, \mathbf{S}_b, \mathbf{S}_{c^T})$  von Systemen ist genau dann strukturell steuerbar, wenn

1.  $\mathcal{S}$  eingangsverbunden ist und
2. die folgende Bedingung erfüllt ist:

$$\text{s-Rang } (\mathbf{S}_A \quad \mathbf{S}_b) = n. \quad (5.22)$$

Die Klasse  $\mathcal{S}$  ist genau dann strukturell beobachtbar, wenn

1.  $\mathcal{S}$  ausgangsverbunden ist und
2. die folgende Bedingung erfüllt ist:

$$\text{s-Rang } \begin{pmatrix} \mathbf{S}_A \\ \mathbf{S}_{c^T} \end{pmatrix} = n. \quad (5.23)$$

**Beispiel 5.3** Strukturelle Analyse des Industrieofens

Für den durch das Zustandsraummodell (5.10) beschriebenen Industrieofen ist der Strukturgraph in Abb. 5.10 gezeichnet. Dieser Strukturgraph gilt für den Industrieofen unabhängig von den Werten der Parameter  $\lambda_1, \lambda_2, d_1, d_2$  usw. Entscheidend für den Strukturgraphen ist ja nur, dass zwei Elemente im Vektor  $\mathbf{b}$  bzw. im Vektor  $\mathbf{c}^T$  gleich null sind, so dass der Graph zwischen  $u$  und  $x_2$  bzw.  $x_1$  und  $y$  keine Kanten besitzt. Die strukturellen Aussagen gelten deshalb unabhängig von den Parameterwerten und ändern sich nicht, wenn Ofenparameter verändert werden.

Die durch den Graphen beschriebene Systemklasse ist eingangsverbunden und ausgangsverbunden. Die Rangbedingungen sind erfüllt, wie die folgenden Matrizen zeigen, in denen die für die Rangbestimmung verwendeten Elemente durch einen Punkt  $\bullet$  gekennzeichnet sind:

$$\text{s-Rang } \begin{pmatrix} * & * & \bullet \\ * & \bullet & 0 \end{pmatrix} = 2, \quad \text{s-Rang } \begin{pmatrix} * & * \\ \bullet & * \\ 0 & \bullet \end{pmatrix} = 2.$$

Die Systemklasse, zu der der Industrieofen gehört, ist folglich strukturell steuerbar und strukturell beobachtbar.

**Diskussion.** Das hier erhaltene strukturelle Ergebnis bringt keine neuen Erkenntnisse über den Industrieofen, weil in den Beispielen 5.1 und 5.2 bereits die vollständige Steuerbarkeit und vollständige Beobachtbarkeit festgestellt wurden. Da die strukturellen Eigenschaften notwendige Bedingungen für die numerischen Eigenschaften sind, ist auf Grund dieser Ergebnisse bekannt, dass die Systemklasse des Industrieofens strukturell steuerbar und strukturell beobachtbar ist. In der Praxis prüft man jedoch die strukturellen Eigenschaften i. Allg. zuerst, weil dies auch ohne Kenntnis der Parameterwerte möglich ist. Erst wenn man sich von den strukturellen Eigenschaften überzeugt hat, stellt man die Steuerbarkeitsmatrix und die Beobachtbarkeitsmatrix auf und überprüft auch die numerischen Eigenschaften, die von den konkreten Parameterwerten abhängen.

Dass der strukturelle Rang einer Matrix mit dem numerischen Rang fast aller Matrizen derselben Struktur übereinstimmt, kann man sich an dem Beispiel klar machen. Der strukturelle Rang wurde hier für die aus der Matrix

$$\begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & 0 \end{pmatrix}$$

entstehende Strukturmatrix  $([A] \ [b])$  geprüft, wobei festgestellt wurde, dass der s-Rang gleich zwei ist. Für die angegebene Matrix ist der (numerische) Rang ebenfalls gleich zwei, wenn die Determinante

$$\det \left( \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & 0 \end{pmatrix}^T \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & 0 \end{pmatrix} \right) = (a_{11}^2 + a_{12}^2 + b_1^2)(a_{21}^2 + a_{22}^2) - (a_{11}a_{21} + a_{12}a_{22})$$

nicht verschwindet. Wenn die beiden durch  $\bullet$  markierten Elemente  $b_1$  und  $a_{22}$  von null verschieden sind, ist der erste Summand positiv. Für feste Werte dieser beiden Elemente gibt es nur ganz spezielle Werte der restlichen Elemente, für die die Determinante verschwindet. Das heißt, dass der strukturelle Rang für „fast alle“ Matrizen der gegebenen Struktur mit dem numerischen Rang übereinstimmt. Die Werte der Matrixelemente, für die der Rang der numerischen Matrix kleiner als der strukturelle Rang ist, liegen auf einer Hyperebene im Raum der Parameter  $a_{11}$ ,  $a_{12}$ ,  $a_{21}$ ,  $a_{22}$  und  $b_1$ . Falls man zufälligerweise einen derartigen Parametersatz erhalten hat, muss man nur eines dieser Elemente geringfügig verändern, um den Rang der numerischen Matrix auf den strukturellen Rang anzuheben.  $\square$

#### Beispiel 5.4 Nicht steuerbares System, das die Rangbedingung aus Satz 5.3 verletzt

Von den beiden im Satz 5.3 angegebenen Bedingungen für die strukturelle Steuerbarkeit bzw. strukturelle Beobachtbarkeit ist bei Anwendungen meistens die erste Bedingung die „kritische“. Deshalb soll hier ein System angegeben werden, das zwar die erste Bedingung erfüllt, jedoch aufgrund der Verletzung der zweiten Bedingung die strukturellen (und damit auch die numerischen) Eigenschaften nicht besitzt.

Betrachtet wird die Parallelschaltung zweier Integratoren, die durch das Zustandsraummodell

$$\text{Integratoren : } \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} u(t) \\ y(t) = (c_1 \ c_2) \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} \end{cases}$$

beschrieben ist. Der Strukturgraph verbindet die Knoten für die Eingangsgröße  $u$  mit beiden Zustandsvariablen  $x_1$ ,  $x_2$ , die ihrerseits auf den Ausgang  $y$  wirken, so dass die ersten Bedingungen aus Satz 5.3 erfüllt sind. Die Rangbedingungen sind jedoch nicht erfüllt, wie man aus den zu untersuchenden Matrizen

$$([A] \ [b]) = \begin{pmatrix} 0 & 0 & * \\ 0 & 0 & * \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} [A] \\ [c^T] \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ * & * \end{pmatrix}$$

erkennt. Die Parallelschaltung zweier Integratoren ist also weder strukturell steuerbar noch strukturell beobachtbar. Wie man mit den Bedingungen (5.4) und (5.14) leicht nachrechnen kann, sind tatsächlich die betrachteten Integratoren für keine Parameterwerte für  $b_1$ ,  $b_2$ ,  $c_1$  und  $c_2$  vollständig steuerbar oder vollständig beobachtbar und deshalb die Klasse der zu den Integratoren strukturell äquivalenten Systeme entsprechend Definition 5.3 nicht strukturell steuerbar bzw. strukturell beobachtbar.

Das Beispiel zeigt, dass auch die jeweils zweite Bedingung aus Satz 5.3 die „kritische“ sein kann und erklärt darüber hinaus, warum es nicht ausreicht, bei der Analyse der strukturellen Eigenschaften nur die Eingangs- bzw. Ausgangsverbundenheit des Strukturgraphen zu überprüfen.  $\square$

**Aufgabe 5.2** Steuerbarkeit und Beobachtbarkeit linearer Systeme

Gegeben sind folgende Systeme in Zustandsraumdarstellung:

$$\Sigma_1 : \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} 1 & 0 \\ 4 & 3 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ 2 \end{pmatrix} u(t) \\ y(t) = (1 \ 0) \mathbf{x}(t) \end{cases}$$

$$\Sigma_2 : \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} 1 & 3 \\ 4 & 0 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ 2 \end{pmatrix} u(t) \\ y(t) = (1 \ 0) \mathbf{x}(t) \end{cases}$$

$$\Sigma_3 : \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} 1 & 3 & 2 \\ 0 & 3 & 1 \\ 0 & 0 & 2 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} u(t) \\ y(t) = (1 \ 0 \ 4) \mathbf{x}(t). \end{cases}$$

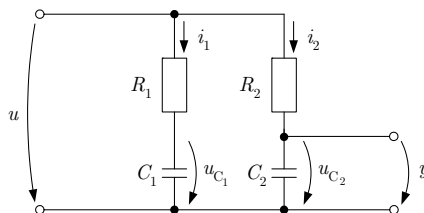
Prüfen Sie diese Systeme strukturell und numerisch auf Steuerbarkeit und Beobachtbarkeit. Wenn Sie ein nicht steuerbares oder nicht beobachtbares System finden, überlegen Sie sich, durch welche (möglichst „kleinen“) Veränderungen diese Systeme vollständig steuerbar bzw. beobachtbar gemacht werden können.  $\square$

**Aufgabe 5.3\*** Vollständige Steuerbarkeit und strukturelle Steuerbarkeit

Betrachten Sie das System  $\Sigma$  mit der Zustandsgleichung

$$\Sigma : \dot{\mathbf{x}}(t) = \begin{pmatrix} a_{11} & a_{12} \\ 0 & 0 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ b_2 \end{pmatrix} u(t),$$

in der die Parameter  $a_{11}$ ,  $a_{12}$  und  $b_2$  von null verschiedene Werte annehmen können. Untersuchen Sie, ob die Klasse  $\mathcal{S}$ , zu der dieses System gehört, strukturell steuerbar ist. Verwenden Sie für diese Untersuchung im ersten Schritt nur die Definition 5.3 und im zweiten Schritt die notwendigen und hinreichenden Bedingungen für die strukturelle Steuerbarkeit aus dem Satz 5.3.  $\square$



**Abb. 5.11:** RC-Schaltung

**Aufgabe 5.4\*** Steuerbarkeit und Beobachtbarkeit einer RC-Schaltung

Gegeben ist die Schaltung in Abb. 5.11 mit der Eingangsspannung  $u(t)$  und der Ausgangsspannung  $y(t)$ .

1. Stellen Sie das Zustandsraummodell der Schaltung auf.
2. Welche Kopplungen bestehen zwischen den beiden Zweigen der Schaltung? Wird die Schaltung vermutlich steuerbar und beobachtbar sein?
3. Untersuchen Sie die Steuerbarkeit und Beobachtbarkeit strukturell und numerisch. Gibt es Parameterwerte, für die sich die numerischen Eigenschaften von den strukturellen Eigenschaften unterscheiden?
4. Verändern Sie die Schaltung so, dass sie für alle Parameterwerte vollständig beobachtbar ist. Ist das System dann auch für alle Parameterwerte steuerbar? □

**Aufgabe 5.5\*** Konstruktion eines strukturell nicht steuerbaren Systems

Entsprechend Satz 5.3 muss ein System zwei Bedingungen erfüllen, um strukturell steuerbar zu sein: Es muss eingangsverbunden sein und die Rangbedingung (5.22) erfüllen. Geben Sie ein System an, dass zwar eingangsverbunden ist, aber die Rangbedingung verletzt und demzufolge nicht strukturell steuerbar ist. Wie schlägt sich die Tatsache, dass das System nicht steuerbar ist, in der Bewegungsgleichung nieder? □

**5.4.3 Strukturelle Analyse nichtlinearer Systeme**

Die bisherigen Betrachtungen können auf nichtlineare Systeme mit dem Zustandsraummodell (5.1) erweitert werden. Die Zustandsvariable  $x_i$  hat einen Einfluss auf die Zustandsvariable  $x_j$ , wenn  $\dot{x}_j$  von  $x_i$  abhängt, wenn also  $x_i$  in der Funktion  $g_j(\mathbf{x}, u)$  vorkommt. Ob dies der Fall ist, sieht man bei differenzierbaren Funktionen  $g_j$  an der Bedingung

$$\frac{\partial g_j(\mathbf{x}, u)}{\partial x_i} \neq 0. \quad (5.24)$$

Das Zeichen  $\neq$  bedeutet, dass die partielle Ableitung von  $g_j$  nach  $x_i$  nicht für alle Werte aller Argumente  $x_1, \dots, x_n, u$  verschwindet. In diesem Fall gibt es im Strukturgraphen eine Kante vom Knoten  $x_i$  zum Knoten  $x_j$ .

In gleicher Weise untersucht man den Einfluss der Eingangsgröße  $u$  und den Einfluss, den die Zustandsvariablen auf den Ausgang  $y$  haben, und zeichnet vom Knoten  $u$  zum Knoten  $x_j$  bzw. vom Knoten  $x_j$  zum Knoten  $y$  eine Kante, wenn

$$\frac{\partial g_j(\mathbf{x}, u)}{\partial u} \neq 0, \quad \text{und} \quad \frac{\partial h(\mathbf{x}, u)}{\partial x_j} \neq 0 \quad (5.25)$$

gilt. Mit dem Strukturgraphen können die Eingangsverbundenheit und die Ausgangsverbundenheit überprüft werden, die auch bei nichtlinearen Systemen notwendig für die Steuerbarkeit bzw. Beobachtbarkeit sind.

Die im Satz 5.3 angegebenen Rangbedingungen müssen jedoch noch auf nichtlineare Systeme übertragen werden, was zunächst eine Definition der strukturellen Steuerbarkeit und Beobachtbarkeit für nichtlineare Systeme erfordert, denn die Definition 5.3 bezieht sich auf Klassen linearer Systeme. Diese Betrachtungen sollen hier dadurch vereinfacht werden, dass für nichtlineare Systeme nicht die in den Definitionen 5.1 und 5.2 beschriebenen Eigenschaften gefordert werden, sondern die Steuerbarkeit und Beobachtbarkeit des linearisierten Modells untersucht werden soll.

Das linearisierte Modell gilt in der Umgebung eines Arbeitspunkts  $(\bar{x}, \bar{u}, \bar{y})$ . In diesem Modell treten die für den Arbeitspunkt berechneten Ableitungen (5.24) – (5.25) als Modellparameter auf (vgl. Gl. (3.42), (3.43), (3.44)). Solange diese Ableitungen von null verschieden sind, treten die entsprechenden Kanten im Strukturgraphen auf. Wenn eine Ableitung in einem Arbeitspunkt verschwindet, so verschwindet auch die zugehörige Kante. Auf diese Weise kann durch die hier beschriebene Erweiterung der strukturellen Analyse auf nichtlineare Systeme die strukturelle Steuerbarkeit bzw. strukturelle Beobachtbarkeit für alle Arbeitspunkte untersucht werden. Man muss nur darauf aufpassen, für welche Arbeitspunkte sich die Strukturmatrizen und folglich der Strukturgraph ändern.

## 5.5 Systemzerlegung entsprechend den Steuerbarkeits- und Beobachtbarkeitseigenschaften

Im Folgenden wird die Betrachtung von Steuerbarkeit und Beobachtbarkeit auf die Eingangs- bzw. Ausgangsverbundenheit des Systems beschränkt. Die Verbundenheit wird jetzt nicht auf das System als Ganzes, sondern auf die einzelnen Zustandsvariablen bezogen. Eine Zustandsvariable ist eingangsverbunden, wenn es einen Pfad vom Eingang zu dieser Zustandsvariablen gibt. Sie heißt ausgangsverbunden, wenn ein Pfad von ihr zum Ausgangsknoten existiert. Dementsprechend kann man die Menge der Zustandsvariablen in vier Teilmengen aufteilen:

- Zustandsvariable, die eingangs- und ausgangsverbunden sind,
- Zustandsvariable, die eingangsverbunden, aber nicht ausgangsverbunden sind,
- Zustandsvariable, die ausgangsverbunden, aber nicht eingangsverbunden sind,
- Zustandsvariable, die weder eingangsverbunden noch ausgangsverbunden sind.

In einer nicht ganz korrekten Verwendung der Begriffe bezeichnet man die Elemente der ersten Gruppe auch als die strukturell steuerbaren und strukturell beobachtbaren Zustandsgrößen.

Die Zerlegung ist in Abb. 5.12 dargestellt. Wenn man zunächst die gestrichelt eingetragenen Kanten ignoriert, so entsprechen die gekennzeichneten Gruppen von Zustandsvariablen von oben nach unten genau den hier angegebenen Gruppen. Dann kann man sich überlegen, dass die gestrichelt eingetragenen Verbindungen zwischen Zustandsvariablen unterschiedlicher Gruppen diese Einteilung nicht verändern.

Die strukturelle Zerlegung des Systems hat wichtige Konsequenzen für die Automatisierung. Offenbar kann durch die Steuergröße  $u$  nur auf die in Abb. 5.12 oben dargestellten Zustandsvariablen  $x_1, x_2, \dots, x_6$  Einfluss genommen werden und die Ausgangsgröße hängt nur von

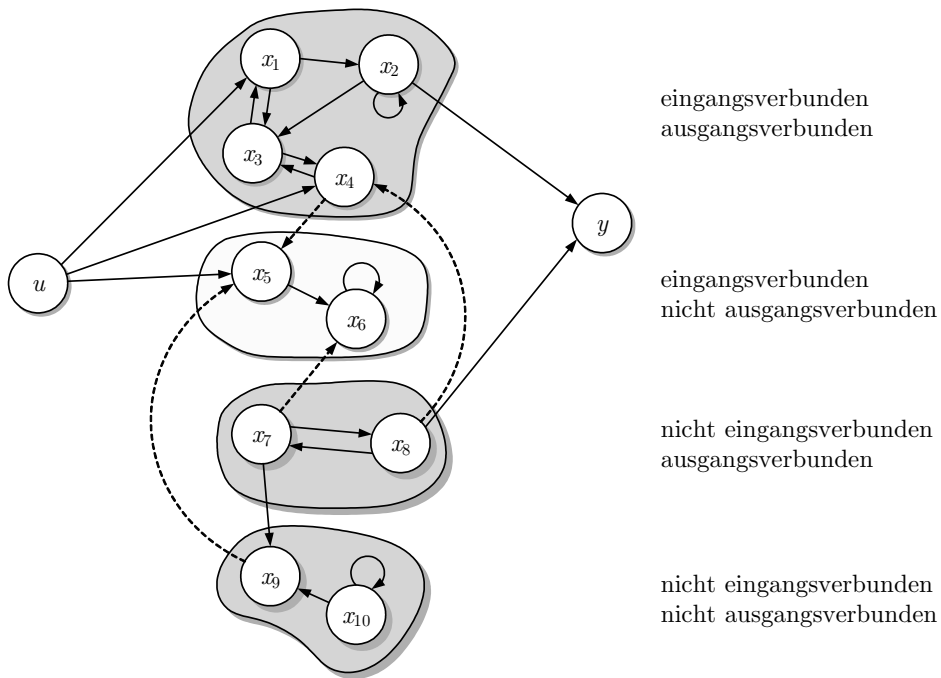


Abb. 5.12: Strukturelle Zerlegung eines Systems

$x_1$  bis  $x_4$  sowie  $x_7$  und  $x_8$  ab. Die Zustandsvariablen  $x_9$  und  $x_{10}$  können weder durch die Eingangsgröße beeinflusst werden, noch sind sie in der Messgröße erkennbar. Ein Regler, der die Ausgangsgröße  $y$  auf die Eingangsgröße  $u$  zurückführt, kann nur die Dynamik der sowohl eingangsverbundenen als auch ausgangsverbundenen Zustandsgrößen verändern, denn nur diese Zustandsgrößen liegen tatsächlich in einem geschlossenen Regelkreis. Wenn also ein instabiles System durch eine Regelung stabilisiert werden soll, so müssen die instabilen Vorgänge vollständig durch die Zustandsvariablen der obersten Gruppe dargestellt werden. Andernfalls ist das System nicht stabilisierbar. Bezogen auf die instabilen Eigenwerte  $\lambda_i$  der Systemmatrix  $A$  (Eigenwerte mit  $\text{Re}\{\lambda_i\} \geq 0$ ) heißt dieses Ergebnis:

|| Ein instabiles System kann nur dann durch einen Regler stabilisiert werden, wenn seine instabilen Eigenwerte zum steuerbaren und beobachtbaren Teilsystem gehören.

Wenn diese Bedingung nicht erfüllt ist, so kann man die Stabilisierbarkeit des Systems gegebenenfalls durch die Wahl einer anderen Stellgröße  $u$  oder einer anderen Regelgröße  $y$  erreichen.

Für lineare Systeme  $\Sigma$  führt die Zerlegung der Zustandsvariablen in vier Gruppen auf eine spezielle Struktur des Zustandsraummodells, wenn man die zu den einzelnen Gruppen gehörenden Zustandsvariablen in den Teilvektoren  $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$  bzw.  $\tilde{x}_4$  zusammengefasst. Für das in der Abbildung dargestellte System gilt also

$$\tilde{\mathbf{x}}_1(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{pmatrix}, \quad \tilde{\mathbf{x}}_2(t) = \begin{pmatrix} x_5(t) \\ x_6(t) \end{pmatrix}, \quad \tilde{\mathbf{x}}_3(t) = \begin{pmatrix} x_7(t) \\ x_8(t) \end{pmatrix}, \quad \tilde{\mathbf{x}}_4(t) = \begin{pmatrix} x_9(t) \\ x_{10}(t) \end{pmatrix}.$$

Dann kann das Modell in folgender Weise geschrieben werden, die man KALMAN-Zerlegung des Zustandsraummodells nennt:

$$\Sigma : \begin{cases} \begin{pmatrix} \dot{\tilde{\mathbf{x}}}_1(t) \\ \dot{\tilde{\mathbf{x}}}_2(t) \\ \dot{\tilde{\mathbf{x}}}_3(t) \\ \dot{\tilde{\mathbf{x}}}_4(t) \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{A}}_{11} & \mathbf{O} & \tilde{\mathbf{A}}_{13} & \mathbf{O} \\ \tilde{\mathbf{A}}_{21} & \tilde{\mathbf{A}}_{22} & \tilde{\mathbf{A}}_{23} & \tilde{\mathbf{A}}_{24} \\ \mathbf{O} & \mathbf{O} & \tilde{\mathbf{A}}_{33} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \tilde{\mathbf{A}}_{43} & \tilde{\mathbf{A}}_{44} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{x}}_1(t) \\ \tilde{\mathbf{x}}_2(t) \\ \tilde{\mathbf{x}}_3(t) \\ \tilde{\mathbf{x}}_4(t) \end{pmatrix} + \begin{pmatrix} \tilde{\mathbf{b}}_1 \\ \tilde{\mathbf{b}}_2 \\ \mathbf{O} \\ \mathbf{O} \end{pmatrix} u(t) \\ y(t) = (\mathbf{c}_1^T \quad \mathbf{0} \quad \mathbf{c}_3^T \quad \mathbf{0}) \begin{pmatrix} \tilde{\mathbf{x}}_1(t) \\ \tilde{\mathbf{x}}_2(t) \\ \tilde{\mathbf{x}}_3(t) \\ \tilde{\mathbf{x}}_4(t) \end{pmatrix}. \end{cases} \quad (5.26)$$

Da die strukturelle Steuerbarkeit bzw. Beobachtbarkeit eine notwendige Bedingung für die vollständige Steuerbarkeit bzw. Beobachtbarkeit darstellt, kann nur das erste Teilsystem steuerbar und beobachtbar sein. Schneidet man dieses Teilsystem heraus, so erhält man

$$\begin{array}{l} \text{steuerbares und} \\ \text{beobachtbares Teilsystem:} \end{array} \quad \begin{cases} \dot{\tilde{\mathbf{x}}}_1(t) = \tilde{\mathbf{A}}_{11}\tilde{\mathbf{x}}_1(t) + \tilde{\mathbf{b}}_1 u(t) \\ y(t) = \mathbf{c}_1^T \tilde{\mathbf{x}}_1(t), \end{cases} \quad (5.27)$$

wobei der Einfluss von  $\tilde{\mathbf{x}}_3(t)$  weggelassen wurde, weil  $\tilde{\mathbf{x}}_3(t)$  strukturell nicht von der Eingangsgröße  $u(t)$  abhängt. Dieses Teilsystem ist vollständig steuerbar, wenn die mit der Matrix  $\tilde{\mathbf{A}}_{11}$  und dem Vektor  $\tilde{\mathbf{b}}_1$  aufgestellte Steuerbarkeitsmatrix vollen Rang hat und es ist beobachtbar, wenn die mit der Matrix  $\tilde{\mathbf{A}}_{11}$  und dem Vektor  $\tilde{\mathbf{c}}_1^T$  aufgestellte Beobachtbarkeitsmatrix vollen Rang hat. Aus der strukturellen Analyse geht hervor, dass die anderen drei Teile des Systems nicht steuerbar bzw. beobachtbar sein können.

Zustandsvariable, die nicht steuerbar oder nicht beobachtbar sind, können keinen Einfluss auf das E/A-Verhalten des Regelkreises haben und können deshalb in dem für die Analyse des E/A-Verhaltens eingesetzten Modell weggelassen werden. Wenn man beispielsweise ermitteln will, wie der Regelkreis mit der Regelstrecke (5.26) und einem Regler einer Sollwertänderung folgt, so muss man nur das vereinfachte Modell (5.27) mit dem Reglergesetz kombinieren und aus dem erhaltenen Modell die Regelgröße berechnen. Die aus dem vollständigen Modell (5.26) weggelassenen Teile haben darauf keinen Einfluss.

**Aufgabe 5.6\*** Strukturelle Analyse des Modells für den Abbremsvorgang eines Fahrzeugs

Das im Beispiel 3.9 auf S. 107 abgeleitete Modell besteht aus den Gleichungen

$$\begin{aligned} \lambda(t) &= \frac{v(t) - r\omega(t)}{v(t)} \\ \dot{v}(t) &= -gf_2(\lambda(t)) \\ \dot{\omega}(t) &= \frac{mr}{J} (gf_2(\lambda(t)) - u(t)) \\ \dot{s}(t) &= v(t) \end{aligned}$$



sowie der in Abb. 3.15 gezeigten  $\mu(\lambda)$ -Kennlinie, die die Funktion  $f_2(\lambda)$  grafisch darstellt. Es beschreibt den Zusammenhang zwischen der Sollbremsbeschleunigung  $u(t)$  und der Winkelgeschwindigkeit  $\omega(t)$  des Rades, der Fahrzeuggeschwindigkeit  $v(t)$  und dem während des Bremsvorgangs zurückgelegten Weg  $s(t)$ . Messgröße ist die Fahrzeuggeschwindigkeit  $v(t)$ .

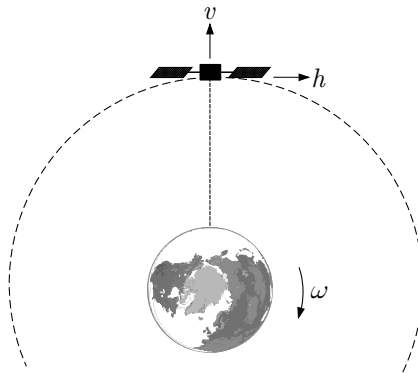
1. Zeichnen Sie den Strukturgraphen und vergleichen Sie ihn mit dem Blockschaltbild aus Abb. 3.16.
2. Ist das System strukturell steuerbar und beobachtbar? Interpretieren Sie Ihr Ergebnis.  $\square$

**Aufgabe 5.7** Strukturelle Steuerbarkeit und Beobachtbarkeit des Fensterhebers

Der im Beispiel 10.2 auf S. 299 beschriebene Fensterheber wird durch das Zustandsraummodell (10.6) beschrieben, dass hier bezüglich der Steuergröße  $u(t)$  und der Messgröße  $\omega(t)$  betrachtet wird:

$$\text{Fensterheber : } \begin{cases} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} \\ \frac{k_T}{J} & -\frac{k_B}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix} u(t) \\ y(t) = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} \end{cases}$$

1. Prüfen Sie die strukturelle Steuerbarkeit und strukturelle Beobachtbarkeit des Fensterhebers.
2. Die Position der Fensterscheibe kann anhand des Drehwinkels  $\phi(t)$  des Motors ermittelt werden, wobei zwischen der Winkelgeschwindigkeit  $\omega(t)$  und dem Winkel  $\phi(t)$  die Beziehung  $\omega(t) = \dot{\phi}(t)$  gilt. Wiederholen Sie Ihre Steuerbarkeits- und Beobachtbarkeitsuntersuchungen unter der Annahme, dass der Winkel  $\phi(t)$  gemessen wird.
3. Vergleichen und interpretieren Sie Ihre Ergebnisse.  $\square$



**Abb. 5.13:** Satellit über der Beobachtungsstation

**Aufgabe 5.8\*** Beobachtbarkeit eines Kommunikationssatelliten

Ein Satellit bewegt sich geostationär über einer Beobachtungsstation am Äquator, wie es in Abb. 5.13 gezeigt ist. Es wird angenommen, dass vom Boden aus nur die Positionsabweichung in Richtung  $h$  gemessen werden kann. Diese Abweichung hat aber i. Allg. auch Abweichungen in radialer Richtung  $v$  zur Folge. Kann der Zustand des Satelliten aus der Messgröße beobachtet werden?

Die Bewegung des Satelliten ist durch die Gleichungen

$$\ddot{v}(t) - 2\omega\dot{h}(t) - 3\omega^2 v(t) = 0 \quad (5.28)$$

$$\ddot{h}(t) + 2\omega\dot{v}(t) = u(t) \quad (5.29)$$

beschrieben, wobei  $\omega = \frac{2\pi}{360 \cdot 24} \frac{\text{rad}}{\text{sec}}$  die Winkelgeschwindigkeit der Erde und des Satelliten um den Erdmittelpunkt ist und  $u$  die Beschleunigung des Satelliten durch seine Steuerrücken in  $h$ -Richtung bezeichnet.  $\square$

## Literaturhinweise

Die Begriff der Steuerbarkeit linearer Systeme wurde von KALMAN in [50] eingeführt.

Strukturelle Betrachtungen zur Steuerbarkeit und Beobachtbarkeit entstanden aus Zerlegungsmethoden für große Gleichungssysteme, von denen man wusste, dass die Lösung im Wesentlichen von strukturellen Eigenschaften abhängt. Erste Arbeiten hierzu wurden in der Mitte der siebziger Jahre [22, 32, 60] veröffentlicht, bevor das Thema zehn Jahre später intensiver aufgegriffen und in Monografien beschrieben wurde [85, 105].

Aufgabe 5.8 ist aus [70] entnommen.

# 6

## Stabilität

*Technische Systeme müssen stabil sein, denn dann reagieren sie auf eine begrenzte Erregung mit einer begrenzten Bewegung um die Ruhelage. Dieses Kapitel behandelt Methoden zur Stabilitätsprüfung und zeigt, dass bei robusten Regelkreisen die Stabilitätseigenschaft auch bei Parameteränderungen der Regelstrecke erhalten bleibt.*

### 6.1 Stabilitätsdefinition

Unter Stabilität versteht man die Eigenschaft dynamischer Systeme, auf eine beschränkte Erregung mit einer beschränkten Bewegung zu reagieren. Im Folgenden wird die Zustandsstabilität untersucht, bei der die Erregung in einer Anfangsauslenkung  $x_0$  aus dem Arbeitspunkt  $\bar{x}$  besteht und man das System als stabil bezeichnet, wenn es sich auf Grund dieser Erregung nicht zu weit vom Arbeitspunkt entfernt. Man spricht von asymptotischer Stabilität, wenn das stabile System asymptotisch in den Arbeitspunkt  $\bar{x}$  zurückkehrt.

Die Stabilität ist eine sehr wichtige Eigenschaft technischer Systeme. Nur wenn das System stabil ist, kann es über längere Zeit und unter der Einwirkung von Störungen betrieben werden. Ist es instabil, so muss es durch eine Regelung stabilisiert werden.

Das invertierte Pendel (Beispiel 2.10 auf S. 73) ist ein anschauliches Beispiel für ein instabiles System, denn das unregelte Pendel fällt bei der kleinsten Störung um. Nur durch eine stabilisierende Regelung kann es in der oberen Ruhelage gehalten werden. Im täglichen Leben müssen wir ein ähnliches Stabilisierungsproblem beim Fahrradfahren lösen, denn auch das Fahrrad kippt ohne den als Regler wirkenden Fahrer um.

**Ruhelage.** Stabilitätsuntersuchungen beziehen sich auf Ruhelagen des Systems. Unter einer Ruhelage oder einem Gleichgewichtszustand versteht man dasselbe, was im Kap. 3 als Arbeitspunkt bezeichnet wurde, nämlich einen Zustand  $\bar{x}$ , in dem das System bei konstanter Eingangsgröße verharrt. Entsprechend Gl. (3.29) auf S. 103 muss das nichtlineare System

$$\dot{x}(t) = g(x(t), u(t)), \quad x(0) = x_0 \quad (6.1)$$

für die konstante Eingangsgröße

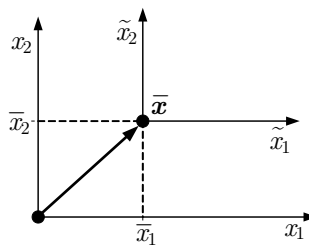
$$u(t) = \bar{u}$$

die Bedingung

$$0 = g(\bar{x}, \bar{u}) \quad (6.2)$$

erfüllen, damit  $\bar{x}$  ein Gleichgewichtszustand ist.

Die Existenz eines Gleichgewichtszustands sagt allerdings noch nichts über dessen Stabilität aus. Wenn also ein System einen Gleichgewichtszustand besitzt, so heißt das nicht, dass es sich diesem Gleichgewichtszustand nähert und schließlich in ihm verbleibt.



**Abb. 6.1:** Verschiebung des Arbeitspunkts in den Nullpunkt des Zustandsraumes

Da ein System nur dann in einem Gleichgewichtszustand verharren kann, wenn sich die Eingangsgröße nicht ändert, wird im Folgenden entweder ein autonomes System ( $u(t) = 0$ ) oder ein System mit konstanter Eingangsgröße betrachtet ( $u(t) = \bar{u}$ ). Für diese Systeme kann man die Funktion  $g(x(t), \bar{u})$  als eine neue Funktion  $g_{\bar{u}}(x(t))$  schreiben, die nur noch vom Zustand  $x(t)$  abhängt:

$$\dot{x}(t) = g_{\bar{u}}(x(t)) \quad \text{für } u(t) = \bar{u}.$$

Durch eine Koordinatenverschiebung

$$\tilde{x}(t) = x(t) - \bar{x}$$

mit  $\bar{x}$  als Lösung von Gl. (6.2) kann man dann erreichen, dass der betrachtete Gleichgewichtspunkt im Ursprung des Zustandsraumes liegt (Abb. 6.1). Dabei verändert sich die Zustandsgleichung in

$$\dot{\tilde{x}}(t) = g_{\bar{u}}(\tilde{x}(t) + \bar{x}) = \tilde{g}(\tilde{x}(t)),$$

wobei die neu eingeführte Funktion  $\tilde{g}$  die Eigenschaft

$$\mathbf{0} = \tilde{\mathbf{g}}(\mathbf{0})$$

besitzt. Die folgenden Untersuchungen können sich deshalb auf autonome Systeme

$$\Sigma_{\text{NL}} : \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (6.3)$$

mit der Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  beschränken, wobei die Funktion auf der rechten Seite der Gleichung der Einfachheit halber wieder mit  $\mathbf{g}$  und der Zustand wieder mit  $\mathbf{x}(t)$  bezeichnet werden. Wenn das System mehrere Ruhelagen besitzt, so wird die Stabilität dieser Ruhelagen nacheinander auf diese Weise überprüft.

**Stabilitätsdefinition.** Es wird nun untersucht, wie sich der Zustand  $\mathbf{x}(t)$  verhält, wenn sich das System (6.3) zum Zeitpunkt  $t = 0$  im Anfangszustand  $\mathbf{x}_0 \neq \mathbf{0}$  befindet. Der Abstand des aktuellen Zustands  $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_n(t))^T$  vom Gleichgewichtszustand  $\bar{\mathbf{x}} = \mathbf{0}$  wird durch die Vektornorm  $\|\mathbf{x}(t)\|$  beschrieben, die sich mit dem Zustand zeitlich ändert. Dabei kann jede Vektornorm eingesetzt werden, beispielsweise die euklidische Norm

$$\|\mathbf{x}(t)\| = \sqrt{\sum_{i=1}^n x_i^2(t)}.$$

Da sich die Komponenten  $x_i(t)$  des Vektors  $\mathbf{x}(t)$  zeitlich ändern, hängt auch die Norm  $\|\mathbf{x}(t)\|$  des Zustands von der Zeit  $t$  ab.

Die Vektornorm verschwindet nur, wenn alle Zustandsvariablen gleich null sind, das System also den Gleichgewichtspunkt  $\bar{\mathbf{x}} = \mathbf{0}$  erreicht hat:

$$\|\mathbf{x}(t)\| = 0 \quad \text{genau dann wenn} \quad |x_i(t)| = 0, \quad i = 1, 2, \dots, n.$$

Wenn andererseits eine oder mehrere Zustandsvariablen unendlich groß werden, so steigt die Vektornorm über alle Grenzen:

$$\|\mathbf{x}(t)\| \longrightarrow \infty \quad \text{genau dann wenn} \quad |x_i(t)| \longrightarrow \infty \quad \text{für mindestens ein } i.$$

Mit diesen Vorbemerkungen kann nun die Stabilität der Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  definiert werden.

#### **Definition 6.1 (Zustandsstabilität)**

*Der Gleichgewichtszustand  $\bar{\mathbf{x}} = \mathbf{0}$  des Systems (6.3) heißt stabil, wenn für jedes  $\varepsilon > 0$  eine Zahl  $\delta > 0$  existiert, so dass bei jedem Anfangszustand, der die Bedingung*

$$\|\mathbf{x}_0\| < \delta \quad (6.4)$$

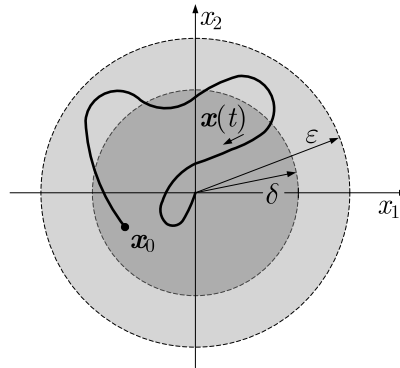
*erfüllt, die Eigenbewegung des Systems (6.3) die Bedingung*

$$\|\mathbf{x}(t)\| < \varepsilon \quad \text{für alle} \quad t > 0 \quad (6.5)$$

*erfüllt. Der Gleichgewichtszustand heißt asymptotisch stabil, wenn er stabil ist und außerdem die Bedingung*

$$\lim_{t \rightarrow \infty} \|\mathbf{x}(t)\| = 0 \quad (6.6)$$

*erfüllt ist.*



**Abb. 6.2:** Trajektorie eines asymptotisch stabilen Systems

Diese von LJAPUNOW<sup>1</sup> eingeführte Stabilitätsdefinition bedarf einiger Erläuterungen. Durch  $\varepsilon$  wird die Größe einer Umgebung um den Gleichgewichtspunkt  $\bar{x}$  vorgegeben, die in Abb. 6.2 hellgrau markiert ist. Da hier mit der euklidischen Vektornorm gerechnet wird, ist dies eine Kreiseumgebung um den Nullpunkt  $\bar{x} = \mathbf{0}$  des Zustandsraumes. Die Stabilitätsdefinition fordert nun, dass es eine Umgebung mit dem Radius  $\delta$  gibt, so dass alle Trajektorien, die in dieser Umgebung beginnen, die gegebene  $\varepsilon$ -Umgebung nicht verlassen. In der Abbildung ist eine Beispieltrajektorie eingetragen. Die Forderung, die  $\varepsilon$ -Umgebung nicht zu verlassen, muss für sämtliche Trajektorien gelten, die in allen möglichen Anfangszuständen  $x_0$  der  $\delta$ -Umgebung beginnen.

Die Stabilitätsdefinition fordert nun außerdem, dass die soeben betrachtete Bedingung für jede beliebige Vorgabe von  $\varepsilon$  erfüllt wird. Es muss also zu *jedem*  $\varepsilon$  einen Radius  $\delta$  geben, so dass die in der  $\delta$ -Umgebung beginnenden Trajektorien die  $\varepsilon$ -Umgebung nicht verlassen.

Die asymptotische Stabilität, von der im Folgenden stets die Rede sein wird, fordert darüber hinaus, dass das System für  $t \rightarrow \infty$  in die Ruhelage  $\bar{x} = \mathbf{0}$  zurückkehrt. Diese Forderung soll bei allen weiteren Untersuchungen stets erhoben werden, so dass mit dem Attribut „stabil“ stets „asymptotisch stabil“ gemeint ist.

## 6.2 Stabilitätsanalyse linearer Systeme

### 6.2.1 Stabilitätsanalyse anhand der Eigenwerte der Systemmatrix

In diesem Abschnitt werden Bedingungen aufgestellt, unter denen ein lineares autonomes System

$$\Sigma: \dot{x}(t) = Ax(t), \quad x(0) = x_0 \quad (6.7)$$

asymptotisch stabil ist. Aus der Bewegungsgleichung (4.5) erhält man für die Zustandstrajektorie die Beziehung

<sup>1</sup> ALEXANDER MICHAJLOWITSCH LJAPUNOW (1857 – 1918), russischer Mathematiker

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{x}_0$$

und daraus die Ungleichung

$$\|\mathbf{x}(t)\| \leq \|e^{\mathbf{A}t}\| \|\mathbf{x}_0\|,$$

in der das Symbol  $\|\cdot\|$  sowohl für die Vektornorm als auch für eine dieser Vektornorm zugeordnete Matrixnorm steht. Einzelheiten dieser Normungleichung können hier übergangen werden, weil sich sogleich herausstellen wird, dass die Normberechnung durch eine Betragsabschätzung ersetzt werden kann. Die in der Stabilitätsdefinition genannte Forderung (6.5) lässt sich genau dann für einen beliebigen Wert von  $\varepsilon$  durch die geeignete Wahl von  $\delta$  für alle Anfangszustände nach Gl. (6.4) erfüllen, wenn die Norm der Matrix  $e^{\mathbf{A}t}$  für alle  $t$  beschränkt ist, es also eine obere Schranke  $a_{\max}$  für die Norm dieser Matrix gibt:

$$\|e^{\mathbf{A}t}\| < a_{\max} < \infty. \quad (6.8)$$

Wählt man  $\delta$  entsprechend der Ungleichung

$$\delta \leq \frac{\varepsilon}{a_{\max}},$$

so ist die Bedingung (6.5) erfüllt. Für jedes beliebige  $\varepsilon$  gibt es also ein  $\delta$ , mit dem die in der Stabilitätsdefinition genannten Bedingungen erfüllt sind, wenn  $a_{\max}$  endlich ist. Für die asymptotische Stabilität muss zusätzlich

$$\lim_{t \rightarrow \infty} \|e^{\mathbf{A}t}\| = 0 \quad (6.9)$$

gelten.

Aus den Forderungen (6.8) und (6.9) können unter Verwendung der kanonischen Darstellung (4.19) der Bewegungsgleichung von S. 137 Bedingungen für die Eigenwerte  $\lambda_i$  der Matrix  $\mathbf{A}$  abgeleitet werden. Die Bewegungsgleichung heißt für das autonome System

$$\mathbf{x}(t) = \sum_{i=1}^n \mathbf{v}_i e^{\lambda_i t} \tilde{x}_{i0}. \quad (6.10)$$

Die Bedingung (6.9) ist genau dann erfüllt, wenn alle Modi  $e^{\lambda_i t}$  für  $t \rightarrow \infty$  verschwinden, was für

$$\operatorname{Re}\{\lambda_i\} < 0, \quad i = 1, 2, \dots, n \quad (6.11)$$

der Fall ist. Unter dieser Bedingung gilt auch die Ungleichung (6.8).

Das heißt, die Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  eines linearen autonomen Systems  $\Sigma$  ist genau dann asymptotisch stabil, wenn alle Eigenwerte der Systemmatrix  $\mathbf{A}$  einen negativen Realteil haben. Bildlich gesprochen müssen alle Eigenwerte in der linken komplexen Halbebene liegen. Man sagt deshalb auch, dass die Imaginärachse der komplexen Ebene die Stabilitätsgrenze markiert.

**Satz 6.1 (Kriterium für die Zustandsstabilität)**

*Der Gleichgewichtszustand  $\bar{x} = 0$  des linearen Systems (6.7) ist genau dann asymptotisch stabil, wenn alle Eigenwerte der Matrix  $A$  die Bedingung*

$$\operatorname{Re}\{\lambda_i\} < 0, \quad i = 1, 2, \dots, n$$

*erfüllen.*

Die angegebene Stabilitätsbedingung hat zur Folge, dass die Bedingung  $\det A \neq 0$  notwendigerweise erfüllt sein muss, damit das System asymptotisch stabil ist. Alle asymptotisch stabilen Systeme haben deshalb genau einen Gleichgewichtspunkt, denn aus

$$0 = A\bar{x}$$

folgt dann

$$\bar{x} = A^{-1}0 = 0.$$

Auch bei linearen Systemen

$$\Sigma: \dot{x}(t) = Ax(t) + b\bar{u}, \quad (6.12)$$

die durch eine konstante Eingangsgröße  $u = \bar{u}$  angeregt werden, stellt sich genau ein Gleichgewichtszustand ein, denn aus der Beziehung

$$0 = A\bar{x} + b\bar{u}$$

erhält man die eindeutige Ruhelage

$$\bar{x} = -A^{-1}b\bar{u}.$$

Die Bewegung um diese Ruhelage kann wieder durch ein Modell der Form (6.7) beschrieben werden, wenn mit

$$\tilde{x}(t) = x(t) - \bar{x}$$

der Abstand von der Ruhelage eingeführt und aus Gl. (6.12) die Zustandsgleichung

$$\Sigma: \dot{\tilde{x}}(t) = A\tilde{x}(t)$$

abgeleitet wird. In dieser Gleichung tritt dieselbe Systemmatrix  $A$  wie im autonomen Fall auf, so dass sich die Stabilitätseigenschaft vom autonomen System direkt auf das System mit beliebiger konstanter Eingangsgröße überträgt. Damit kann der Begriff der Stabilität, der in Definition 6.1 auf eine Gleichgewichtslage bezogen wurde, auch als eine Eigenschaft des Systems  $\Sigma$  aufgefasst werden. Bei linearen Systemen spricht man deshalb oft von der Stabilität des Systems an Stelle der Stabilität der (einzigen) Ruhelage.



### 6.2.2 Hurwitzkriterium

Wenn man die Stabilität eines linearen Systems überprüfen will, muss man entsprechend Satz 6.1 testen, ob alle Eigenwerte einen negativen Realteil haben. Dies kann man natürlich dadurch tun, dass man die Eigenwerte ausrechnet und sich ihre Realteile ansieht. Im Folgenden wird ein Kriterium angegeben, das ohne die Berechnung der Eigenwerte auskommt.

Ausgangspunkt ist das charakteristische Polynom der Matrix  $\mathbf{A}$

$$\det(\lambda \mathbf{I} - \mathbf{A}) = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \dots + a_1 \lambda + a_0, \quad (6.13)$$

bei dem der vor  $\lambda^n$  stehende Koeffizient  $a_n$  den Wert eins hat. Die im Folgenden angegebenen Bedingungen ermöglichen es zu entscheiden, ob sämtliche Nullstellen dieses Polynoms einen negativen Realteil haben, ohne die Nullstellen selbst auszurechnen.

Das nach HURWITZ<sup>2</sup> benannte Kriterium bezieht sich auf eine  $(n \times n)$ -Matrix  $\mathbf{H}$ , in der die Koeffizienten des charakteristischen Polynoms folgendermaßen angeordnet sind:

$$\mathbf{H} = \begin{pmatrix} a_1 & a_3 & a_5 & a_7 & \dots \\ a_0 & a_2 & a_4 & a_6 & \dots \\ 0 & a_1 & a_3 & a_5 & \dots \\ 0 & a_0 & a_2 & a_4 & \dots \\ 0 & 0 & a_1 & a_3 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

In der ersten Zeile stehen die Koeffizienten mit den ungeraden Indizes, darunter die mit den geraden Indizes. Die weiteren Zeilen erhält man, indem man die Indizes der Koeffizienten jeweils um eins verkleinert. Da es keine Koeffizienten mit negativen Indizes gibt, werden diese durch Nullen ersetzt. Gleiches gilt, wenn Koeffizienten mit Indizes eingesetzt werden sollen, die die Systemordnung  $n$  übersteigen. Entsprechend der Systemordnung muss die aufgestellte Matrix  $\mathbf{H}$   $n$  Zeilen und  $n$  Spalten haben.

Das Hurwitzkriterium bezieht sich auf die führenden Hauptabschnittsdeterminanten  $D_i$  der Matrix  $\mathbf{H}$ . Dies sind die Determinanten der in der linken oberen Ecke stehenden  $(i \times i)$ -Matrizen, die man für  $i = 1, 2, \dots, n$  entsprechend dem folgenden Schema aus der Matrix  $\mathbf{H}$  „herausschneiden“ kann:

$$\left( \begin{array}{cc|cc|cc|cc} a_1 & & a_3 & & a_5 & & a_7 & & \dots \\ & a_0 & & a_2 & & a_4 & & a_6 & \dots \\ \hline & & 0 & a_1 & & a_3 & & a_5 & \dots \\ & & & & 0 & a_0 & & a_2 & \dots \\ \hline & & & & & & 0 & 0 & \dots \\ & & & & & & & & \vdots \\ & & & & & & & & \vdots \end{array} \right). \quad (6.14)$$

<sup>2</sup> ADOLF HURWITZ (1859 – 1919), deutscher Mathematiker

Dabei erhält man für die ersten drei Determinanten

$$\begin{aligned} D_1 &= a_1 \\ D_2 &= \det \begin{pmatrix} a_1 & a_3 \\ a_0 & a_2 \end{pmatrix} \\ D_3 &= \det \begin{pmatrix} a_1 & a_3 & a_5 \\ a_0 & a_2 & a_4 \\ 0 & a_1 & a_3 \end{pmatrix}. \end{aligned}$$

Diese Determinanten werden auch als Hurwitzdeterminanten bezeichnet.

**Satz 6.2 (Hurwitzkriterium)**

*Ein lineares System mit dem charakteristischen Polynom (6.13) ist genau dann asymptotisch stabil, wenn die beiden folgenden Bedingungen erfüllt sind:*

1. *Alle Koeffizienten  $a_i$  sind positiv:*

$$a_i > 0, \quad i = 0, 1, 2, \dots, n.$$

2. *Die  $n$  führenden Hauptabschnittsdeterminanten  $D_i$  der Matrix  $\mathbf{H}$  sind positiv:*

$$D_i > 0, \quad i = 1, 2, \dots, n.$$

Die Vorzeichenbedingung  $a_i > 0$  kann man relativ schnell überprüfen, denn dieser Bedingung zufolge müssen in dem charakteristischen Polynom alle Potenzen von  $\lambda$  vorkommen und alle Koeffizienten *gleiches* Vorzeichen haben. Ist dies nicht der Fall, so ist das System instabil. Sind alle Vorzeichen negativ, so ist die erste Bedingung nach Multiplikation des Polynoms mit  $-1$  erfüllt.

Für Systeme zweiter Ordnung ( $n = 2$ ) ergibt sich aus dem Hurwitzkriterium als notwendige und hinreichende Stabilitätsbedingung, dass alle Koeffizienten des charakteristischen Polynoms

$$a_2\lambda^2 + a_1\lambda + a_0 = 0$$

positiv sein müssen. Man muss für diese Systeme die Hurwitzmatrix also gar nicht bilden, sondern kann die Stabilität sofort an den Koeffizienten ablesen. Der Grund dafür liegt in der Tatsache, dass die beiden Determinanten  $D_1$  und  $D_2$  stets positiv sind, wenn die drei Koeffizienten  $a_0$ ,  $a_1$  und  $a_2$  positiv sind.

**Beispiel 6.1** Anwendung des Hurwitzkriteriums

Gegeben ist ein lineares System mit der Systemmatrix

$$\mathbf{A} = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -2 & 1 \\ 2 & 0 & -3 \end{pmatrix}.$$

Für diese Matrix erhält man das charakteristische Polynom

$$\begin{aligned}\det(\lambda \mathbf{I} - \mathbf{A}) &= \det \begin{pmatrix} \lambda + 1 & -1 & 0 \\ 0 & \lambda + 2 & -1 \\ -2 & 0 & \lambda + 3 \end{pmatrix} \\ &= \lambda^3 + 6\lambda^2 + 11\lambda + 4,\end{aligned}$$

in dem offensichtlich alle Koeffizienten positiv sind. Für die  $(3 \times 3)$ -Matrix  $\mathbf{H}$

$$\mathbf{H} = \begin{pmatrix} 11 & 1 & 0 \\ 4 & 6 & 0 \\ 0 & 11 & 1 \end{pmatrix}$$

bekommt man folgende Hauptabschnittsdeterminanten

$$\begin{aligned}D_1 &= 11 > 0 \\ D_2 &= \det \begin{pmatrix} 11 & 1 \\ 4 & 6 \end{pmatrix} = 62 > 0 \\ D_3 &= \det \mathbf{H} = 1 \cdot D_2 > 0,\end{aligned}$$

wobei  $D_3$  nach der letzten Spalte entwickelt wurde. Alle Determinanten sind positiv und das System demzufolge asymptotisch stabil.  $\square$

Dieses Beispiel illustriert, dass die Stabilitätsprüfung ohne die Berechnung der Eigenwerte auskommt. Es zeigt auch, dass für ein System dritter Ordnung die Stabilität per Hand geprüft werden kann, obwohl die Nullstellen eines Polynoms dritter Ordnung nicht analytisch ausgerechnet werden können. Dies ist auch im Zusammenhang mit der Bestimmung geeigneter Reglerparameter der entscheidende Vorteil dieses Stabilitätskriteriums gegenüber anderen Stabilitätstests.

#### **Aufgabe 6.1\*** *Stabilitätsanalyse eines Wärmeübertragers*

Der in Aufgabe 3.7 auf S. 114 beschriebene Wärmeübertrager wird durch das linearisierte Zustandsraummodell

$$\text{Wärmeübertrager : } \begin{cases} \begin{pmatrix} \dot{T}_1(t) \\ \dot{T}_2(t) \end{pmatrix} = \begin{pmatrix} -\frac{kA+w_1c_1}{m_1c_1} & +\frac{kA}{m_1c_1} \\ +\frac{kA}{m_2c_2} & -\frac{kA+w_2c_2}{m_2c_2} \end{pmatrix} \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix} \\ y(t) = (1 \quad 0) \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix}. \end{cases}$$

beschrieben, in dem  $T_1(t)$  und  $T_2(t)$  die Abweichungen der Temperaturen der beiden Flüssigkeiten von deren Arbeitspunktwerten als Zustandsvariablen auftreten. Gegeben sind folgende Parameter:

$A = 0,0628 \text{ m}^2$	Wärmeaustauschfläche (für ein Rohr von 1 m Länge und 2 cm Durchmesser)
$w_1 = 2 \frac{\text{kg}}{\text{min}}$	Massenstrom der zu kühlenden Flüssigkeit
$w_2 = 3 \frac{\text{kg}}{\text{min}}$	Massenstrom der Kühlflüssigkeit
$m_1 = 2 \text{ kg}$	Masse der sich in der Rohrleitung befindenden Flüssigkeit
$m_2 = 1 \text{ kg}$	Masse des sich im Wärmeübertrager befindenden Kühlmittels
$c_1 = c_2 = 1 \frac{\text{cal}}{\text{kg} \cdot \text{K}}$	spezifische Wärmekapazität der Flüssigkeiten
$k = 100 \frac{\text{cal}}{\text{min} \cdot \text{m}^2 \cdot \text{K}}$	Koeffizient, der den Wärmeaustausch pro Flächeneinheit und Temperaturdifferenz beschreibt

Überprüfen Sie die Stabilität des Wärmeübertragers und geben Sie eine physikalische Erklärung für das Ergebnis. Gilt das Ergebnis nur für die gegebenen Parameterwerte oder bleibt es bei Parameteränderungen unverändert?  $\square$

## 6.3 Stabilitätsanalyse nichtlinearer Systeme

### 6.3.1 Lösungswege

**Vergleich der Stabilitätseigenschaften linearer und nichtlinearer Systeme.** Die Prüfung der Stabilität nichtlinearer Systeme ist i. Allg. viel aufwändiger als die Analyse linearer Systeme. Dies liegt zum einen daran, dass es für nichtlineare Systeme keine Eigenwerte als wichtige Systemparameter gibt, anhand derer man das Systemverhalten in einer geschlossenen Form ähnlich der Beziehung (6.10) darstellen könnte. Andererseits können nichtlineare Systeme durch „beliebige“ Funktionen  $g(x)$  in Gl. (6.3) beschrieben werden, so dass man gar nicht erwarten kann, sämtliche nichtlinearen Systeme mit einer einfachen Methode auf Stabilität überprüfen zu können.

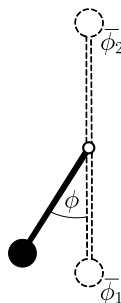


Abb. 6.3: Pendel mit zwei Ruhelagen

Auf zwei wichtige Eigenheiten der Stabilität nichtlinearer Systeme, insbesondere im Vergleich zur Stabilität linearer Systeme, sei zunächst hingewiesen. Die erste betrifft die Tatsache,

dass bei nichtlinearen Systemen mehrere Ruhelagen auftreten können, denn Gl. (6.2) kann für eine feste Amplitude  $\bar{u}$  der Stellgröße mehrere Lösungen in  $\bar{x}$  haben. Als Beispiel sei auf das in Abb. 6.3 gezeigte Pendel verwiesen, das die zwei Ruhelagen  $\bar{\phi}_1$  und  $\bar{\phi}_2$  besitzt. Deshalb ist es wichtig, daran zu erinnern, dass sich die Definition 6.1 auf die Stabilität eines Zustands  $\bar{x}$  bezieht. Während man bei linearen Systemen wegen der Eindeutigkeit des Gleichgewichtszustands auch von der Stabilität des Systems spricht, kann man dies bei nichtlinearen Systemen nicht mehr tun. Beim Pendel haben die beiden Ruhelagen unterschiedliche Stabilitätseigenschaften: Die Ruhelage  $\bar{\phi}_1$  ist offensichtlich asymptotisch stabil, während die Ruhelage  $\bar{\phi}_2$  instabil ist.

Die zweite Eigenheit betrifft das Verhalten des Systems in der Umgebung einer Ruhelage. Die Stabilitätsdefinition fordert, dass das System bei einer Auslenkung aus der Ruhelage nach einer gewissen Zeit in diese Ruhelage zurückkehrt. Bei asymptotisch stabilen linearen Systemen ist dies bei beliebig großen Auslenkungen der Fall, denn der Wert von  $x_0$  geht nicht in das Stabilitätskriterium ein. Bei nichtlinearen Systemen kommt es jedoch häufig vor, dass das System bei hinreichend kleinen Auslenkungen zu einer stabilen Ruhelage zurückkehrt, sich bei großen Auslenkungen aber entweder beliebig weit von dieser Ruhelage entfernt oder einer anderen Ruhelage nähert. Man muss deshalb bei nichtlinearen Systemen stets angeben, auf welche Umgebung der Ruhelage sich eine bestimmte Stabilitätseigenschaft bezieht. Häufig werden dabei Umgebungen untersucht, die durch eine Normungleichung der Form

$$\|x_0 - \bar{x}\| \leq r$$

beschrieben werden, bei der  $r$  die Größe der Umgebung um die Ruhelage  $\bar{x}$  festlegt.

Die Existenz derartiger Umgebungen wird offensichtlich, wenn man ein System mit einer stabilen und einer instabilen Ruhelage untersucht. Dann muss es eine Trennlinie im Zustandsraum um die stabile Ruhelage geben. Solange der Zustand diese Trennlinie nicht überschreitet, kehrt das System zur stabilen Ruhelage zurück (vgl. Aufgabe 6.4). Das Verhalten des Systems ist dann entscheidend vom Anfangszustand  $x_0$  abhängig, nämlich von der Frage, ob sich dieser Anfangszustand im Einzugsgebiet der stabilen oder der instabilen Ruhelage befindet.

**Wege für die Stabilitätsprüfung nichtlinearer Systeme.** Für die Stabilitätsprüfung nichtlinearer Systeme werden hier zwei Methoden behandelt. Im Abschn. 6.3.2 wird davon ausgegangen, dass sich das System in der Nähe eines Arbeitspunkts befindet, in der es mit guter Näherung durch ein lineares Modell beschrieben werden kann. Die Stabilität des Arbeitspunkts wird mit Hilfe der für lineare Systeme behandelten Methoden analysiert. Wenn das System mehrere Ruhelagen hat, erhält man unterschiedliche linearisierte Modelle, die einzeln auf Stabilität geprüft werden müssen.

Bei der zweiten Methode wird die Entfernung des Zustands von einer Ruhelage durch eine Energiefunktion (Ljapunowfunktion) beschrieben, die in der Ruhelage verschwindet. Wenn sich herausstellt, dass der Wert der Energiefunktion entlang der Trajektorie des Systems abnimmt, kann man sicher sein, dass das System nach hinreichend langer Zeit die Ruhelage erreicht und dort verbleibt. Diese auf LJAPUNOW zurückgehende Vorgehensweise wird als Direkte Methode von Ljapunow bezeichnet. Sie wird im Abschn. 6.3.3 erläutert.

### 6.3.2 Stabilitätsprüfung mit dem linearisierten Modell

Der erste Weg für die Stabilitätsprüfung des nichtlinearen Systems

$$\Sigma_{\text{NL}} : \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0$$

verwendet ein linearisiertes Modell, das das System  $\Sigma_{\text{NL}}$  in der Umgebung der auf Stabilität zu untersuchenden Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  näherungsweise beschreibt. Dieses Modell wird mit der im Abschn. 3.5 beschriebenen Methode gebildet und führt auf die Zustandsgleichung

$$\Sigma : \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{x}(0) = \mathbf{x}_0$$

mit

$$\mathbf{A} = \left. \frac{d\mathbf{g}}{d\mathbf{x}} \right|_{\mathbf{x}=\bar{\mathbf{x}}}.$$

Nun wird die Stabilität der Ruhelage  $\bar{\mathbf{x}} = \mathbf{0}$  mit den im Abschn. 6.2 beschriebenen Methoden geprüft. Die dabei erhaltene Aussage gilt für das Gebiet des Zustandsraumes, für das das linearisierte Modell  $\Sigma$  das nichtlineare System  $\Sigma_{\text{NL}}$  mit hinreichender Genauigkeit beschreibt.

#### Beispiel 6.2 Stabilitätsanalyse eines Pendels mit Hilfe des linearisierten Modells

Die Stabilität der Ruhelage des in Abb. 6.4 gezeigten Pendels kann mit dem linearisierten Modell bewiesen werden.

**Modellbildung.** Um das nichtlineare Modell des Pendels aufzustellen, berechnet man die Kraft  $F(t)$ , die das Pendel bei einer Auslenkung  $\phi(t)$  beschleunigt, entsprechend

$$F(t) = mg \sin \phi(t),$$

so dass

$$J \frac{d^2 \phi(t)}{dt^2} = -lF(t) - lF_B(t)$$

mit dem Trägheitsmoment

$$J = ml^2$$

für das Pendel gilt.  $F_B(t)$  ist eine Bremskraft, die durch den Luftwiderstand und die Reibung im Pendel entsteht und deren Betrag proportional zur Drehgeschwindigkeit ist:

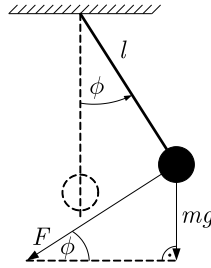
$$F_B(t) = k \frac{d\phi(t)}{dt}.$$

Damit erhält man die Beziehung

$$\frac{d^2 \phi(t)}{dt^2} = -\frac{g}{l} \sin \phi(t) - \frac{k}{ml} \frac{d\phi(t)}{dt}.$$

Das Pendel hat die Zustandsvariablen  $x_1(t) = \phi(t)$  und  $x_2(t) = \dot{\phi}(t)$ , mit denen man zu dem folgenden nichtlinearen Zustandsraummodell kommt:

$$\text{Pendel : } \begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{g}{l} \sin x_1(t) - \frac{k}{ml} x_2(t). \end{cases} \quad (6.15)$$



**Abb. 6.4:** Pendel mit dem für die Modellbildung wichtigen Kräftedreieck

**Stabilitätsanalyse.** Für die Ruhelage gilt  $\bar{x}_1 = \bar{\phi} = 0$  und  $\bar{x}_2 = 0$ . Die Linearisierung um diesen Punkt ergibt das Modell

$$\begin{aligned}\dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -\frac{g}{l}x_1(t) - \frac{k}{ml}x_2(t),\end{aligned}$$

das als Zustandsgleichung

$$\text{Pendel: } \dot{\mathbf{x}}(t) = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{k}{ml} \end{pmatrix} \mathbf{x}(t) \quad (6.16)$$

geschrieben werden kann. Die charakteristische Gleichung

$$\begin{aligned}\det(\lambda \mathbf{I} - \mathbf{A}) &= \det \begin{pmatrix} \lambda & -1 \\ \frac{g}{l} & \lambda + \frac{k}{ml} \end{pmatrix} \\ &= \lambda^2 + \frac{k}{ml}\lambda + \frac{g}{l} = 0\end{aligned}$$

erfüllt offensichtlich das Hurwitzkriterium, weil alle Parameter des Pendels positiv sind. Damit ist die Stabilität der Ruhelage des Pendels nachgewiesen. Das Einzugsgebiet für diese Ruhelage ist mindestens so groß wie der Gültigkeitsbereich des linearisierten Modells. Dieser Gültigkeitsbereich wird durch die Beziehung  $\sin \phi \approx \phi$  abgeschätzt (also etwa  $|\phi| \leq 30^\circ$ ).  $\square$

#### **Aufgabe 6.2** Stabilitätsanalyse des invertierten Pendels

Betrachten Sie in Erweiterung zum Beispiel 6.2 ein Pendel, das auch eine obere Ruhelage hat (vgl. Abb. 6.3). Diese Ruhelage ist offenbar instabil. Weisen Sie die Instabilität mit dem linearisierten Modell nach, das Sie aus dem nichtlinearen Zustandsraummodell (6.15) für die obere Ruhelage erhalten. Was ändert sich gegenüber dem Beispiel 6.2?  $\square$

**Aufgabe 6.3\*** *Stabilität des Unterwasserfahrzeugs*

Gegeben ist das reduzierte Modell des in Beispiel 3.8 beschriebenen Unterwasserfahrzeugs

$$\text{Drifter : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} x_2(t) \\ g - \frac{V_0 \rho_0 g}{m} (1 + (\kappa_W - \kappa_D) \rho_0 g x_1(t)) - \frac{k_W}{m} x_2(t) \end{pmatrix} \\ y(t) = x_1(t), \end{cases}$$

das sich auf den Zustand  $\mathbf{x}(t) = (z(t) \quad \dot{z}(t))^T$  bezieht und die folgenden Parameter besitzt:

$g$	Erdbeschleunigung
$V_0$	Volumen des Drifters in der Position $z = 0$
$\rho_0$	Dichte des Wassers bei $z = 0$
$m$	Masse des Drifters
$\kappa_W, \kappa_D$	Kompressibilität des Wassers und des Drifters
$k_W$	Fluidwiderstand des Drifters.

1. Linearisieren Sie das angegebene Zustandsraummodell um eine Gleichgewichtslage.
2. Unter welchen Bedingungen ist die Gleichgewichtslage des Unterwasserfahrzeugs stabil? Interpretieren Sie diese Bedingungen. □

**Aufgabe 6.4\*** *Stabilitätsanalyse des Schlupfverhaltens beim Abbremsen eines Fahrzeugs*

Das in der Aufgabe 3.6 abgeleitete linearisierte Modell

$$\text{Fahrzeug : } \begin{cases} \dot{\omega}(t) = -\frac{mr^2 g k_{\mu\lambda}}{J\bar{v}} \omega(t) - \frac{mr}{J} u(t) \\ y(t) = -\frac{r k_{\mu\lambda}}{\bar{v}} \omega(t) \end{cases} \quad (6.17)$$

beschreibt den Zusammenhang zwischen der Sollbeschleunigung  $u(t)$  und der Istbeschleunigung  $y(t)$  beim Abbremsen eines Fahrzeugs, wobei  $k_{\mu\lambda}$  die Steigung der  $\mu(\lambda)$ -Kennlinie darstellt:

$$k_{\mu\lambda} = \left( \frac{d\mu}{d\lambda} \right)_{\lambda=\bar{\lambda}}. \quad (6.18)$$

Der Arbeitspunkt, für dessen Umgebung das linearisierte Modell gilt, ist durch

$$\bar{\mu} = \frac{\bar{u}}{g} \quad \text{und} \quad \bar{\omega} = \frac{(1 - \bar{\lambda})\bar{v}}{r}$$

gegeben, wobei  $\bar{\mu}$  und  $\bar{\lambda}$  die in Abb. 3.15 auf S. 108 grafisch dargestellte  $\mu(\lambda)$ -Beziehung erfüllen. In Abhängigkeit von der Eingangsgröße  $\bar{u}$  ergeben sich ein oder zwei Arbeitspunkte (vgl. Abb. A.18 auf S. 588).

1. Sind die Arbeitspunkte stabil? Welche Stabilitätseigenschaft ergibt sich, wenn  $\bar{\mu} = \mu_H$  ist, also die beiden in der Abbildung eingetragenen Arbeitspunkte  $A_1$  und  $A_2$  zusammenfallen?
2. Beschreiben Sie, wie sich das System auf Grund dieser Stabilitätseigenschaften verhält, wenn es durch eine Eingangsgröße  $u(t) \neq 0$ , ( $0 \leq t < \bar{t}$ ) aus dem Arbeitspunkt ausgelenkt wird und sich anschließend wieder autonom bewegt ( $u(t) = 0$  für  $t \geq \bar{t}$ ). Nutzen Sie dazu das in Abb. A.19 auf S. 589 dargestellte Blockschaltbild des Modells (6.17). □



### 6.3.3 Direkte Methode von Ljapunow

In diesem Abschnitt wird ein Verfahren vorgestellt, mit dem man die Stabilität einer Ruhelage mit Hilfe des nichtlinearen Modells überprüfen kann. Die Grundidee dieses als Direkte Methode von Ljapunow bezeichneten Verfahrens kann man sich an dem in Abb. 6.5 gezeigten Pendel überlegen. Die Ruhelage  $\bar{\phi} = 0$  des Pendels ist bekanntermaßen asymptotisch stabil, denn aus einer Anfangsauslenkung  $\phi(0)$  schwingt das Pendel mit sich ständig vermindender Amplitude, bis es schließlich in der Ruhelage  $\bar{\phi} = 0$  verharret.

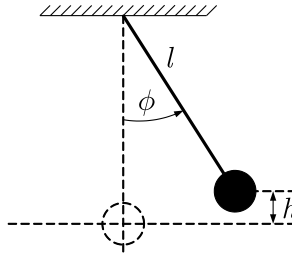


Abb. 6.5: Pendel

Die Ursache für die Stabilität des Pendels ist die Dämpfung der Bewegung durch den Luftwiderstand, durch den sich die im Pendel gespeicherte Energie kontinuierlich verkleinert. Die Energie des Pendels setzt sich aus der kinetischen Energie  $v_{\text{kin}}(t)$  und der potentiellen Energie  $v_{\text{pot}}(t)$  zusammen, wobei die kinetische Energie von der Bewegungsgeschwindigkeit  $\dot{\phi}(t)$  und die potentielle Energie von der Höhe  $h(t)$  des Pendelschwerpunkts über der Ruhelage abhängt (Abb. 6.5). Die Gesamtenergie

$$v(t) = v_{\text{kin}}(t) + v_{\text{pot}}(t)$$

nimmt stetig ab

$$v(t) \xrightarrow{t \rightarrow \infty} 0. \quad (6.19)$$

Da  $v(t) = 0$  bedeutet, dass die Geschwindigkeit  $\dot{\phi}(t)$  und die Höhe  $h(t)$  gemeinsam verschwinden, zeigt die Beziehung (6.19), dass das Pendel aus einer beliebigen Anfangsauslenkung in die Ruhelage  $\bar{\phi} = 0$  zurückkehrt und diese Ruhelage folglich asymptotisch stabil ist.

Diese Idee kann man für beliebige nichtlineare Systeme (6.3)

$$\Sigma_{\text{NL}} : \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (6.20)$$

anwenden, indem man für diese Systeme eine Funktion  $v(t)$  mit den Eigenschaften einer Energiefunktion aufstellt und zeigt, dass diese Energiefunktion entlang der Trajektorie des Systems monoton abnimmt. Ob  $v(t)$  dabei tatsächlich die im System gespeicherte Energie beschreibt oder lediglich eine Funktion mit denselben Eigenschaften darstellt, ist für das Ergebnis der Stabilitätsprüfung gleichgültig.

Um die von der Funktion  $v(t)$  geforderten Eigenschaften aufzuschreiben, muss man sich zunächst überlegen, wie diese Energie zu jedem Zeitpunkt  $t$  vom Zustand  $\mathbf{x}(t)$  abhängt. Die

Zeitabhängigkeit der Energie ist durch die Zeitabhängigkeit des Zustands begründet:  $v(\mathbf{x}(t))$ . Als erste Eigenschaft fordert man, dass für die Funktion  $v(\mathbf{x})$  die beiden Bedingungen

$$\left. \begin{array}{l} v(\mathbf{x}) > 0 \quad \text{für } \mathbf{x} \neq \mathbf{0} \\ v(\mathbf{0}) = 0 \end{array} \right\} \quad (6.21)$$

gelten. Diese Bedingungen besagen, dass die Energie auf die Ruhelage bezogen ist, also in der Ruhelage verschwindet und ansonsten positive Werte hat.

Für die Stabilitätsanalyse muss man nun nachweisen, dass sich der Wert von  $v(\mathbf{x}(t))$  entlang der Trajektorie  $\mathbf{x}(t)$  ständig verkleinert, so dass

$$\dot{v}(\mathbf{x}(t)) < 0, \quad t \geq 0 \quad (6.22)$$

gilt. Eine Funktion  $v(\mathbf{x})$ , die die beiden Eigenschaften (6.21) und (6.22) erfüllt, wird *Ljapunowfunktion* genannt. Die Existenz einer Ljapunowfunktion ist hinreichend für die asymptotische Stabilität des Systems.

Da die Funktion  $v(\mathbf{x})$  nur über den Zustand  $\mathbf{x}(t)$  von der Zeit  $t$  abhängt, gilt für ihre zeitliche Ableitung die Beziehung

$$\frac{dv(t)}{dt} = \left( \frac{dv}{d\mathbf{x}} \right)^T \frac{d\mathbf{x}(t)}{dt}, \quad (6.23)$$

wobei die Ableitung von  $v$  nach dem Zustandsvektor  $\mathbf{x}$  entsprechend

$$\frac{dv}{d\mathbf{x}} = \begin{pmatrix} \frac{dv}{dx_1} \\ \frac{dv}{dx_2} \\ \vdots \\ \frac{dv}{dx_n} \end{pmatrix} = \nabla v$$

gebildet und mit  $\nabla v$  bezeichnet<sup>3</sup> wird.  $\nabla v$  ist der Gradient von  $v$ . Für den zweiten Faktor in in Gl. (6.23) kann die in der Differentialgleichung (6.20) auftretende Funktion  $\mathbf{g}$  eingesetzt werden, was auf die Beziehung

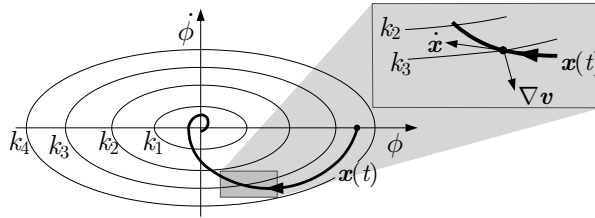
$$\frac{dv}{dt} = \left( \frac{dv}{d\mathbf{x}} \right)^T \mathbf{g}(\mathbf{x}(t)) \quad (6.24)$$

führt. Die Forderung (6.22) ist erfüllt, wenn das auf der rechten Seite von Gl. (6.24) stehende Produkt für alle  $\mathbf{x} \neq \mathbf{0}$  negativ ist. Um dies zu prüfen, muss man die Trajektorie  $\mathbf{x}(t)$  nicht ausrechnen, sondern nur zeigen, dass das Produkt  $\nabla v^T \mathbf{g}(\mathbf{x})$  für alle nicht verschwindenden  $\mathbf{x} \in \mathbb{R}^n$  negativ ist.

Die beiden genannten Eigenschaften sind in Abb. 6.6 veranschaulicht. Die Ellipsen verbinden die Zustände  $\mathbf{x}(t)$ , für die die Funktion  $v(\mathbf{x}(t))$  denselben Wert hat:

$$v(\mathbf{x}(t)) = k_i.$$

<sup>3</sup> gesprochen: „Nabla vau“



**Abb. 6.6:** Veranschaulichung der Stabilitätsprüfung unter Verwendung einer Ljapunowfunktion

Dabei gilt  $k_1 < k_2 < k_3 < k_4$ . An diesen Linien kann man sehen, dass der Wert von  $v$  entlang der eingezeichneten Trajektorie ständig abnimmt. In der Detaildarstellung sieht man, dass die Projektion der Bewegungsrichtung  $\dot{x}(t)$  auf den Gradienten  $\nabla v$  in negative Gradientenrichtung zeigt, so dass wie gefordert die Ungleichung

$$\dot{v}(x(t)) = \nabla v^T \dot{x}(t) < 0, \quad t \geq 0$$

gilt. Die Energiefunktion nimmt ab und das System bewegt sich auf den Ruhezustand zu.

**Satz 6.3 (Stabilitätsanalyse unter Verwendung einer Ljapunowfunktion)**

Die Ruhelage  $\bar{x} = \mathbf{0}$  des nichtlinearen Systems  $\dot{x}(t) = g(x(t))$  ist asymptotisch stabil, wenn es eine stetige Funktion  $v(x)$  gibt, die die Eigenschaften (6.21) und (6.22) besitzt.

Dieses Kriterium beschreibt eine hinreichende Stabilitätsbedingung, denn wenn eine Ljapunowfunktion gefunden ist, ist die asymptotische Stabilität des Systems nachgewiesen. Es fordert, dass die beiden Eigenschaften für beliebige Anfangsauslenkungen  $x_0$  gelten. Das System kehrt also von jedem Punkt des Zustandsraums in die Ruhelage zurück. Es ist deshalb *global* asymptotisch stabil. Wenn man die geforderten Eigenschaften nur für eine Teilmenge  $\mathcal{X} \subset \mathbb{R}^n$  beweisen kann, beschreibt  $\mathcal{X}$  den Einzugsbereich der Ruhelage  $\bar{x} = \mathbf{0}$ .

In einer etwas abgeschwächten Form kann man anstelle von Gl. (6.22) die Bedingung

$$\dot{v}(x(t)) \leq 0 \tag{6.25}$$

stellen, womit gesichert ist, dass die Funktion  $v(x)$  entlang der Trajektorie  $x(t)$  nicht ansteigt. Im Allgemeinen kann man daraus allerdings nur die Stabilität und nicht die asymptotische Stabilität folgern.

Wenn man allerdings zeigen kann, dass  $\dot{v}(x(t))$  für  $x(t) \neq \mathbf{0}$  nur für einzelne Zeitpunkte und nicht für Zeitintervalle verschwinden kann, so wird  $v(x(t))$  auch unter der schwächeren Bedingung (6.25) asymptotisch verschwinden und es kann auch hier die asymptotische Stabilität gefolgert werden. Diese Erweiterung wird in der Literatur als *Invarianzprinzip von LASALLE* bezeichnet.

Für den praktischen Einsatz stellen sich zwei Fragen:

- Wie muss für ein gegebenes System die „Energiefunktion“  $v$  definiert werden?

Diese Frage ist vor allem für Systeme schwierig zu beantworten, die sich gar nicht auf physikalische Vorgänge beziehen, in der eine Energie eine Rolle spielt. Beispielsweise ist der Zustand von Rechnernetzen durch die Anzahl der Datenpakete in den Warteschlangen beschrieben, die nichts mit einer Energie zu tun hat, aber möglicherweise für die Definition einer Ljapunowfunktion herangezogen werden kann.

- Wie kann man nachweisen, dass die Eigenschaft (6.22) gilt, ohne  $\mathbf{x}(t)$  für eine gegebene Anfangsbedingung  $\mathbf{x}_0$  auszurechnen?

Im Folgenden werden Hinweise zur Beantwortung dieser Fragen gegeben. Zunächst wird das Pendelbeispiel mit der Direkten Methode von Ljapunow analysiert.

### Beispiel 6.3 Stabilitätsanalyse eines Pendels mit der Direkten Methode von Ljapunow

Für das in Abb. 6.7 gezeigte und durch die Gleichungen (6.15)

$$\text{Pendel : } \begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{g}{l} \sin x_1(t) - \frac{k}{ml} x_2(t) \end{cases} \quad (6.26)$$

beschriebene Pendel kann die Ljapunowfunktion direkt aus einer Betrachtung der Energie abgeleitet werden. Die im Pendel beim Zustand  $\mathbf{x}(t)$  gespeicherte Energie

$$v(\mathbf{x}(t)) = v_{\text{kin}}(\mathbf{x}(t)) + v_{\text{pot}}(\mathbf{x}(t))$$

setzt sich aus der kinetischen Energie

$$v_{\text{kin}}(\mathbf{x}(t)) = \frac{m}{2} (lx_2(t))^2$$

und der potentiellen Energie

$$v_{\text{pot}}(\mathbf{x}(t)) = mgh(t)$$

zusammen, wobei sich die Höhe  $h(t)$  aus

$$h(t) = l(1 - \cos x_1(t))$$

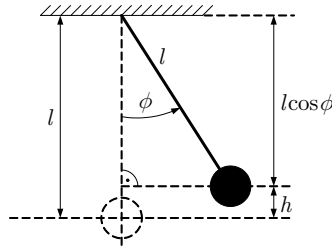
ergibt. Damit erhält man für die gespeicherte Energie die Beziehung

$$v(\mathbf{x}(t)) = \frac{m}{2} (lx_2(t))^2 + mgl(1 - \cos x_1(t)).$$

Für den hier betrachteten Bereich  $-90^\circ \leq \phi \leq 90^\circ$  ist die Bedingung (6.21) offensichtlich erfüllt.

Zum Nachweis, dass die Funktion  $v$  auch die Eigenschaft (6.22) erfüllt, werden folgende Rechenschritte durchgeführt:

$$\begin{aligned} \dot{v}(\mathbf{x}(t)) &= \frac{\partial v}{\partial x_1} \dot{x}_1(t) + \frac{\partial v}{\partial x_2} \dot{x}_2(t) \\ &= mgl \sin x_1(t) x_2(t) + ml^2 x_2(t) \left( -\frac{g}{l} \sin x_1(t) - \frac{k}{ml} x_2(t) \right) \\ &= mgl x_2(t) \sin x_1(t) - mgl x_2(t) \sin x_1(t) - kl x_2^2(t) \\ &= -kl x_2^2(t) \leq 0. \end{aligned}$$

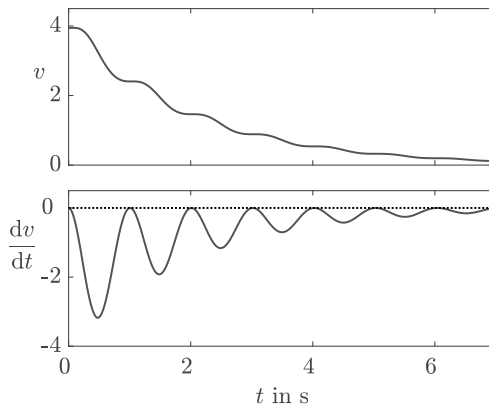


**Abb. 6.7:** Pendel mit den für die Bestimmung der potentiellen Energie maßgebenden Größen

Da  $k$  und  $l$  positive Konstanten sind, ist die Bedingung (6.25) erfüllt und die Stabilität der Ruhelage  $\bar{\phi} = 0$  des Pendels nachgewiesen.

Für den Nachweis der asymptotischen Stabilität muss man das Invarianzprinzip von LASALLE anwenden. Für  $x_2(t) \neq 0$  nimmt  $v$  streng monoton ab:  $\dot{v}(t) < 0$ . Da außerdem entsprechend Gl. (6.26) die Zustandsvariable  $x_2(t)$  nicht dauerhaft verschwinden kann, solange das Pendel nicht in der Ruhelage  $\bar{x} = \mathbf{0}$  ist, nimmt die Funktion  $v(t)$  monoton ab, was die asymptotische Stabilität des Pendels nachweist.

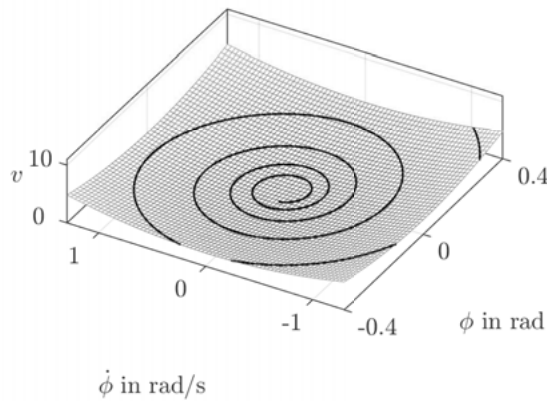
**Diskussion.** Der Nachweis, dass die im Satz 6.3 genannten Bedingungen für alle Anfangsbedingungen  $x_0$  erfüllt sind, wurde hier analytisch geführt. Damit wurde auch gezeigt, dass diese Bedingungen sogar für beliebige Parameter des Pendels erfüllt sind. Das Ergebnis ist nicht überraschend, denn beim Pendel ist ja bekannt, dass die Ruhelage asymptotisch stabil ist. Wichtig ist vor allem die Vorgehensweise, die man auf andere Systeme übertragen kann.



**Abb. 6.8:** Verhalten der Ljapunowfunktion für das Pendel

Abbildung 6.8 zeigt den Verlauf der Ljapunowfunktion  $v(t)$  (oben) und deren Ableitung  $\dot{v}(t)$  (unten). Erwartungsgemäß nimmt die Energie des Pendels monoton ab, allerdings nicht streng monoton.

Wenn man den Wert der Ljapunowfunktion als dritte Koordinate einführt, beschreibt  $v(\mathbf{x})$  eine Fläche über dem Zustandsraum (Abb. 6.9). Die Wölbung der Fläche erklärt sich aus der Tatsache, dass die Energie des Pendels überproportional mit der Entfernung von der Ruhelage steigt. Wenn man für



**Abb. 6.9:** Darstellung der Zustandstrajektorie des Pendels auf der durch die Ljapunowfunktion aufgespannten Oberfläche

das Pendel über der Trajektorie  $\mathbf{x}(t)$  den Wert  $v(t)$  der Ljapunowfunktion aufträgt, erhält man die in der Abbildung gezeigte Kurve. Satz 6.3 fordert, dass der Wert der Ljapunowfunktion entlang der Trajektorie abnimmt. Die Trajektorie führt auf der Fläche also immer „bergab“. Dies veranschaulicht für das Pendel die Idee der Direkten Methode der Stabilitätsanalyse nichtlinearer Systeme.

Für den Bereich für  $\phi$  und  $\dot{\phi}$ , in dem sich das Pendel bewegen kann, ist die asymptotische Stabilität global nachgewiesen.  $\square$

**Diskussion der Vorgehensweise.** Die in diesem Abschnitt beschriebene Vorgehensweise für die Stabilitätsprüfung wird als Direkte Methode von Ljapunow bezeichnet, weil sie ohne die Lösung der Differentialgleichung auskommt. Als Indirekte Methode hat LJAPUNOW das im Abschn. 6.3.2 beschriebene Vorgehen bezeichnet, das Systemverhalten in der Nähe der Ruhelagen durch ein linearisiertes Modell zu beschreiben und die Stabilität mit Methoden für lineare Systeme zu überprüfen.

Im Prinzip könnte man die Stabilität der Ruhelagen eines nichtlinearen Systems dadurch analysieren, dass man die Eigenschaften der Lösung der Differentialgleichung überprüft. Dieses Vorgehen ist jedoch nur in den Spezialfällen möglich, in denen für die Lösung der nichtlinearen Differentialgleichung (6.20) ein geschlossener Ausdruck angegeben werden kann, denn die Lösung hängt vom Anfangszustand  $\mathbf{x}_0$  ab und die Stabilitätsprüfung muss sämtliche Anfangszustände berücksichtigen. Demgegenüber erhält man mit der Direkten Methode bei vielen nichtlinearen Differentialgleichungen einen analytischen Ausdruck für das Produkt  $\nabla v^T \mathbf{g}(\mathbf{x})$  und kann mit diesem die Forderung (6.22) für alle  $\mathbf{x}_0 \in \mathbb{R}^n$ ,  $\mathbf{x} \neq \mathbf{0}$  überprüfen.

Die beschriebene Vorgehensweise nutzt eine (verallgemeinerte) Energiefunktion und weist die Stabilität einer Ruhelage dadurch nach, dass gezeigt wird, dass das System ständig einen Teil seiner Energie verliert und deshalb nach hinreichend langer Zeit in die Ruhelage zurückkehren muss. Systeme, die ständig Energie verlieren, werden *dissipativ* genannt. Beispiele für derartige Systeme sind neben dem Pendel auch Widerstandsnetzwerke oder Feder-Masse-Systeme. Die hier beschriebene Stabilitätsprüfung gelingt bei dissipativen Systemen dadurch, dass man  $v$  als die tatsächlich im System gespeicherte Energie ansetzt.

### 6.3.4 Anwendung der Direkten Methode auf lineare Systeme

Die Ljapunowsche Methode der Stabilitätsprüfung kann man natürlich auch auf lineare Systeme anwenden. Dies soll hier kurz erläutert werden, weil man dabei auch einen Weg findet, um Ljapunowfunktionen aufzustellen.

Für das lineare System

$$\Sigma : \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad \mathbf{x}(0) = \mathbf{x}_0$$

macht man für die Ljapunowfunktion den Ansatz

$$v(\mathbf{x}(t)) = \mathbf{x}^T(t) \mathbf{P} \mathbf{x}(t),$$

wobei  $\mathbf{P}$  eine symmetrische, positiv definite Matrix ist. Auf Grund dieser Eigenschaft von  $\mathbf{P}$  ist die erste geforderte Eigenschaft (6.21) von  $v(\mathbf{x})$  erfüllt.

Um die zweite Eigenschaft zu überprüfen, wird  $\dot{v}(\mathbf{x})$  folgendermaßen berechnet:

$$\begin{aligned} \dot{v}(\mathbf{x}(t)) &= \dot{\mathbf{x}}^T(t) \mathbf{P} \mathbf{x}(t) + \mathbf{x}^T(t) \mathbf{P} \dot{\mathbf{x}}(t) \\ &= \mathbf{x}^T(t) \mathbf{A}^T \mathbf{P} \mathbf{x}(t) + \mathbf{x}^T(t) \mathbf{P} \mathbf{A} \mathbf{x}(t) \\ &= \mathbf{x}^T(t) \left( \mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} \right) \mathbf{x}(t) \\ &= -\mathbf{x}^T(t) \mathbf{Q} \mathbf{x}(t). \end{aligned}$$

Dabei wurde im letzten Schritt die Matrix  $\mathbf{Q}$  eingeführt, mit der die Beziehung

Ljapunowgleichung:  $\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} = -\mathbf{Q}$

(6.27)

gilt. Diese Gleichung wird als Ljapunowgleichung bezeichnet.

Die Matrix  $\mathbf{Q}$  ist symmetrisch. Wenn sie auch positiv definit ist, so gilt

$$\dot{v}(\mathbf{x}(t)) = -\mathbf{x}^T(t) \mathbf{Q} \mathbf{x}(t) < 0 \quad \text{für } \mathbf{x} \neq \mathbf{0}$$

und  $v(\mathbf{x}(t))$  erfüllt auch die zweite geforderte Eigenschaft (6.25). Dann ist das lineare System asymptotisch stabil. Man muss also nach einer positiv definiten Matrix  $\mathbf{P}$  suchen, für die die aus diesem  $\mathbf{P}$  und der Systemmatrix  $\mathbf{A}$  berechnete Matrix  $\mathbf{Q}$  positiv definit ist. Wenn dies gelungen ist, hat man die Stabilität des Systems nachgewiesen.

Um diese Suche zu vereinfachen, geht man den umgekehrten Weg. Man gibt eine positiv definite Matrix  $\mathbf{Q}$  vor und berechnet die Matrix  $\mathbf{P}$  aus der Ljapunowgleichung. Das ist nicht ganz einfach, weil man keinen expliziten Ausdruck für  $\mathbf{P}$  hat, aber es gibt sehr gute numerische Algorithmen, mit denen man dies für ein gegebenes Paar  $\mathbf{A}$ ,  $\mathbf{Q}$  tun kann. Wenn die Lösung  $\mathbf{P}$  der Ljapunowgleichung positiv definit ist, sind wieder beide Bedingungen des Satzes 6.3 erfüllt und die Stabilität des Systems ist nachgewiesen.

Bei dieser Vorgehensweise stellt sich die Frage, wie man die Matrix  $\mathbf{Q}$  wählen muss, damit  $\mathbf{P}$  positiv definit ist. Dies ist glücklicherweise kein Problem, weil man nachweisen kann, dass die aus der Ljapunowgleichung für eine beliebige positiv definite Matrix  $\mathbf{Q}$  erhaltene Matrix  $\mathbf{P}$

genau dann positiv definit ist, wenn das lineare System asymptotisch stabil ist. Für lineare Systeme führt die Direkte Methode also auf eine notwendige und hinreichende Stabilitätsbedingung.

Ein lineares System mit der Systemmatrix  $A$  ist genau dann asymptotisch stabil, wenn die Ljapunowgleichung (6.27) für eine beliebig gewählte positiv definite Matrix  $Q$  eine positiv definite Lösung  $P$  besitzt.

Die Lösung  $P$  der Ljapunowgleichung (6.27) kann nur für sehr einfache Systeme von Hand berechnet werden. Es gibt jedoch vielfältige Verfahren, die in Programmpaketen für die rechnergestützte Systemanalyse wie z. B. MATLAB enthalten sind und auch für das folgende Beispiel angewendet werden.

### Beispiel 6.3 (Forts.) Stabilitätsanalyse eines Pendels

Für das Pendel wird das linearisierte Modell (6.16)

$$\text{Pendel: } \dot{\mathbf{x}}(t) = \begin{pmatrix} 0 & 1 \\ -\frac{g}{l} & -\frac{k}{ml} \end{pmatrix} \mathbf{x}(t)$$

betrachtet und für dieses nach der hier erläuterten Methode eine Ljapunowfunktion

$$v(\mathbf{x}(t)) = \mathbf{x}^T(t) \mathbf{P} \mathbf{x}(t)$$

angesetzt. Für die Parameter

$$\begin{array}{lll} m & = & 3 \text{ kg} \quad \text{Masse} \\ l & = & 1 \text{ m} \quad \text{Pendellänge} \\ k & = & 1,5 \frac{\text{Ns}}{\text{rad}} \quad \text{Reibungskoeffizient} \end{array}$$

erhält man die Systemmatrix

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -9,81 & -0,5 \end{pmatrix}.$$

Nimmt man die positiv definite Matrix

$$\mathbf{Q} = \begin{pmatrix} 3 & 0,9 \\ 0,9 & 3 \end{pmatrix},$$

so liefert die Ljapunowgleichung die Lösung

$$\mathbf{P} = \begin{pmatrix} 31,61 & 0,15 \\ 0,15 & 3,31 \end{pmatrix}.$$

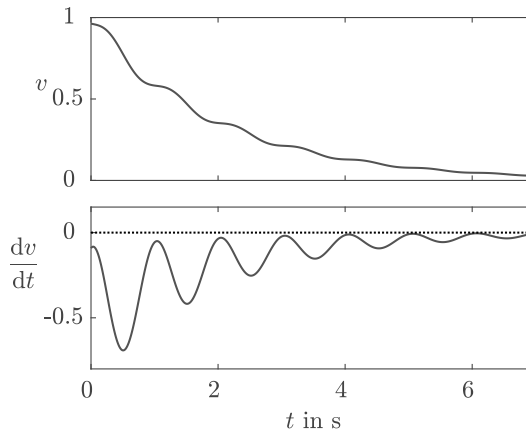
Das heißt,

$$v(\mathbf{x}(t)) = 31,61\phi^2(t) + 0,3\phi(t)\dot{\phi}(t) + 3,31\dot{\phi}^2(t) \quad (6.28)$$

ist eine Ljapunowfunktion für das Pendel. Die Eigenschaft (6.25) ist erfüllt, denn es gilt

$$\begin{aligned} \dot{v}(t) &= -\mathbf{x}^T(t) \mathbf{Q} \mathbf{x}(t) \\ &= -\left(3\phi^2(t) + 1,8\phi(t)\dot{\phi}(t) + 3\dot{\phi}^2(t)\right) < 0 \quad \text{für } \phi(t), \dot{\phi}(t) \neq 0. \end{aligned}$$





**Abb. 6.10:** Verhalten der Ljapunowfunktion für das Pendel

Der Verlauf von  $v(t)$  und  $\dot{v}(t)$  bei 10 Grad Anfangsauslenkung des Pendels, für die das lineare Modell die Pendelbewegung sehr gut beschreibt, ist in Abb. 6.10 dargestellt.

**Diskussion.** Diese Rechnung zeigt, dass man Ljapunowfunktionen auch ohne Energiebetrachtungen finden kann. Die dabei erhaltene Funktion erfüllt ihren Zweck sogar besser als die früher betrachtete Energiefunktion. Der Wert dieser Energiefunktion fiel zwar monoton ( $\dot{v}(t) \leq 0$ ), aber nicht streng monoton ( $\dot{v}(t) < 0$ ), so dass Zusatzüberlegungen notwendig waren, um die asymptotische Stabilität nachzuweisen. Die hier erhaltene Ljapunowfunktion fällt streng monoton und weist die asymptotische Stabilität ohne Zusatzbetrachtungen nach.

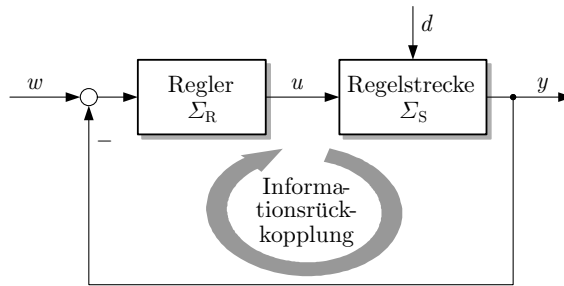
Für die Analyse des Pendels mit dem nichtlinearen Modell kann man die Funktion (6.28) verwenden und versuchen nachzuweisen, dass diese Funktion die Eigenschaft (6.25) erfüllt. Für einen vollständigen Stabilitätsnachweis muss man dabei zeigen, dass diese Funktion auf *allen* Trajektorien abnimmt.  $\square$

## 6.4 Stabilität von Regelkreisen

### 6.4.1 Stabilität der Regelstrecke und des Regelkreises

Mit einer Steuerung im geschlossenen Wirkungskreis kann man das dynamische Verhalten eines Systems nachhaltig verändern (Abb. 6.11). So wurde im Beispiel 2.10 auf S. 73 gezeigt, dass es auf Grund der Informationsrückkopplung von der Regelstrecke über den Regler zurück zur Regelstrecke möglich ist, ein instabiles invertiertes Pendel zu stabilisieren.

Die Möglichkeit, mit einer Regelung die Stabilitätseigenschaften eines Systems zu verändern, umfasst – bei Verwendung eines unzuweckmäßigen Reglers – auch die Möglichkeit, dass man bei einer stabilen Strecke einen instabilen Regelkreis erzeugt. Ganz allgemein gilt, dass die Stabilität der Regelstrecke und des Reglers weder notwendig noch hinreichend für die Stabilität des Regelkreises ist. Aus den Stabilitätseigenschaften beider Bestandteile eines Regelkreises kann man also nicht ohne genauere Analyse auf die Stabilitätseigenschaft des Regelkreises



**Abb. 6.11:** Informationsrückkopplung im Regelkreis

schließen. Diese Aussage gilt für nichtlineare und lineare Regelkreise gleichermaßen, auch wenn im Folgenden nur lineare Regelkreise näher untersucht werden.

Im Abschn. 6.2.1 wurde gezeigt, dass die Stabilität linearer Systeme unabhängig davon ist, ob das System durch externe Signale beeinflusst wird. Bei der Stabilitätsanalyse linearer Regelkreise kann man sich deshalb auf den autonomen Regelkreis beschränken, also die Störung  $d(t)$  und die Führungsgröße  $w(t)$  gleich null setzen.

Für die Stabilitätsanalyse mit den in diesem Kapitel beschriebenen Methoden muss das Zustandsraummodell des Regelkreises aufgestellt werden. Aus dem Streckenmodell

$$\Sigma_S : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}d(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) \end{cases}$$

und dem Reglergesetz, für das hier eine proportionale Rückführung

$$\Sigma_R : u(t) = -k_P(y(t) - w(t))$$

eingesetzt wird, erhält man für den Regelkreis  $\bar{\Sigma}$  mit  $d(t) = 0$  und  $w(t) = 0$  die Beziehung

$$\bar{\Sigma} : \begin{cases} \dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{b}k_P \mathbf{c}^T) \mathbf{x}(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t), \end{cases}$$

von der für die Stabilitätsanalyse nur die Systemmatrix

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{b}k_P \mathbf{c}^T$$

maßgebend ist. Die charakteristische Gleichung des Regelkreises lautet deshalb

$$\det(\lambda \mathbf{I} - \bar{\mathbf{A}}) = \det(\lambda \mathbf{I} - \mathbf{A} + \mathbf{b}k_P \mathbf{c}^T) = 0. \quad (6.29)$$

Formt man dieses Polynom um, so erhält man eine Beziehung der Form (6.13)

$$a_n(k_P)\lambda^n + a_{n-1}(k_P)\lambda^{n-1} + \dots + a_1(k_P)\lambda + a_0(k_P) = 0, \quad (6.30)$$

in der die Koeffizienten  $a_i$ , ( $i = 0, 1, \dots, n$ ) vom Reglerparameter  $k_P$  abhängen. Erwartungsgemäß ist die Stabilität auch von  $k_P$  abhängig.

Für die Bestimmung von Reglerparametern, für die der Regelkreis stabil ist, setzt man zweckmäßigerweise das Hurwitzkriterium ein, denn dieses Kriterium kann angewendet werden, ohne dass alle Systemparameter zahlenmäßig bekannt sind. Für die charakteristische Gleichung (6.30), in der  $k_P$  ein noch festzulegender Reglerparameter ist, erhält man aus dem Hurwitzkriterium Ungleichungen für  $k_P$ , deren Erfüllung die Stabilität des Kreises garantiert (vgl. Aufgabe 6.6). Würde man statt des Hurwitzkriteriums das in Satz 6.1 angegebene Stabilitätskriterium einsetzen, so müsste zunächst  $k_P$  festgelegt werden, weil die Eigenwerte des Regelkreises nur dann zahlenmäßig ausgerechnet werden können, wenn alle Modellparameter bekannt sind.

**Aufgabe 6.5\*** Stabilisierung eines Systems zweiter Ordnung

Gegeben ist ein System zweiter Ordnung

$$\Sigma_S : \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} -\frac{1}{T_1} & 0 \\ \frac{1}{T_2} & -\frac{1}{T_2} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} \frac{1}{T_1} \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ \frac{k_d}{T_2} \end{pmatrix} d(t) \\ y(t) = (0 \quad k_s) \mathbf{x}(t), \end{cases}$$

dessen Parameter durch die beiden Zeitkonstanten  $T_1, T_2$ , den Parameter  $k_d$  sowie die statische Verstärkung  $k_s$  dargestellt sind. Mit einem P-Regler

$$\Sigma_R : u(t) = -k_P y(t)$$

wird der Regelkreis  $\bar{\Sigma}$  geschlossen, dessen Stabilität im Folgenden betrachtet wird.

1. Zeichnen Sie das Blockschaltbild des Regelkreises.
2. Weisen Sie nach, dass der Regelkreis  $\bar{\Sigma}$  für beliebige Zeitkonstanten  $T_1, T_2$ , statische Verstärkung  $k_s$  und Reglerverstärkung  $k_P$  asymptotisch stabil ist, sofern  $k_P k_s > 0$  ist.
3. Durch die geeignete Wahl der Reglerverstärkung  $k_P$  soll erreicht werden, dass eine konstante Störgröße  $d(t) = \sigma(t)$  im eingeschwungenen Regelkreis nur auf eine Regelabweichung von höchstens 5% führt. Wie groß muss die Reglerverstärkung gewählt werden? Ist der Regelkreis für diese Reglerverstärkung stabil?
4. Wenn zwischen die Regelstrecke und den Regler ein Messglied

$$\Sigma_M : \begin{cases} \dot{x}_m(t) = -\frac{1}{T_m} x_m(t) + \frac{1}{T_m} y(t) \\ y_m(t) = x_m(t) \end{cases}$$

eingebaut wird, das den Messwert  $y_m(t)$  für die Regelgröße  $y(t)$  liefert, wird der Regelkreis mit dem Regler

$$\Sigma_R : u(t) = -k_P y_m(t)$$

geschlossen, der die Messgröße  $y_m(t)$  zurückführt. Erweitern Sie das Blockschaltbild dementsprechend.

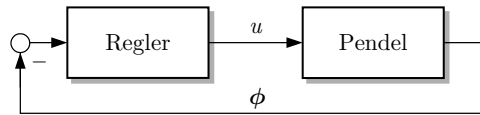
5. Ermitteln Sie die Bedingungen, die die Reglerverstärkung  $k_P$  jetzt erfüllen muss, damit der Regelkreis stabil ist. Welche Konsequenz hat der Einbau des Messgliedes in Bezug auf die Erfüllung der für das Störverhalten im vorherigen Aufgabenteil angegebenen Forderung?  $\square$

**Aufgabe 6.6\*** Stabilitätsprüfung und Stabilisierung des geregelten invertierten Pendels

Das in Abb. 2.22 auf S. 73 gezeigte invertierte Pendel wird durch folgendes linearisiertes Zustandsraummodell beschrieben:

$$\text{Pendel : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ \frac{3}{ml^2} \end{pmatrix} u(t), & \mathbf{x}(0) = \begin{pmatrix} \phi(0) \\ \dot{\phi}(0) \end{pmatrix} \\ y(t) = (1 \ 0) \mathbf{x}(t) \end{cases} \quad (6.31)$$

(zur Herleitung dieses Modells vgl. Gln. (A.13) auf S. 583).



**Abb. 6.12:** Stabilisierung des invertierten Pendels durch eine Rückführung des Pendelwinkels  $\phi$

1. Das Pendel ist offensichtlich instabil. Weisen Sie dies mit Hilfe des Hurwitzkriteriums nach.
2. Das Pendel soll dadurch stabilisiert werden, dass man den Winkel  $\phi(t)$  misst, mit dem Faktor  $k_P$  multipliziert und daraus die Stellgröße  $u(t)$  berechnet:

$$\text{Regler : } u(t) = -k_P \phi(t)$$

(Abb. 6.12). Für welches Intervall für  $k_P$  ist das geregelte Pendel stabil? □

**Aufgabe 6.7\*** Stabilität des Temperaturregelkreises eines Reaktors

In Erweiterung des in Aufg. 4.2 angegebenen Modells für das Temperaturverhalten eines Rührkesselreaktors wird jetzt mit einem Zustandsraummodell dritter Ordnung gerechnet, in dem auch das Verhalten des Temperatursensors berücksichtigt ist:

$$\text{Rührkessel : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} -8,16 & -17,82 & -10,03 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} u(t) \\ y(t) = (0 \ 0 \ 14,36) \mathbf{x}(t). \end{cases} \quad (6.32)$$

$u(t)$  ist die Stellgröße für die Temperaturregelung,  $y(t)$  der Messwert für die Reaktortemperatur. Die Temperatur soll durch eine I-Regelung

$$\text{I-Regler : } \begin{cases} \dot{x}_r(t) = y(t) - w(t) \\ u(t) = -k_I x_r(t) \end{cases} \quad (6.33)$$

auf den Sollwert  $w(t) = \bar{w}$  gebracht werden. Für welche Reglerparameter  $k_I$  ist der Regelkreis stabil? □

### 6.4.2 Robuste Stabilität

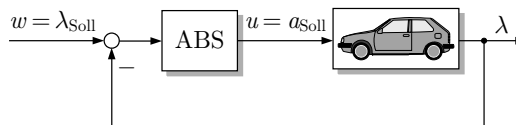
Häufig sind die bei der Stabilitätsanalyse eingesetzten Modelle mit unsicheren Parametern versehen. Entweder ändern sich die Systemeigenschaften während des Betriebs oder die Parameter können auf Grund unvollständiger Systemkenntnisse nicht exakt bestimmt werden. Wird das Modell durch Linearisierung eines nichtlinearen Modells um einen Arbeitspunkt gewonnen, so ändern sich die Modellparameter in Abhängigkeit vom aktuellen Arbeitspunkt. Das Modell kann dann mit sehr guter Genauigkeit gelten, wenn man die Parameter dem aktuellen Arbeitspunkt anpasst, aber es gilt eben nicht mit konstanten Parametern.

Unter diesen Bedingungen ist es wichtig zu wissen, ob die Stabilität eines Regelkreises auch bei Veränderungen der aktuellen Parameterwerte bzw. für alle während des Betriebs auftretenden Parameterkombinationen gewährleistet ist. Wenn Parameteränderungen die Stabilität nicht gefährden, spricht man von robuster Stabilität. Ein Regler, für den die Stabilität des Regelkreises für vorgegebene Parameterintervalle gesichert ist, nennt man einen *robusten Regler*.

Eine einfache Methode zur Überprüfung der robusten Stabilität besteht darin, das Hurwitzkriterium anzuwenden und dabei Ungleichungen abzuleiten, die die unsicheren Parameter erfüllen müssen, damit das System stabil bleibt. Statt den Wert eines Parameters bei der Ableitung der charakteristischen Gleichung einzusetzen, arbeitet man mit dem Symbol des Parameters und ermittelt aus den im Hurwitzkriterium stehenden Ungleichungen Bereiche, in denen sich dieser Parameter aufhalten muss, damit das betrachtete System asymptotisch stabil ist.

**Aufgabe 6.8\*** Robustheit eines Antiblockiersystems

Ein Antiblockiersystem soll die vom Fahrer gewünschte Bremsbeschleunigung so verändern, dass die geforderte Kraftschlussbeanspruchung nicht über dem physikalisch realisierbaren Wert  $\mu_H$  liegt. Aus technischen Gründen wird dabei nicht die Istbeschleunigung, sondern der Schlupf  $\lambda(t)$  als Regelgröße und dementsprechend ein Sollschlupf  $\lambda_{\text{Soll}}$  als Führungsgröße verwendet, wobei bei allen Größen die im linearisierten Modell vorkommenden Abweichungen vom Arbeitspunkt gemeint sind. Abbildung 6.13 zeigt den betrachteten Regelkreis.



**Abb. 6.13:** Antiblockiersystem als Regler in einem Schlupfregelkreis

Die Regelstrecke hat das linearisierte Modell

$$\text{Fahrzeug : } \begin{cases} \dot{x}(t) = -\frac{mr^2g}{J\bar{v}}k_{\mu\lambda}(\bar{\lambda})x(t) - \frac{mr}{J}u(t), & x(0) = x_0 \\ y(t) = -\frac{r}{\bar{v}}x(t), \end{cases}$$

in dem die Winkelgeschwindigkeit  $x(t) = \omega(t)$  des Rades als Zustandsgröße auftritt und dessen Parameter von den Arbeitspunktwerten  $\bar{v}$  und  $\bar{\lambda}$  abhängen (vgl. Aufgabe 3.6 auf S. 114). Als Antiblockiersystem wird ein PI-Regler

$$\text{PI-Regler: } \begin{cases} \dot{x}_r(t) = y(t) - w(t), & x_r(0) = x_{r0} \\ u(t) = -k_P(y(t) - w(t)) - k_I x_r(t) \end{cases}$$

eingesetzt.

1. Stellen Sie das Zustandsraummodell des Regelkreises  $\bar{\Sigma}$  auf.
2. Unter welcher Bedingung ist der Regelkreis stabil?
3. Die Parameter der Regelstrecke ändern sich in Abhängigkeit von der Geschwindigkeit  $\bar{v}$ , vom Schlupf  $\bar{\lambda}$  sowie vom Straßenbelag, der die  $\mu(\lambda)$ -Kennlinie bestimmt (vgl. Abb. 3.15 auf S. 108). Gibt es Reglerparameter, für die der Regelkreis unter allen diesen Umständen stabil ist?  $\square$

### Aufgabe 6.9 Robuste Stabilisierung des invertierten Pendels



In der Aufgabe 6.6 wurde gezeigt, dass das invertierte Pendel nicht durch eine proportionale Rückführung stabilisiert werden kann. Verwenden Sie deshalb in dieser Aufgabe eine Regelung, die sowohl den Winkel als auch die Winkelgeschwindigkeit zurückführt (PD-Regler):

$$\text{PD-Regler: } u(t) = -k_P \phi(t) - k_D \dot{\phi}(t).$$

Untersuchen Sie das Pendel für die folgenden Parameter und verwenden Sie die Maßeinheit Sekunde für die Zeit:

$$\begin{aligned} m &= 0,3 \text{ kg} && \text{Masse der Stange} \\ l &= 0,5 \text{ m} && \text{Länge der Stange} \\ g &= 9,81 \frac{\text{m}}{\text{s}^2} && \text{Erdbeschleunigung} \end{aligned}$$

1. Ermitteln Sie unter Verwendung des Pendelmodells (6.31) die Bereiche für die Reglerparameter  $k_P$  und  $k_D$ , für die der Regelkreis stabil ist.
2. Stellen Sie die Eigenbewegung des Pendels für den durch  $\phi(0) = 10^\circ$  und  $\dot{\phi}(0) = 0$  beschriebenen Anfangszustand für unterschiedliche Reglerparameter aus den ermittelten Intervallen dar.
3. Zeigen Sie anhand der Eigenbewegung, dass das Pendel für Reglerparameter, die außerhalb der von Ihnen ermittelten Bereiche liegen, instabil ist.
4. Für eine Demonstration wollen Sie nicht nur zeigen, dass Ihr Pendel durch den verwendeten PD-Regler asymptotisch stabilisiert wird, sondern Sie wollen auch die Robustheit der Regelung dadurch veranschaulichen, dass Sie auf das Pendel ein gefülltes Glas Bier stellen (was natürlich nicht herunterfallen soll). Unter welchen Bedingungen ist das geregelte Pendel mit und ohne Bierglas asymptotisch stabil. Wie sieht die Eigenbewegung des Pendels in beiden Fällen aus? Sind Sie sich sicher, dass das Glas bei den schwingenden Bewegungen, die das Pendel für Ihren Regler ausführt, nicht herunter fällt?  $\square$

## 6.5 Ausblick: Weitere Verfahren für die Stabilitätsprüfung

Da die Sicherung der Stabilität ein fundamentales Problem der Regelungstechnik ist, wurden für die Stabilitätsanalyse rückgekoppelter Systeme vielfältige Methoden entwickelt. Dieses Kapitel konnte nur einen kleinen Einblick in die Probleme und Lösungsmethoden vermitteln.

Eine große Gruppe der heute praktisch eingesetzten Stabilitätskriterien für lineare Systeme bezieht sich auf die Systemdarstellung im Frequenzbereich. Bei dieser hier nicht behandelten Beschreibungsform wird das System durch eine Übertragungsfunktion repräsentiert, die angibt, wie das System die Amplitude und die Phase einer sinusförmigen Eingangsgröße verändert, um daraus die Ausgangsgröße zu erzeugen. Im Unterschied zur Differentialgleichung bzw. zum Zustandsraummodell wird das System dabei durch algebraische Gleichungen beschrieben. Für die Stabilitätsanalyse gibt es Kriterien, die sich auf die Übertragungsfunktion beziehen.

Für die technische Interpretation und den Reglerentwurf sind vor allem diejenigen Kriterien interessant, die die Stabilität des geschlossenen Kreises anhand der Übertragungsfunktion der offenen Kette überprüfen. Da in der Frequenzbereichsdarstellung die Übertragungsfunktion des Reglers linear in die Übertragungsfunktion der offenen Kette eingeht, kann man aus diesen Stabilitätskriterien auf besonders einfache Weise Rückschlüsse auf die Wahl der Reglerparameter ziehen.

Diese Vorgehensweise lässt sich auf nichtlineare Systeme übertragen, indem man diese Systeme gleichfalls unter der Wirkung sinusförmiger Eingangsgrößen untersucht. Allerdings kann das als Beschreibungsform bezeichnete Modell die Übertragungsfunktion linearer Systeme nur näherungsweise ersetzen. Die Methode der harmonischen Balance untersucht, unter welchen Bedingungen die Ausgangsgröße der offenen Kette dieselbe Amplitude und dieselbe Phasenlage wie die sinusförmige Eingangsgröße hat, so dass sich das Signal im geschlossenen Kreis reproduziert, der geschlossene Kreis also eine Dauerschwingung ausführt. Dieses Verfahren wird für einfache nichtlineare Regelkreise praktisch eingesetzt.

Um notwendige und hinreichende Stabilitätskriterien für nichtlineare Systeme anwenden zu können, muss man allerdings mit einem nichtlinearen Zustandsraummodell arbeiten. Obwohl es bereits für große Klassen nichtlinearer Systeme wirksame Prüfverfahren gibt, ist die Stabilitätsanalyse nichtlinearer Systeme immer noch ein aktueller Forschungsgegenstand. Gesucht sind Bedingungen, die man bei der Wahl des Reglers erfüllen muss, um auf einen stabilen Regelkreis zu kommen. In engem Zusammenhang damit steht die Frage, ob es überhaupt möglich ist, durch die geeignete Wahl eines Reglers ein gegebenes instabiles System zu stabilisieren. Wenn die Frage bejaht werden kann, nennt man das System *stabilisierbar*.

## Literaturhinweise

Die Stabilität nichtlinearer Systeme wird häufig mit einem linearisierten Modell überprüft, wenn man weiß, dass sich das System in der Umgebung eines Arbeitspunkts ähnlich wie ein lineares System verhält. Ausführliche Darstellungen hierzu finden sich in [28] oder [51].

Einen Einblick in die moderne Theorie nichtlinearer Systeme gibt z. B. [45].





# 7

## Einschleifige Regelkreise

Anhand des Modells des Standardregelkreises werden in diesem Kapitel grundlegende Eigenschaften einschleifiger Regelkreise wie Sollwertfolge, Störkompensation und Robustheit behandelt. Dabei wird auch gezeigt, welche dynamischen Komponenten die eingesetzten Regler besitzen müssen, um die genannten Eigenschaften zu gewährleisten.

### 7.1 Regelungsaufgabe für kontinuierliche Systeme

Die Regelungsaufgabe besteht im Entwurf und der Realisierung eines Reglers, mit dem der Regelkreis die an ihn gestellten Güteforderungen erfüllt. Im Folgenden wird der in Abb. 7.1 gezeigte Standardregelkreis betrachtet und die folgende Entwurfsaufgabe behandelt:

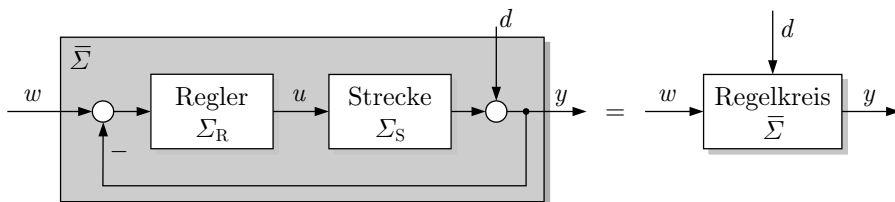


Abb. 7.1: Standardregelkreis

### Reglerentwurfsaufgabe für kontinuierliche Systeme

Gegeben: Modell der Regelstrecke  $\Sigma_S$   
 Forderungen an das Verhalten des Regelkreises  $\bar{\Sigma}$   
 Gesucht: Reglergesetz  $\Sigma_R$

Die Güteforderungen an den Regelkreis lassen sich in vier Gruppen unterteilen:

- (I) **Stabilitätsforderung:** Der geschlossene Regelkreis  $\bar{\Sigma}$  muss asymptotisch stabil sein.
- (II) **Störkompensation und Sollwertfolge:** Für eine vorgegebene Klasse von Stör- und Führungssignalen soll die Regelgröße der Führungsgröße asymptotisch folgen:
 
$$\lim_{t \rightarrow \infty} (w(t) - y(t)) = 0. \quad (7.1)$$
- (III) **Dynamikforderungen:** Die Kennwerte des dynamischen Übergangsverhaltens sollen vorgegebene Forderungen erfüllen.
- (IV) **Robustheit:** Die Güteforderungen (I) – (III) sollen trotz Unbestimmtheiten in dem für den Reglerentwurf eingesetzten Modell der Regelstrecke erfüllt werden.

Die Erfüllung der Stabilitätsforderung ist eine Grundvoraussetzung für die Anwendbarkeit eines Reglers. Deshalb spielt die Stabilität bei jedem Reglerentwurf eine große Rolle und muss vor der Inbetriebnahme des Reglers überprüft werden. Die durch den Regler realisierte Rückführung der Ausgangsgröße  $y(t)$  auf die Eingangsgröße  $u(t)$  der Regelstrecke macht es einerseits möglich, dass eine instabile Regelstrecke durch einen geeignet gewählten Regler stabilisiert wird. Andererseits bringt diese Rückkopplung die Gefahr mit sich, dass ein instabiler Regelkreis  $\bar{\Sigma}$  entsteht, selbst wenn die Regelstrecke  $\Sigma_S$  stabil ist.

Störkompensation und Sollwertfolge können nicht für beliebige Störsignale  $d(t)$  und Führungssignale  $w(t)$  erreicht werden, wie im Abschn. 7.3.1 noch erläutert wird. Hier werden vor allem sprungförmige Signale untersucht, die durch die Beziehungen

$$\begin{aligned} d(t) &= \bar{d}\sigma(t) \\ w(t) &= \bar{w}\sigma(t) \end{aligned}$$

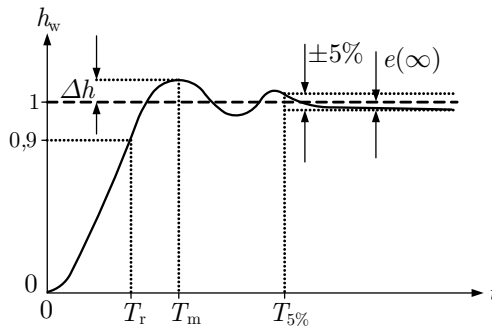
beschrieben werden. Dabei sind  $\bar{d}$  und  $\bar{w}$  beliebige Amplituden und  $\sigma(t)$  ist die Sprungfunktion (4.24). Die Sollwertfolge bedeutet, dass die Forderung (7.1) für sprungförmige Signale  $d(t)$  und  $w(t)$  mit beliebigen Amplituden  $\bar{d}$  bzw.  $\bar{w}$  erfüllt wird.

**Kennwerte des Übergangsverhaltens.** Die Dynamikforderungen beziehen sich auf das E/A-Verhalten des Regelkreises, wobei zwei Eingänge und ein Ausgang betrachtet werden müssen (Abb. 7.1). Sie werden i. Allg. durch Kennwerte beschrieben, die die Führungsübergangsfunktion oder die Störübergangsfunktion des Regelkreises betreffen:

- Die *Führungsübergangsfunktion*  $h_w(t)$  entsteht am Ausgang des Regelkreises, wenn die Führungsgröße sprungförmig verändert wird ( $w(t) = \sigma(t)$ ),

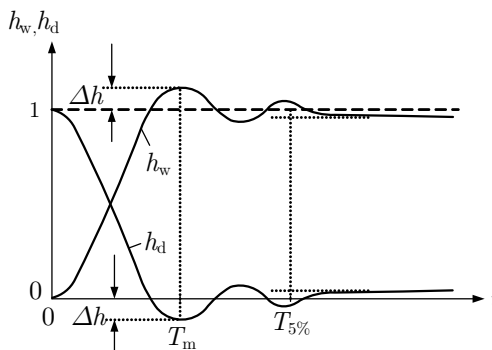
- die *Störübergangsfunktion*  $h_d(t)$ , wenn sich die Störgröße sprungförmig ändert ( $d(t) = \sigma(t)$ ),

wobei die jeweils andere Eingangsgröße des Regelkreises verschwindet.



**Abb. 7.2:** Führungsübergangsfunktion des Regelkreises mit Kennzeichnung wichtiger Kennwerte

Abbildung 7.2 zeigt gebräuchliche Kennwerte für die Führungsübergangsfunktion. Die Überschwingweite  $\Delta h$  beschreibt das erste Maximum und die Überschwingzeit  $T_m$  den Zeitpunkt dieses Maximums. Die Anstiegszeit  $T_r$  ist erreicht, wenn die Führungsübergangsfunktion den Wert 0,9 überschreitet. Die Beruhigungszeit  $T_{5\%}$  beschreibt den Zeitpunkt, bei dem die Führungsübergangsfunktion zum letzten Mal in ein Toleranzband um den Endwert 1 mit der Breite  $2 \cdot 5\%$  einschwingt. Die in der Abbildung gekennzeichnete bleibende Regelabweichung  $e(\infty)$  tritt nicht auf, wenn die Güteforderung (II) erfüllt wird.



**Abb. 7.3:** Vergleich von Störübergangsfunktion und Führungsübergangsfunktion

Abbildung 7.3 vergleicht die Störübergangsfunktion  $h_d(t)$  mit der Führungsübergangsfunktion  $h_w(t)$  und zeigt, dass das Einschwingen der Regelgröße auf den Sollwert  $w = 0$

mit derselben Überschwingzeit, Überschwingweite und Beruhigungszeit erfolgt wie das Einschwingen der Führungsübergangsfunktion auf den Sollwert  $w = 1$ . Aus diesem Grund kann man im Standardregelkreis das Stör- und das Führungsverhalten nicht getrennt voneinander beeinflussen. Voraussetzung hierfür ist, dass die Störung  $d(t)$  am Ausgang der Regelstrecke eingreift, wie es in Abb. 7.1 (links) gezeigt wird.

Die Robustheitsforderung wird gestellt, weil das für den Reglerentwurf verwendete Modell i. Allg. erhebliche Modellunbestimmtheiten aufweist und trotzdem gesichert sein soll, dass der Regler die gegebenen Güteforderungen im Zusammenspiel mit der realen Regelstrecke erfüllt. Neben den Unsicherheiten, mit denen viele Modellparameter behaftet sind, entstehen Modellunbestimmtheiten beispielsweise auch aus der Linearisierung des Regelstreckenmodells um einen Arbeitspunkt. Durch systematische Abschätzung der Robustheit oder durch Simulationsuntersuchungen mit veränderten Modellparametern kann man überprüfen, ob der entworfene Regler die Robustheitsforderung erfüllt. Obwohl die systematische Robustheitsanalyse im Folgenden nicht ausführlich beschrieben werden kann, wird bei vielen Beispielen auf den Robustheitsaspekt der Regelung eingegangen.

## 7.2 Modell des Standardregelkreises

### 7.2.1 Linearer Regelkreis

Im Folgenden wird der in Abb. 7.4 gezeigte Standardregelkreis betrachtet. Das Signal  $r(t)$  stellt die Messstörung dar, die beim Reglerentwurf später nicht beachtet wird, deren Wirkung hier aber mit untersucht werden soll.

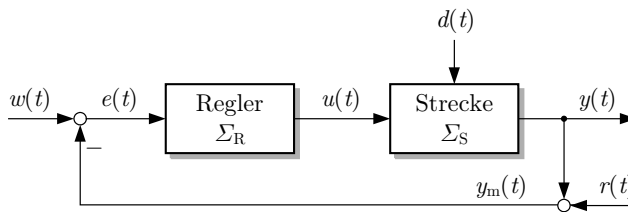


Abb. 7.4: Standardregelkreis

Die Regelstrecke hat das Zustandsraummodell

$$\Sigma_S : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}d(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t), \end{cases} \quad (7.2)$$

das gegenüber den bisher verwendeten Modellen um den Summanden  $\mathbf{g}d(t)$  erweitert ist, um den Einfluss der Störung  $d(t)$  darzustellen. Wenn die Störung wie in Abb. 7.1 am Ausgang der Regelstrecke angreift, entfällt dieser Summand und  $+d(t)$  tritt als zusätzlicher Summand in der

Ausgabegleichung auf. Das in Abb. 7.4 eingetragene Messrauschen  $r(t)$  bewirkt, dass an Stelle von  $y(t)$  die Größe

$$y_m(t) = y(t) + r(t)$$

durch den Regler zurückgeführt wird. Zur Vereinfachung der Darstellung wird davon ausgegangen, dass das System bezüglich der Eingangsgröße  $u(t)$  und der Störgröße  $d(t)$  nicht sprunghaft ist. Diese Voraussetzung ist für eine direkt den Ausgang  $y(t)$  beeinflussende Störung  $d(t)$  nicht erfüllt. Die folgenden Gleichungen können jedoch ohne Probleme für diesen Fall erweitert werden.

**Regelkreis mit proportionalem Regler.** Es wird zunächst der sehr einfache Regler

$$\text{P-Regler: } u(t) = -k_P (y_m(t) - w(t)) \quad (7.3)$$

betrachtet, bei dem die Stellgröße proportional zur Regelabweichung  $w(t) - y_m(t)$  ist. Dieser Regler wird als P-Regler (Proportionalregler) bezeichnet. Das Modell des geschlossenen Kreises erhält man durch Kombination der Gln. (7.2) und (7.3):

$$\bar{\Sigma} : \begin{cases} \dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{b} k_P \mathbf{c}^T) \mathbf{x}(t) + \mathbf{g} d(t) + \mathbf{b} k_P w(t) - \mathbf{b} k_P r(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t). \end{cases} \quad (7.4)$$

In diesem Zustandsraummodell treten drei Eingänge auf:  $w(t)$ ,  $d(t)$  und  $r(t)$ .

Den Einfluss des Reglers auf die Dynamik der Strecke erkennt man an der Veränderung der Systemmatrix. An Stelle von  $\mathbf{A}$  steht im Modell jetzt die Matrix

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{b} k_P \mathbf{c}^T,$$

wobei der Querstrich über  $\mathbf{A}$  den Bezug zum Regelkreis kennzeichnet.

### Beispiel 7.1 Zustandsraummodell des geregelten Gleichstrommotors

Für den im Beispiel 3.2 auf S. 90 behandelten Gleichstrommotor mit dem Zustandsraummodell (3.16)

$$\text{Gleichstrommotor : } \begin{cases} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} \\ \frac{k_T}{J} & -\frac{k_B}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix} u(t) \\ \begin{pmatrix} i(0) \\ \omega(0) \end{pmatrix} = \begin{pmatrix} i_0 \\ \omega_0 \end{pmatrix} \\ y(t) = (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} \end{cases}$$

wird eine proportionale Drehzahlregelung (7.3) mit  $y(t) = y_m(t)$

$$\text{Drehzahlregelung: } u(t) = -k_P (y(t) - w(t)) = -k_P \omega(t) + k_P \omega_{\text{Soll}}(t)$$

eingesetzt, bei der der Sollwert mit  $\omega_{\text{Soll}}(t)$  bezeichnet wird, um Verwechslungen des sonst gebräuchlichen Symbols  $w(t)$  mit der Winkelgeschwindigkeit  $\omega(t)$  zu vermeiden. Für den Regelkreis erhält man das Zustandsraummodell

$$\text{geregelter Motor : } \begin{cases} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{k_P}{L} \\ 0 \end{pmatrix} \omega_{\text{Soll}}(t) \\ y(t) = (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix}. \end{cases}$$

Die Systemmatrix

$$\bar{A} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix}$$

ist gegenüber der Systemmatrix des Motors nur im oberen rechten Element verändert, weil die Stellgröße nur die Ableitung der ersten Zustandsvariablen direkt beeinflusst und der Regler als Messgröße die zweite Zustandsvariable  $\omega(t)$  erhält. Dennoch kann der Regler die dynamischen Eigenschaften des Regelkreises entscheidend beeinflussen. Beispielsweise sind beide Eigenwerte von  $\bar{A}$  von  $k_P$  abhängig.  $\square$

**Regelkreis mit dynamischem Regler.** Als Regler können, im Prinzip, beliebige dynamische Systeme

$$\Sigma_R : \begin{cases} \dot{\mathbf{x}}_r(t) = \mathbf{A}_r \mathbf{x}_r(t) + \mathbf{b}_r(y_m(t) - w(t)), & \mathbf{x}_r(0) = \mathbf{x}_{r0} \\ u(t) = -\mathbf{k}_r^T \mathbf{x}_r(t) - k_P(y_m(t) - w(t)) \end{cases} \quad (7.5)$$

eingesetzt werden, wobei  $\mathbf{x}_r(t)$  den Zustandsvektor und  $\mathbf{A}_r$  die Systemmatrix des Reglers bezeichnet. Der Reglerzustand wird nach Multiplikation mit dem Zeilenvektor  $\mathbf{k}_r^T$  auf den Streckeneingang zurückgeführt. Außerdem hat der Regler denselben Proportionalanteil wie der P-Regler (7.3). Die Minuszeichen in der Ausgabegleichung (7.5) sollen kennzeichnen, dass der Regler zur Regelstrecke stets gegengekoppelt ist. Da die Ausgabegleichung wie angegeben mit dem Minuszeichen verwendet wird, sind die Reglerparameter meist positiv.

Wird der dynamische Regler (7.5) verwendet, so setzt sich der Zustand des Regelkreises aus dem Zustand der Regelstrecke und dem Zustand des Reglers zusammen

$$\bar{\mathbf{x}}(t) = \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{x}_r(t) \end{pmatrix},$$

wobei beide Zustandsvektoren in  $\bar{\mathbf{x}}$  untereinander geschrieben sind. Das Zustandsraummodell des Regelkreises erhält man durch Kombination der Gln. (7.2) und (7.5):

$$\bar{\Sigma} : \begin{cases} \begin{pmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_r(t) \end{pmatrix} = \begin{pmatrix} \mathbf{A} - \mathbf{b}k_P \mathbf{c}^T & -\mathbf{b}\mathbf{k}_r^T \\ \mathbf{b}_r \mathbf{c}^T & \mathbf{A}_r \end{pmatrix} \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{x}_r(t) \end{pmatrix} + \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix} d(t) + \\ \quad + \begin{pmatrix} \mathbf{b}k_P \\ -\mathbf{b}_r \end{pmatrix} w(t) + \begin{pmatrix} -\mathbf{b}k_P \\ \mathbf{b}_r \end{pmatrix} r(t) \\ y(t) = (\mathbf{c}^T \quad \mathbf{0}^T) \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{x}_r(t) \end{pmatrix}. \end{cases}$$

Dieses Modell hat die Form

$$\bar{\Sigma} : \begin{cases} \dot{\bar{\mathbf{x}}}(t) = \bar{\mathbf{A}}\bar{\mathbf{x}}(t) + \bar{\mathbf{b}}_d d(t) + \bar{\mathbf{b}}_w w(t) + \bar{\mathbf{b}}_r r(t), & \bar{\mathbf{x}}(0) = \bar{\mathbf{x}}_0 \\ y(t) = \bar{\mathbf{c}}^T \bar{\mathbf{x}}(t), \end{cases} \quad (7.6)$$

wobei die neu eingeführten Größen die Parameter der Strecke und des Reglers enthalten, wie ein Vergleich mit den vorher angegebenen Gleichungen zeigt. Von besonderer Bedeutung ist dabei wieder die Systemmatrix

$$\bar{\mathbf{A}} = \begin{pmatrix} \mathbf{A} - \mathbf{b}k_P \mathbf{c}^T & -\mathbf{b}k_r^T \\ \mathbf{b}_r \mathbf{c}^T & \mathbf{A}_r \end{pmatrix}$$

des Regelkreises, mit Hilfe derer beispielsweise die Stabilität des Regelkreises überprüft werden kann.

Das Modell zeigt, dass das Messrauschen  $r(t)$  und die Führungsgröße  $w(t)$  über dieselben Eingangsvektoren, jedoch mit unterschiedlichem Vorzeichen, auf den Regelkreis einwirken ( $\bar{\mathbf{b}}_w = -\bar{\mathbf{b}}_r$ ), was man sich auch anhand des Blockschaltbilds erklären kann. Beide Signale werden deshalb in derselben Weise verstärkt oder abgeschwächt am Ausgang des Regelkreises wirksam.

Da man die Signalwege von  $w(t)$  nach  $y(t)$  und von  $r(t)$  nach  $y(t)$  nicht unabhängig voneinander beeinflussen kann, wird beim Reglerentwurf häufig die Wirkung des Messrauschens ignoriert und das hier angegebene Modell mit  $r(t) = 0$  verwendet. Man muss sich beim Reglerentwurf im Klaren sein, dass ein Regelkreis, dessen Regelgröße schnell auf Änderungen der Führungsgröße reagiert, auch das Messrauschen gut überträgt. Abhilfe schafft die Erweiterung des Standardregelkreises um ein Filter, das das Messrauschen unterdrückt.

### 7.2.2 Nichtlinearer Regelkreis

Die in Abb. 7.4 gezeigte Struktur des Regelkreises bleibt unverändert, wenn die Regelstrecke nichtlinear ist. Um das Modell des Regelkreises zu erhalten, muss dann jedoch das nichtlineare Zustandsraummodell (3.26)

$$\Sigma_S : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = h(\mathbf{x}(t)) \end{cases}$$

mit dem Reglergesetz kombiniert werden. Bei Verwendung eines P-Reglers

$$\Sigma_R : u(t) = -k_P(y(t) - w(t))$$

erhält man dabei das Modell

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{g}(\mathbf{x}(t), -k_P(h(\mathbf{x}(t)) - w(t))), & \mathbf{x}(0) &= \mathbf{x}_0 \\ y(t) &= h(\mathbf{x}(t)), \end{aligned}$$

das in

$$\bar{\Sigma} : \begin{cases} \dot{\mathbf{x}}(t) = \bar{\mathbf{g}}(\mathbf{x}(t), w(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = h(\mathbf{x}(t)) \end{cases} \quad (7.7)$$

umgeformt werden kann. Die in der Zustandsgleichung auf der rechten Seite stehende Funktion  $\bar{\mathbf{g}}$  wird von der Regelstrecke  $\Sigma_S$  und vom Regler  $\Sigma_R$  bestimmt. Inwieweit man diese Funktion als geschlossenen analytischen Ausdruck aufschreiben kann, hängt von den nichtlinearen Eigenschaften der Regelstrecke ab. Als Argument tritt bei dieser Funktion neben dem Zustand  $\mathbf{x}(t)$  die Führungsgröße  $w(t)$  auf.

Die Funktion  $h$  in der Ausgangsgleichung bleibt unverändert, weil hier angenommen wurde, dass die Ausgangsgröße  $y(t)$  der Regelstrecke nicht direkt von der Stellgröße  $u(t)$  abhängt.

Wenn man einen dynamischen Regler (7.5) einsetzt, so wird wie beim linearen Regelkreis der Zustandsvektor der Regelstrecke um den des Reglers verlängert. Das Modell des Regelkreises hat dann die Form

$$\bar{\Sigma} : \begin{cases} \begin{pmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_r(t) \end{pmatrix} = \bar{\mathbf{g}} \left( \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{x}_r(t) \end{pmatrix}, w(t) \right), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \bar{h} \left( \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{x}_r(t) \end{pmatrix} \right). \end{cases}$$

Auf Modelle derselben Form kommt man, wenn man nichtlineare Regler einsetzt.

### Beispiel 7.2 Regelung eines Radioteleskops

Das in Aufgabe 3.3 behandelte Radioteleskop wird durch das nichtlineare Zustandsraummodell

$$\text{Radioteleskop} : \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} x_2(t) \\ \frac{g(m_G l_G - m_A l_A)}{J} \cos x_1(t) - \frac{k_B}{J} x_2(t) + \frac{1}{J} u(t) \end{pmatrix} \\ \begin{pmatrix} \phi(0) \\ \dot{\phi}(0) \end{pmatrix} = \begin{pmatrix} \phi_0 \\ \dot{\phi}_0 \end{pmatrix} \\ y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} \end{cases}$$

beschrieben. Mit dem P-Regler

$$\text{P-Regler: } u(t) = -k_P(y(t) - w(t))$$

erhält man für den Regelkreis das folgende Modell:

$$\text{geregeltes Radioteleskop:} \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} x_2(t) \\ \frac{g(m_G l_G - m_A l_A)}{J} \cos x_1(t) - \frac{k_P}{J} x_1(t) - \frac{k_B}{J} x_2(t) + \frac{k_P}{J} w(t) \end{pmatrix} \\ \begin{pmatrix} \phi(0) \\ \dot{\phi}(0) \end{pmatrix} = \begin{pmatrix} \phi_0 \\ \dot{\phi}_0 \end{pmatrix} \\ y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}. \end{cases}$$



Die zweite Zeile beschreibt die in Gl. (7.7) mit  $\bar{g}$  bezeichnete Funktion. Sie hängt außer von  $x_1(t)$  und  $x_2(t)$  auch von der Führungsgröße  $w(t)$  ab.  $\square$

## 7.3 Wichtige Eigenschaften von Regelkreisen

Dieser Abschnitt behandelt wichtige Eigenschaften von Regelkreisen, die einerseits die Notwendigkeit und die Zweckmäßigkeit von Steuerungen im geschlossenen Wirkungskreis deutlich werden lassen, andererseits offensichtliche Schranken für die erreichbare Regelgüte aufzeigen. Viele der folgenden Betrachtungen können mathematisch viel allgemeiner und tiefergründiger dargestellt werden als im Folgenden. Es kommt hier jedoch auf das intuitive Verständnis der behandelten Regelkreiseigenschaften an, das besser durch Beispiele und Erläuterungen als durch mathematische Beweise vermittelt wird.

### 7.3.1 Störkompensation und Sollwertfolge

Unter Störkompensation und Sollwertfolge werden die zwei Eigenschaften eines Regelkreises verstanden, die Störung  $d(t)$  asymptotisch zu unterdrücken und die Regelgröße asymptotisch der Führungsgröße  $w(t)$  anzupassen. Es gilt also

$$\lim_{t \rightarrow \infty} (w(t) - y(t)) = 0. \quad (7.8)$$

Der Regelkreis soll diese Eigenschaften auf Grund einer zweckmäßigen Wahl des Reglers erhalten. Dies ist, wie bereits erwähnt, nicht für beliebige Signale  $d(t)$  und  $w(t)$  möglich, sondern nur für vorgegebene Klassen von Störsignalen und Führungssignalen. In diesem Abschnitt werden Störkompensation und Sollwertfolge für impulsförmige und für sprungförmige Signale untersucht, weil diese beiden Klassen in der Praxis am häufigsten vorkommen.

**Impulsförmige Stör- und Führungsgrößen.** Impulsförmige Signale

$$\begin{aligned} d(t) &= \bar{d}\delta(t) \\ w(t) &= \bar{w}\delta(t) \end{aligned}$$

werden durch den Diracimpuls  $\delta(t)$  dargestellt, der ein unendlich kurzer, aber dafür unendlich hoher Impuls ist. Er ist die mathematische Idealisierung eines im Vergleich zu den Zeitkonstanten des Systems sehr kurzen Impulses mit der Impulsenergie (Impulsdauer · Impulsamplitude) von eins. Impulsförmige Signale verändern den aktuellen Zustand des Regelkreises, haben jedoch keine bleibende Wirkung auf den Regelkreis.

Betrachtet man das Regelkreismodell (7.6)

$$\bar{\Sigma} : \begin{cases} \dot{\bar{\mathbf{x}}}(t) = \bar{\mathbf{A}}\bar{\mathbf{x}}(t) + \bar{\mathbf{b}}_d d(t) + \bar{\mathbf{b}}_w w(t), & \bar{\mathbf{x}}(0) = \bar{\mathbf{x}}_0 \\ y(t) = \bar{\mathbf{c}}^T \bar{\mathbf{x}}(t) \end{cases}$$

bei  $w = 0$ , so wird offensichtlich, dass eine impulsförmige Störung dieselbe Wirkung hat wie eine Veränderung des Anfangszustands von  $\bar{x}_0$  auf  $\bar{x}_0 + \bar{b}_d \bar{d}$ , denn die Modelle

$$\bar{\Sigma} : \begin{cases} \dot{\bar{x}}(t) = \bar{A}\bar{x}(t) + \bar{b}_d \bar{d} \delta(t), & \bar{x}(0) = \bar{x}_0 \\ y(t) = \bar{c}^T \bar{x}(t) \end{cases}$$

und

$$\bar{\Sigma} : \begin{cases} \dot{\bar{x}}(t) = \bar{A}\bar{x}(t), & \bar{x}(0) = \bar{x}_0 + \bar{b}_d \bar{d} \\ y(t) = \bar{c}^T \bar{x}(t) \end{cases}$$

führen auf dieselbe Ausgangsgröße  $y(t)$  (vgl. Bewegungsgleichung (4.9) auf S. 127).

|| Impulsförmige Störungen zur Zeit  $t = 0$  können durch eine Veränderung des Anfangszustands dargestellt werden.

Sollwertfolge (7.8) tritt ein, wenn der Systemausgang für  $t \rightarrow \infty$  verschwindet. Dies geschieht immer dann, wenn der Regelkreis stabil ist. Für impulsförmige Signale  $d(t)$  und  $w(t)$  ist also mit der Entwurfsforderung (I) nach asymptotischer Stabilität des Regelkreises gleichzeitig die Forderung (II) nach Sollwertfolge und Störkompensation erfüllt. Es werden keine Forderungen an die Art des verwendeten Reglers gestellt.

**Sprungförmige Stör- und Führungsgrößen.** Im Folgenden wird der in Abb. 7.5 gezeigte Standardregelkreis betrachtet. Bei ihm sind die auf die Regelstrecke einwirkenden Störungen an den Ausgang der Regelstrecke transformiert. Dabei wird die Wirkung  $d_t(t)$ , die die Störung  $d(t)$  auf den Ausgang  $y(t)$  hat, durch das Störmodell dargestellt, das man aus dem Zustandsraummodell (7.2) für  $u = 0$  erhält:

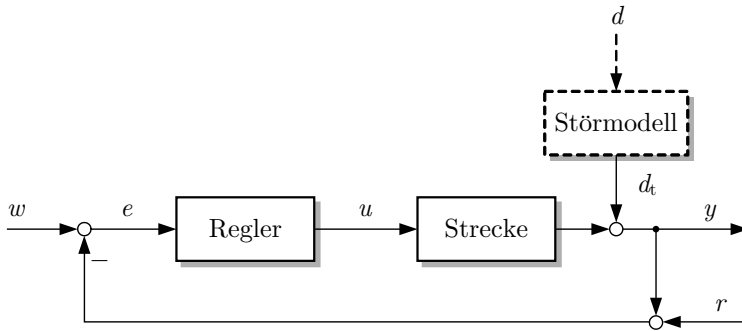
$$\text{Störmodell : } \begin{cases} \dot{x}_d(t) = A x_d(t) + g d(t), & x_d(0) = 0 \\ d_t(t) = c^T x_d(t). \end{cases}$$

In diesem Modell wurde der Zustandsvektor in  $x_d(t)$  umbenannt, um diesen Zustand von dem der Regelstrecke zu unterscheiden. Die Ausgangsgröße  $d_t(t)$  dieses Modells zeigt, wie die Regelgröße  $y(t)$  durch die Störung  $d(t)$  verändert wird. Da die Störung ein nicht beeinflussbares Signal ist, dessen Verlauf i. Allg. nicht bekannt ist, kann man an Stelle von  $d(t)$  mit der transformierten Störung  $d_t(t)$  arbeiten (und diese zur Vereinfachung wieder mit  $d(t)$  bezeichnen). Wichtig für die folgenden Betrachtungen ist also nur, dass es möglich ist, eine irgendwo an der Regelstrecke angreifende Störung an den Ausgang der Regelstrecke zu transformieren.

Die nachfolgenden Untersuchungen beziehen sich auf sprungförmige Führungssignale und sprungförmige Störsignale, die am Ausgang der Regelstrecke angreifen:

$$\begin{aligned} d(t) &= \bar{d} \sigma(t) \\ w(t) &= \bar{w} \sigma(t). \end{aligned}$$

Es wird jetzt angenommen, dass die Regelstrecke und der Regelkreis stabil sind, und erläutert, weshalb die Sollwertfolge nicht mit einem P-Regler



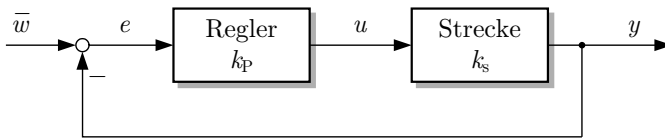
**Abb. 7.5:** Standardregelkreis mit an den Ausgang transformierter Störung

$$u(t) = -k_P(y(t) - w(t))$$

erzielt werden kann. Das statische Verhalten der Regelstrecke ist durch die Gleichung

$$y(\infty) = k_s u(\infty)$$

beschrieben, in der  $k_s$  den in Gl. (3.24) eingeführten statischen Verstärkungsfaktor bezeichnet.



**Abb. 7.6:** Statisches Verhalten des ungestörten Regelkreises

Betrachtet man zunächst eine sprungförmige Führungsgröße und setzt die Störung gleich null, so muss für die Stellgröße die Beziehung

$$u(\infty) = \frac{1}{k_s} \bar{w} \quad (7.9)$$

gelten, wenn die Regelgröße für  $t \rightarrow \infty$  den Sollwert  $\bar{w}$  annehmen soll. Unter dieser Bedingung gilt nämlich wie gefordert

$$y(\infty) = k_s u(\infty) = k_s \frac{1}{k_s} \bar{w} = \bar{w}.$$

Diese Stellgröße kann nicht mit einem proportionalen Regler erzeugt werden, denn wenn die Regelgröße dem Sollwert angepasst ist, gibt es keine Regelabweichung ( $e = 0$ ), so dass ein proportionaler Regler  $u(t) = k_P e(t)$  auch keine Stellgröße erzeugt. Diese Überlegung zeigt:

Im Standardregelkreis mit stabiler Regelstrecke und proportionalem Regler tritt bei sprungförmiger Führungsgröße oder sprungförmiger Störung eine bleibende Regelabweichung auf.

Wie groß diese bleibende Regelabweichung

$$e(\infty) = \bar{w} - y(\infty)$$

ist, kann man aus dem stationären Endwert der Regelgröße

$$y(\infty) = k_s u(\infty) = k_s k_P e(\infty)$$

berechnen. Das Produkt

$$k_0 = k_P k_s$$

der statischen Verstärkungen der Regelstrecke und des P-Reglers heißt *Kreisverstärkung*. Aus den angegebenen Gleichungen erhält man für die bleibende Regelabweichung die Beziehung

$$\boxed{\text{Bleibende Regelabweichung: } e(\infty) = \frac{1}{1 + k_0} \bar{w}} \quad (7.10)$$

und für den statischen Endwert der Regelgröße die Gleichung

$$y(\infty) = \frac{k_0}{1 + k_0} \bar{w}. \quad (7.11)$$

Die bleibende Regelabweichung ist also umso kleiner, je größer die Kreisverstärkung ist. Man kann also durch eine Erhöhung der Reglerverstärkung  $k_P$  die bleibende Regelabweichung verkleinern, muss dabei jedoch die bei diesen Betrachtungen vorausgesetzte Stabilität des Regelkreises gewährleisten.

### Beispiel 7.3 Bleibende Regelabweichung beim drehzahlgeregelten Gleichstrommotor

Im Beispiel 7.1 wurde das Zustandsraummodell

$$\text{geregelter Motor : } \begin{cases} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{k_P}{L} \\ 0 \end{pmatrix} \omega_{\text{Soll}}(t) \\ y(t) = (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} \end{cases}$$

für den Gleichstrommotor mit proportionalem Drehzahlregler aufgestellt. Für den konstanten Drehzahlsollwert  $\omega_{\text{Soll}}$  erhöht der Regler die Motordrehzahl bis zum statischen Endwert  $\omega(\infty)$ , der aus den Beziehungen

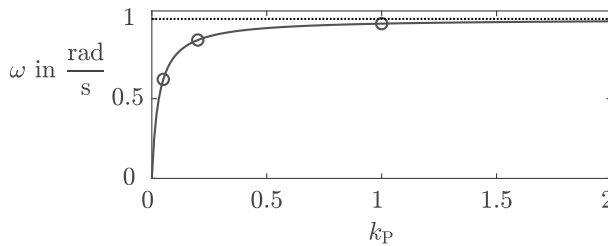
$$\begin{aligned} 0 &= \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix} \begin{pmatrix} i(\infty) \\ \omega(\infty) \end{pmatrix} + \begin{pmatrix} \frac{k_P}{L} \\ 0 \end{pmatrix} \omega_{\text{Soll}} \\ \begin{pmatrix} i(\infty) \\ \omega(\infty) \end{pmatrix} &= - \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix}^{-1} \begin{pmatrix} \frac{k_P}{L} \\ 0 \end{pmatrix} \omega_{\text{Soll}} \end{aligned}$$

und

$$\begin{aligned}\omega(\infty) &= -(0 \quad 1) \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} - \frac{k_P}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} \end{pmatrix}^{-1} \begin{pmatrix} \frac{k_P}{L} \\ 0 \end{pmatrix} \omega_{\text{Soll}} \\ &= \frac{k_P}{k_P + k_M + R \frac{k_R}{k_T}} \omega_{\text{Soll}}\end{aligned}$$

folgt (vgl. die Berechnung von Arbeitspunkten im Abschn. 3.3.6). Offenbar ist der Faktor vor  $\omega_{\text{Soll}}$  auf der rechten Seite nicht gleich eins, so dass  $\omega(\infty) \neq \omega_{\text{Soll}}$  gilt. Je größer die Reglerverstärkung  $k_P$  ist, umso kleiner ist die bleibende Regelabweichung  $\omega(\infty) - \omega_{\text{Soll}}$ .

Auf dieselbe Beziehung wäre man bei der Anwendung von Gl. (7.11) gekommen, wenn man vorher die statische Verstärkung  $k_s$  des Motors ausgerechnet hätte.



**Abb. 7.7:** Statischer Endwert  $\omega(\infty)$  der Drehzahl beim Sollwert  $\omega_{\text{Soll}} = 1$  für den proportional geregelten Gleichstrommotor mit dem Reglerparameter  $k_P$

Abbildung 7.7 zeigt den statischen Endwert  $\omega(\infty)$  der Drehzahl in Abhängigkeit von der Reglerverstärkung  $k_P$  für die im Beispiel 3.2 angegebenen Parameter, für die der Motor die statische Verstärkung  $k_s = 32,78$  hat. Wenn man die bleibende Regelabweichung möglichst klein machen will, muss man eine große Reglerverstärkung wählen ( $k_P > 0,7$ ).

Die Wahl von  $k_P$  beeinflusst aber auch das dynamische Übergangsverhalten. Abbildung 7.8 zeigt die Führungsübergangsfunktion, also die Reaktion des proportional geregelten Gleichstrommotors auf eine Erhöhung des Drehzahlsollwertes von null auf den Wert  $\omega_{\text{Soll}} = 1 \frac{\text{rad}}{\text{s}}$ . Für  $k_P = 0,05$  hat der Motor eine bleibende Regelabweichung von etwa  $0,4 \frac{\text{rad}}{\text{s}}$ . Dieser Reglerparameter entspricht in Abb. 7.7 der linken Markierung auf der Kurve. Vergrößert man die Reglerverstärkung auf 0,2 bzw. 1, so sinkt zwar die bleibende Regelabweichung, aber das dynamische Verhalten wird wesentlich schlechter. Proportionale Regelungen haben hier ihre Grenzen, denn die für die Drehzahlregelung gezeigten Eigenschaften findet man bei vielen Regelungsproblemen wieder. Deshalb reicht i. Allg. eine proportionale Regelung nicht aus und man muss zur Vermeidung von bleibenden Regelabweichungen zusätzliche Maßnahmen ergreifen.  $\square$

Im Folgenden werden zwei Möglichkeiten angegeben, um die bleibende Regelabweichung vollständig zu beseitigen. Erstens wird der Standardregelkreis durch ein Vorfilter erweitert, der die Führungsgröße so verändert, dass die bleibende Regelabweichung verschwindet. Zweitens wird ein Regler mit integralem Anteil verwendet, der die Regelabweichung sogar dann beseitigt, wenn das Verhalten der Regelstrecke von dem des Modells abweicht.

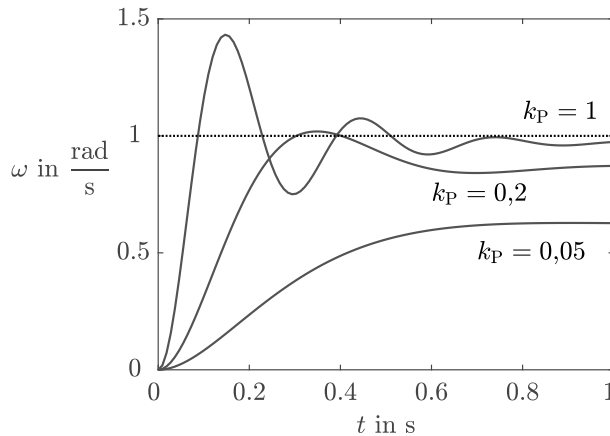


Abb. 7.8: Dynamisches Verhalten des Drehzahlregelkreises

**Regelkreis mit Vorfilter.** Der erste Lösungsweg besteht in einer Veränderung der Führungsgröße. Da bei einer proportionalen Regelung die Regelgröße an Stelle des gewünschten statischen Endwertes  $y(\infty) = \bar{w}$  nur den in Gl. (7.11) beschriebenen Endwert  $y(\infty) = \frac{k_0}{1+k_0} \bar{w}$  erreicht, liegt es nahe, die bleibende Regelabweichung dadurch zu beseitigen, dass man an Stelle der Führungsgröße  $w(t) = \bar{w}\sigma(t)$  mit der Führungsgröße

$$w(t) = \frac{1+k_0}{k_0} \bar{w} \sigma(t)$$

arbeitet. Obwohl man die Regelgröße nur um den Betrag  $\bar{w}$  verändern will, ändert man also die Führungsgröße um den Wert  $\frac{1+k_0}{k_0} \bar{w}$ . Dann gilt wie gewünscht  $y(\infty) = \bar{w}$ .

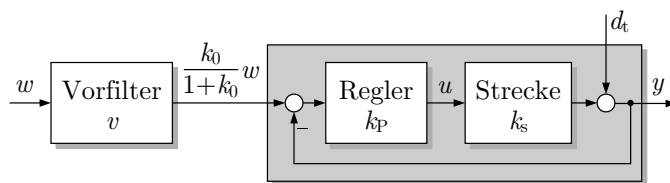


Abb. 7.9: Erweiterter Standardregelkreis mit Vorfilter

Da die Führungsgröße um den von  $\bar{w}$  unabhängigen Faktor  $\frac{1+k_0}{k_0}$  angehoben werden muss, verhindert man eine bleibende Regelabweichung durch die Erweiterung des Standardregelkreises um ein Vorfilter, das den Verstärkungsfaktor

Vorfilter:  $v = \frac{1+k_0}{k_0}$

(7.12)

besitzt (Abb. 7.9). Für beliebige Amplitude  $\bar{w}$  der Führungsgröße gleicht der proportionale Regler dann die Regelgröße dem gewünschten Wert  $\bar{w}$  an.

Man spricht bei dem neuen Block von einem *Vorfilter*, obwohl dieser Block nur eine statische Verstärkung darstellt, weil man einen derartigen Block auch mit dynamischen Eigenschaften ausstatten kann.  $v$  ist dann die statische Verstärkung des Filters.

Dieses Vorgehen hat jedoch zwei Nachteile. Erstens ist die auf diese Weise erreichte Eigenschaft der Sollwertfolge nicht robust gegenüber Modellunbestimmtheiten, denn wenn sich die statische Verstärkung  $k_s$  der Regelstrecke verändert, verändert sich auch die Kreisverstärkung  $k_0$ . Folglich erfüllt der Verstärkungsfaktor  $v$  des Vorfilters nicht mehr die in Gl. (7.12) beschriebene Forderung. Zweitens kann das Vorfilter eine bleibende Regelabweichung bei sprungförmigen Störungen nicht verhindern. Wie man aus Abb. 7.9 sieht, „merkt“ das Vorfilter gar nichts von der Störung und kann das Störverhalten des Regelkreises deshalb nicht beeinflussen. Es wirkt in einer offenen Wirkungskette, in der der Regelkreis die Funktion der Steuerstrecke hat, und kann keine Störungen ausgleichen.

**Regelkreis mit I-Regler.** Der zweite Lösungsweg geht von der bereits erwähnten Tatsache aus, dass bei proportional geregelten Systemen eine bleibende Regelabweichung auftritt, weil der P-Regler die in Gl. (7.9) beschriebene Stellgröße nicht bei verschwindender Regelabweichung erzeugen kann. Der Regler muss die Eigenschaft besitzen, seine Stellgröße solange zu *verändern*, bis die Regelabweichung verschwindet, und wenn die Regelabweichung gleich null ist, die Stellgröße unverändert auf dem erreichten Wert lassen. Diese Eigenschaft besitzen nur Systeme mit integralem Verhalten, was im Weiteren erläutert wird.

Ein Regler mit integralem Verhalten (I-Regler) ist durch das Zustandsraummodell

$$\text{I-Regler : } \begin{cases} \dot{x}_r(t) = y(t) - w(t), & x_r(0) = x_{r0} \\ u(t) = -k_I x_r(t) \end{cases} \quad (7.13)$$

beschrieben. Die Wirkung dieses Reglers erkennt man durch die folgende Überlegung. Wenn der Regelkreis stabil ist und man für konstante Werte der Führungs- und der Stellgröße das stationäre Verhalten betrachtet, so muss die Bedingung  $\dot{x} = 0$  nicht nur für den Zustand der Regelstrecke gelten, sondern auch für den Regler

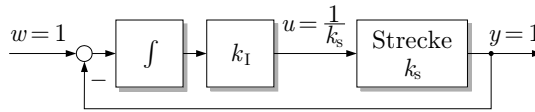
$$0 = \dot{x}_r = y(\infty) - \bar{w} \quad (7.14)$$

(Abb. 7.10). Daraus ergibt sich unmittelbar, dass die Regelgröße der Führungsgröße angeglichen ist:

$$y(\infty) = \bar{w}.$$

Der integrale Anteil (7.13) des Reglers erzwingt also in einem stabilen Regelkreis die Sollwertfolge. Da die Beziehung (7.14) auch bei Einwirkung von sprungförmigen Störungen gilt, erzeugt der integrale Anteil im Regler gleichzeitig eine Störunterdrückung.

Diese Eigenschaft ist übrigens unabhängig vom Reglerparameter  $k_I$ , solange der Regelkreis stabil ist. Der Reglerparameter muss deshalb zur Sicherung der Sollwertfolge und der Störunterdrückung lediglich so gewählt werden, dass die Stabilität des Regelkreises gewährleistet ist. Die Sollwertfolge ist robust gegenüber allen Parameteränderungen der Regelstrecke, für die der Regelkreis stabil bleibt.



**Abb. 7.10:** Statisches Verhalten von Regelkreisen mit I-Regler für  $\bar{w} = 1$

Eine Erweiterung der bisherigen Ergebnisse erhält man, wenn man sich die Tatsache zu Nutze macht, dass man in einer Reihenschaltung linearer Systeme die Reihenfolge der Übertragungsglieder vertauschen kann, ohne dass sich das Gesamtverhalten des Systems ändert. Aus der Reihenschaltung des Reglers mit integralem Anteil und der stabilen Regelstrecke mit dem statischen Übertragungsfaktor  $k_s$  entsteht dann eine Regelstrecke mit integralem Anteil und ein proportionaler Regler. Auch der aus diesen Elementen zusammengesetzte Regelkreis besitzt die Eigenschaften der Sollwertfolge und Störkompensation. Man kann diese Ergebnisse folgendermaßen zusammenfassen:

|| Bei sprungförmigen Führungs- und Störsignalen muss der Regler oder die Regelstrecke ein integrales Verhalten haben, damit keine bleibende Regelabweichung auftritt.

Dieses Ergebnis zeigt einen Grund dafür, warum dynamische Elemente im Regler notwendig sind.

**Beispiel 7.3 (Forts.)** *Bleibende Regelabweichung beim drehzahlgeregelten Gleichstrommotor*

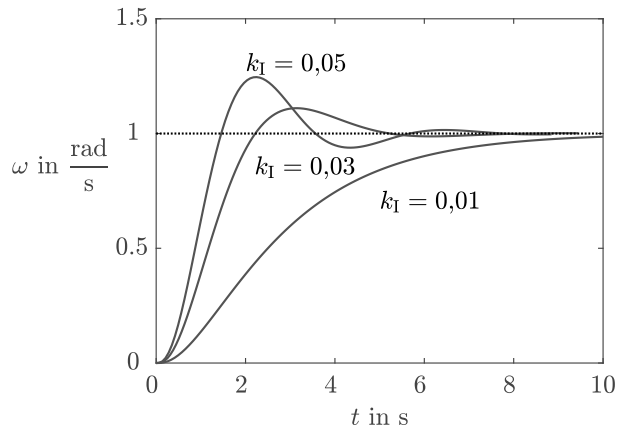
Wenn man für den Gleichstrommotor einen I-Regler (7.13) verwendet, so erhält man für den Regelkreis das folgende Modell:

$$\text{I-geregelter Motor : } \left\{ \begin{array}{l} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \\ x_r(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} & -\frac{k_I}{L} \\ \frac{k_T}{J} & -\frac{k_R}{J} & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \\ x_r(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \omega_{\text{Soll}}(t) \\ \begin{pmatrix} i(0) \\ \omega(0) \\ x_r(0) \end{pmatrix} = \begin{pmatrix} i_0 \\ \omega_0 \\ x_{r0} \end{pmatrix} \\ y(t) = (0 \quad 1 \quad 0) \begin{pmatrix} i(t) \\ \omega(t) \\ x_r(t) \end{pmatrix} \end{array} \right.$$

Abbildung 7.11 zeigt die Führungsübergangsfunktion für unterschiedliche Reglerparameter  $k_I$ . Es ist zu sehen, dass die Drehzahl für alle Parameterwerte den Sollwert erreicht. Der Reglerparameter muss deshalb nur noch in Bezug zu den dynamischen Eigenschaften des Regelkreises gewählt werden. Für alle Parameterwerte ist gesichert, dass es keine bleibende Regelabweichung gibt.

**Diskussion.** Neben dem Vergleich der bleibenden Regelabweichungen bei P- und I-Regelung zeigt das Beispiel auch, dass ein P-Regler schnell auf eine Regelabweichung reagiert und die Drehzahl schnell der Sollwertänderung anpasst, während bei der I-Regelung zwar keine bleibende Regelabweichung auftritt,





**Abb. 7.11:** Führungsübergangsfunktion des Drehzahlregelkreises mit I-Regler

das Übergangsverhalten aber wesentlich langsamer ist (vgl. Abb. 7.8 und 7.11). Diese Tatsache wird später genutzt, um beide Reglertypen zu kombinieren, so dass bei einem PI-Regler durch den P-Anteil der Regelkreis schnell gemacht und durch den I-Anteil eine bleibende Regelabweichung vermieden wird (vgl. Abschn. 7.4.1).  $\square$

#### **Aufgabe 7.1** Regelstrecken und Regler mit I-Anteil

Integrales Verhalten eines Systems bedeutet, dass sich die Ausgangsgröße  $y(t)$  näherungsweise proportional zum Integral der Eingangsgröße  $u(t)$  verhält:

$$y(t) \sim \int_0^t u(\tau) d\tau.$$

Diese Proportionalität gilt vor allem für große Zeit  $t$ , nachdem das Übergangsverhalten abgeklungen ist.

Wie kann man dieses Verhalten im Modell von Strecke und Regler erkennen bzw. durch welche Experimente mit der Regelstrecke kann man ermitteln, ob die Strecke integrales Verhalten hat? Wie sieht die Übergangsfunktion des I-Reglers (7.13) aus?  $\square$

#### **Aufgabe 7.2** Bestimmung der Reglerstruktur

Die folgenden Regelungen sollen Sollwertfolge bei sprungförmigen Führungssignalen besitzen. Entscheiden Sie aufgrund einer Analyse der Regelstrecke, ob dafür ein I-Anteil im Regler notwendig ist.

- Raumtemperaturregelung
- Abstandsregelung von Fahrzeugen
- Drehzahlregelung eines Gleichstrommotors
- Füllstandsregelung eines Reaktors.

$\square$

### 7.3.2 Erreichbare Regelgüte

Dieser Abschnitt untersucht die Frage, wie schnell ein Regler Störungen ausgleichen bzw. die Regelgröße der Führungsgröße nachführen kann. Dabei soll gezeigt werden, warum man dem Wunsch nach einer idealen Regelung, die die Regelgröße ohne Verzögerung der Führungsgröße angleicht ( $y(t) = w(t)$ ), nicht beliebig nahe kommen kann. Die folgenden Erläuterungen geben keine detaillierte mathematische Analyse des Regelkreisverhaltens, sondern weisen auf die aus praktischer Sicht wichtigsten Schwierigkeiten hin.

**Auswirkung von Messfehlern.** Systematische und zufällige Messfehler sind im Standardregelkreis durch die Größe  $r(t)$  repräsentiert (Abb. 7.4). Aus Sicht des Reglers wird deshalb nicht die wahre Regelgröße  $y(t)$ , sondern die um den Messfehler verfälschte Regelgröße

$$y_m(t) = y(t) + r(t)$$

betrachtet, so dass die Regelabweichung verschwindet, wenn  $y(t)$  von  $w(t)$  um den Betrag  $r(t)$  abweicht. Folglich gilt:

|| Die Regelung kann nur so genau sein, wie die Regelgröße gemessen wird.

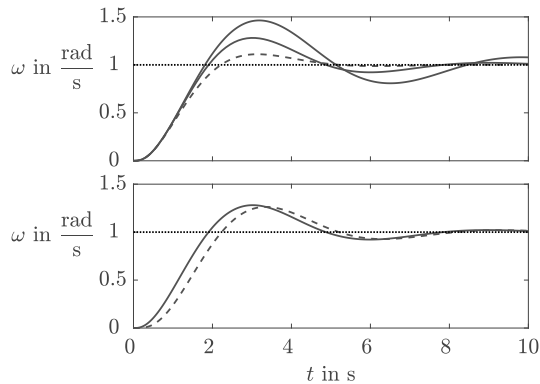
Man kann natürlich ein Filter verwenden, um stochastische Fehler im Messsignal zu unterdrücken. Dies bringt jedoch eine Verzögerung mit sich, mit der der Regler den Wert der Regelgröße erfährt, was zum zweiten Problem überleitet:

|| Der Regelkreis kann nicht schneller sein, als die Regelgröße gemessen wird.

#### Beispiel 7.4 Verschlechterung der Regelgüte durch Messverzögerungen

Die im Beispiel 7.3 beschriebene Drehzahlregelung wurde unter der Annahme parametrisiert, dass der Drehzahlsensor verzögerungslos arbeitet. Wie Abb. 7.12 (oben) zeigt, verschlechtert sich die Regelgüte wesentlich, wenn die Drehzahlmessung eine Zeitverzögerung in den Regelkreis einführt.

Die gestrichelte Kurve zeigt die Führungsübergangsfunktion des Regelkreises mit I-Regelung ohne Messverzögerungen ( $k_I = 0,03$ , vgl. Abb. 7.11). Die durchgezogenen Linien betreffen denselben Regelkreis, wenn das Messglied das Verhalten eines Systems erster Ordnung (4.37) mit der Zeitkonstante  $T = 0,3$  s bzw.  $T = 0,6$  s und der statischen Verstärkung  $k_{is} = 1$  hat. Die Messverzögerungen führen zu einer wesentlichen Verschlechterung der Regelgüte, weil die in der Abbildung dargestellte tatsächliche Drehzahl erst nach der genannten Zeitverzögerung in der Messgröße sichtbar wird. Wie im unteren Teil der Abbildung gezeigt wird, erfährt der Regler eine Änderungen der Drehzahl mit einer erheblichen Verzögerung. Die durchgezogene Kurve zeigt die tatsächliche Drehzahl (wie im oberen Abbildungsteil,  $T = 0,3$  s), die gestrichelte Kurve die Messgröße der Drehzahl, die der Regler für die Berechnung der Stellgröße verwendet. □



**Abb. 7.12:** Verschlechterung der Regelgüte durch Messverzögerungen

**Auswirkung von Stellgrößenbeschränkungen.** Wenn man die Regelgröße der Führungsgröße sehr schnell anpassen will, muss man mit einer großen Stellamplitude  $|u(t)|$  arbeiten.  $|u(t)|$  ist aber in allen Anwendungen beschränkt

$$|u(t)| \leq u_{\max},$$

wobei die Stellgrößenbeschränkung  $u_{\max}$  entweder durch das Stellglied vorgegeben ist oder aus den Gültigkeitsbedingungen für das beim Reglerentwurf verwendete Modell abgeleitet wird. Beispielsweise kann ein Ventilquerschnitt nur zwischen zwei Schranken verändert werden und der Arbeitspunkt des Schlupfregelkreises muss links vom Maximum der  $\mu(\lambda)$ -Kennlinie bleiben.

Die Stellgrößenbeschränkung führt in vielen Anwendungen auf die wichtigsten Beschränkungen für die Regelgüte. Man kann Regler finden, bei denen die mit dem linearen Modell berechnete Führungsübergangsfunktion eine (fast) beliebig kleine Überschwingzeit oder Beruhigungszeit hat. Wenn man das Regelkreisverhalten dann jedoch bezüglich der dabei auftretenden Stellgröße untersucht, stellt man fest, dass die Stellgrößenbeschränkungen weit überschritten sind.

**Verzögerungen im Regelkreis.** Der Regler wird nur dann wirksam, wenn an seinem Eingang eine Regelabweichung erkennbar wird. Auf Störungen kann der Regler deshalb erst reagieren, nachdem die Auswirkungen der Störungen in der Regelgröße sichtbar werden.

Der Regelkreis kann nicht wesentlich schneller gemacht werden, als es die maßgebenden Verzögerungen im Signalweg vom Stellglied durch die Regelstrecke zum Messglied zulassen.

Diese Überlegung zeigt, dass der Regler nicht nur auf den aktuellen Wert der Regelabweichung, sondern auch auf die zeitliche Ableitung der Regelabweichung reagieren muss, wenn der Regelkreis schneller reagieren soll. Dies ist ein wichtiges Argument für die Einführung des D-Anteils in den Regler (Abschn. 7.4.1).

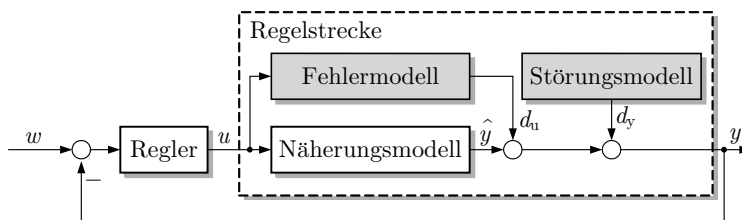
**Kompromiss zwischen Führungsverhalten und Störverhalten.** Beim Reglerentwurf muss häufig ein Kompromiss zwischen einem guten Verhalten des Regelkreises in Bezug zum Führungssignal und dem Verhalten bei Störungen gefunden werden. Dass dies notwendig ist, erkennt man im Standardregelkreis in Abb. 7.4 am einfachsten daran, dass man die Eingriffspunkte der Messstörung  $r(t)$  und der Führungsgröße  $w(t)$  vergleicht. Offenbar greifen Führungsgröße und Messrauschen an derselben Stelle ein und werden deshalb gleichberechtigt vom Regler behandelt. Folgt die Regelgröße einer Führungsgrößenänderung sehr schnell, so wird sie dies auch bezüglich des Messrauschens tun, was unerwünscht ist. Hier muss man offensichtlich einen Kompromiss suchen.

### 7.3.3 Robustheit

Unter Robustheit versteht man die Eigenschaft eines Regelkreises, vorgegebene Güteforderungen trotz Veränderungen in den dynamischen Eigenschaften der Regelstrecke zu erfüllen. Die Veränderungen in den Streckeneigenschaften entstehen entweder durch sich zeitlich verändernde Parameter oder durch Modellunbestimmtheiten. Im Folgenden werden Veränderungen der Regelstrecke zur Vereinfachung einheitlich als Modellunbestimmtheiten bezeichnet.

Die Robustheit ist eine Eigenschaft die sich auf zwei Vorgaben bezieht: die vom Regelkreis zu erfüllende Güteforderung und der Grad der Modellunbestimmtheit. Man bezeichnet einen Regler als *robusten Regler*, wenn der entstehende Regelkreis die Güteforderungen innerhalb der durch die Modellunbestimmtheiten bewirkten Parametertoleranzen erfüllt.

Dass Regelungen auf Grund der ihnen zu Grunde liegenden Informationsrückkopplung robuster sind als Steuerungen in der offenen Wirkungskette, wurde bereits im Beispiel 2.10 auf S. 73 durch einen Vergleich beider Steuerungsstrategien am invertierten Pendel demonstriert. Tatsächlich kann man bei jeder Regelung davon ausgehen, dass sie eine gewisse Robustheit besitzt. Für Anwendungen reicht diese qualitative Bewertung jedoch oft nicht aus. Man will wissen, ob gegebene Güteforderungen für spezifizierte Modellunbestimmtheiten tatsächlich erfüllt sind. Aus diesem Grunde ist ein umfangreiches Repertoire an Methoden zur Robustheitsanalyse entwickelt worden, auf die im Literaturverzeichnis verwiesen wird. Hier soll lediglich an Beispielen die Robustheit als grundlegende Regelkreiseigenschaft demonstriert werden.



**Abb. 7.13:** Regelkreis mit den unvollständig bekannten Komponenten „Fehlermodell“ und „Störgrößenmodell“

**Modellunbestimmtheiten und Störungen.** Modellunbestimmtheiten und Störungen haben dieselbe Wirkung: Der mit dem Modell berechnete Ausgang  $y(t)$  der Regelstrecke stimmt nicht mit dem überein, was man an der Regelstrecke messen kann. Zur Unterscheidung beider Signale wird die durch das Modell berechnete Ausgangsgröße im Folgenden mit  $\hat{y}(t)$  bezeichnet, während der reale Regelstreckenausgang  $y(t)$  heißt. Der durch Modellunsicherheiten und Störungen resultierende Fehler ist

$$\delta y(t) = y(t) - \hat{y}(t).$$

Trotz ähnlicher Wirkungen ist es für das Verhalten des Regelkreises sehr wichtig zu wissen, ob der Fehler  $\delta y(t)$  durch Modellunbestimmtheiten oder durch Störungen hervorgerufen wird. Dieser Unterschied ergibt sich aus der Tatsache, dass Fehler, die durch Modellunbestimmtheiten bedingt sind, von der Eingangsgröße  $u(t)$  abhängen, während die durch Störungen verursachten Fehler davon unabhängig sind. Das ist leicht einzusehen, wenn man Modell und Regelstrecke bei zwei unterschiedlichen Eingangsgrößen (also z. B. bei einer sprungförmigen und einer sinusförmigen Eingangsgröße) vergleicht. Wenn der Fehler  $\delta y(t) = d_u(t)$  durch Modellunbestimmtheiten hervorgerufen wird, ist sein Verlauf bei beiden Eingangsgrößen unterschiedlich. Beschreibt der Fehler die Wirkung einer Störung ( $\delta y(t) = d_y(t)$ ), so ist er unabhängig vom Eingang.

Modellunbestimmtheiten kann man deshalb durch ein Fehlermodell darstellen, das im Blockschaltbild in Abb. 7.13 die Eingangsgröße  $u(t)$  mit dem Modellfehler  $\delta y(t)$  verbindet. Demgegenüber wirkt eine Störung auf die Regelstrecke, ohne dass das Störgrößenmodell, das das Signal  $d_y(t)$  erzeugt, durch die Regelstrecke beeinflusst wird. Fehlermodell und Störungsmodell sind als graue Blöcke dargestellt, weil ihre Eigenschaften nicht genau bekannt sind (andernfalls könnte man den genauen Verlauf der Störung in die Berechnung des Regelstreckenausgangs einbeziehen bzw. die Modellunbestimmtheiten wären keine Unbestimmtheiten).

Im Regelkreis wirkt sich dieser Unterschied sehr stark aus. So entsteht innerhalb des Regelkreises durch das Fehlermodell eine neue Rückkopplungsschleife, die von  $y(t)$  über  $u(t)$  durch das Fehlermodell zurück zu  $y(t)$  führt. Demgegenüber tritt bei der Störung keine zusätzliche Rückkopplung auf. Das hat zur Folge, dass ein Regelkreis, der ohne Fehlermodell und Störungsmodell stabil ist, durch die Wirkung des Fehlermodells instabil gemacht werden kann. Störungen hingegen können die Stabilität des Kreises nicht gefährden.

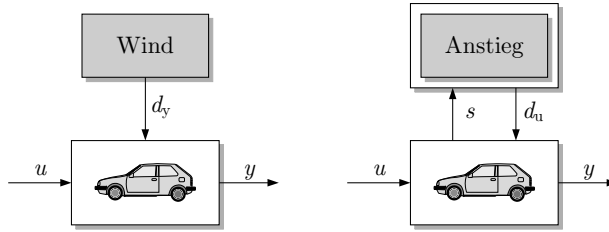
Bei der Lösung praktischer Aufgaben muss man deshalb sehr genau unterscheiden, ob die betrachteten Unbestimmtheiten die Übertragungseigenschaften vom Eingang  $u(t)$  auf den Ausgang  $y(t)$  der Regelstrecke betreffen (Fehlermodell) oder durch ein von außen auf die Strecke wirkendes Signal (Störgrößenmodell) darzustellen sind. Der Begriff der Robustheit bezieht sich auf Modellunbestimmtheiten und nicht auf Störungen (auch wenn in der Praxis manchmal von einer Robustheit gegenüber Störungen gesprochen wird).

#### **Beispiel 7.5** Modellunbestimmtheiten und Störungen bei der Modellierung der Fahrzeugbewegung

Abbildung 7.14 zeigt ein Fahrzeug unter der Wirkung von Wind bzw. auf einer ansteigenden Straße. In beiden Fällen wirkt ein Signal auf das Fahrzeug, das die Geschwindigkeit beeinflusst. Beim Wind stellt das Signal  $d_y(t)$  die durch den Wind auf das Auto ausgeübte Kraft dar, die in die im Beispiel 4.1 abgeleitete Zustandsgleichung (4.13) als zusätzliche Eingangsgröße eingeht:

$$\text{Fahrzeug : } \begin{cases} \dot{x}(t) = -\frac{c}{m}x(t) + \frac{1}{m}u(t) + \frac{1}{m}d_y(t), & x(0) = v_0 \\ y(t) = x(t). \end{cases}$$

Der Wind wirkt auf das Auto, aber das Auto beeinflusst den Wind nicht. Im Blockschaltbild geht deshalb nur ein Pfeil vom oberen zum unteren Block.



**Abb. 7.14:** Wirkung von Wind und Anstieg der Fahrbahn auf die Längsbewegung von Fahrzeugen

Anders sieht es bezüglich des Anstiegs der Fahrbahn aus, denn dessen Wirkung hängt von der Position des Fahrzeugs ab. Man kann die Wirkung des Anstiegs durch ein Signal  $d_u(t)$  beschreiben, dessen Verlauf von der Fahrzeugposition  $s(t)$  abhängt. Bei einem langsamer fahrenden Fahrzeug hat  $d_u(t)$  einen anderen Verlauf als bei einem schneller fahrenden. Für beide Fahrzeuge ist jedoch die Abhängigkeit zwischen  $d_u(t)$  und  $s(t)$  dieselbe.

Um die Wirkung des Anstiegs im Modell zu berücksichtigen, muss zunächst die Position des Fahrzeugs berechnet werden. Dafür gilt

$$\dot{s}(t) = x(t),$$

denn im Fahrzeugsmodell (4.13) beschreibt  $x(t)$  die Geschwindigkeit. Mit dieser Erweiterung erhält man ein Zustandsraummodell zweiter Ordnung

$$\begin{aligned} \begin{pmatrix} \dot{s}(t) \\ \dot{x}(t) \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ 0 & -\frac{c}{m} \end{pmatrix} \begin{pmatrix} s(t) \\ x(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u(t), & \begin{pmatrix} s(0) \\ x(0) \end{pmatrix} &= \begin{pmatrix} s_0 \\ v_0 \end{pmatrix} \\ y(t) &= (0 \quad 1) \begin{pmatrix} s(t) \\ x(t) \end{pmatrix} \\ s(t) &= (1 \quad 0) \begin{pmatrix} s(t) \\ x(t) \end{pmatrix}, \end{aligned}$$

bei dem die zweite Ausgangsgleichung den Weg  $s(t)$  wiedergibt. Ähnlich wie in Gl. (4.15) wird das Modell um den Term  $+gd_u(t)$  erweitert, um die Wirkung des Anstiegs zu erfassen

$$\begin{aligned} \begin{pmatrix} \dot{s}(t) \\ \dot{x}(t) \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ 0 & -\frac{c}{m} \end{pmatrix} \begin{pmatrix} s(t) \\ x(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ g \end{pmatrix} d_u(t) \\ y(t) &= (0 \quad 1) \begin{pmatrix} s(t) \\ x(t) \end{pmatrix} \\ s(t) &= (1 \quad 0) \begin{pmatrix} s(t) \\ x(t) \end{pmatrix}. \end{aligned}$$

Das Signal  $d_u(t)$  ist von der Fahrzeugposition  $s(t)$  abhängig, was man durch eine nichtlineare Funktion  $f_A$  zum Ausdruck bringen kann:

$$d_u(t) = f_A(s(t)).$$

In das Zustandsraummodell eingesetzt erhält man die Beziehung

$$\begin{pmatrix} \dot{s}(t) \\ \dot{x}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -\frac{c}{m} \end{pmatrix} \begin{pmatrix} s(t) \\ x(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ g \end{pmatrix} f_A(s(t))$$

und damit das nichtlineare Zustandsraummodell

$$\text{Fahrzeug am Anstieg : } \begin{cases} \begin{pmatrix} \dot{s}(t) \\ \dot{x}(t) \end{pmatrix} = \begin{pmatrix} x(t) \\ -\frac{c}{m}x(t) + \frac{1}{m}u(t) + gf_A(s(t)) \end{pmatrix} \\ y(t) = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} s(t) \\ x(t) \end{pmatrix}. \end{cases}$$

In dieses Modell geht mit der Funktion  $f_A$  der Anstieg der Straßenoberfläche ein. Der Anstieg verändert also das Modell des Fahrzeugs.

Dieses Beispiel zeigt, dass man sehr genau zwischen Modellfehlern und Störungen unterscheiden muss. Im Beispiel 4.1 wurde vereinfachend der Anstieg als zusätzliche Eingangsgröße des Fahrzeugs betrachtet. Dies geschah dort, um die Wirkung einer zusätzlichen Eingangsgröße zu demonstrieren. Bei der Berechnung des Verhaltens musste dort deshalb der Anstieg als eine Funktion der Zeit vorgegeben werden. Wenn man der Realität entsprechend den Anstieg als eine Funktion des Ortes vorgibt, so erhält man im Blockschaltbild einen mit dem Fahrzeug stark zusammenhängenden zusätzlichen Block (Abb. 7.14), der die Modellunbestimmtheiten repräsentiert und deshalb möglicherweise gar nicht durch ein quantitatives Modell exakt beschrieben werden kann. Wenn man beispielsweise nur weiß, dass irgendwann ein Anstieg mit einem maximalen Winkel zu überwinden sein wird, so kann die Funktion  $f_A(s)$  nicht genau, sondern nur durch obere und untere Schranken beschrieben werden.  $\square$

**Robustheitseigenschaften von Regelkreisen.** Die Untersuchungen der Störkompensation und Sollwertfolge im Abschn. 7.3.1 haben gezeigt, dass es für die Erfüllung dieser Güteforderung nicht auf genaue Parameterwerte, sondern auf die richtig gewählte Struktur des Reglers und die Stabilität des Regelkreises ankommt. Wenn der Regler integrales Verhalten hat und der Regelkreis stabil ist, so sind die Eigenschaften der Störkompensation und der Sollwertfolge gesichert. Vom Standpunkt der Robustheit betrachtet heißt dies: Die Parameter der Regelstrecke (und des Reglers) können sich sehr stark verändern – sie dürfen nur nicht zu einem instabilen Regelkreis führen. Wenn man einen Regler mit integralem Verhalten verwendet, muss man sich deshalb nicht mehr um die Sollwertfolge kümmern, sondern nur noch die Stabilität des Regelkreises gewährleisten. Dabei kann man beispielsweise mit dem Hurwitzkriterium Parameterbereiche ermitteln, für die der Regelkreis robust stabil ist, wie es in Aufgabe 6.8 auf S. 207 getan wurde.

Auch die Stabilität ist eine Eigenschaft, die i. Allg. nicht von exakten Parameterwerten abhängt, sondern sich für große Parameterbereiche einstellt. Dies wurde bereits im Abschn. 6.4.2 diskutiert. Das Ergebnis von Aufgabe 6.8 hat gezeigt, dass der Schlupfregelkreis für große Parameterbereiche funktioniert, so dass die betrachtete Regelung auf unterschiedlichem Fahrbahnbelag eingesetzt werden kann, ohne dass man die Reglerparameter an die aktuelle Fahrbahneigenschaften anpassen muss (was sicherlich technisch unmöglich wäre). Die betrachtete

Regelung funktioniert also aufgrund ihrer Robustheit unter allen praktisch auftretenden Randbedingungen.

Die für die Erfüllung einer Regelungsaufgabe grundlegenden Eigenschaften der Steuerbarkeit und Beobachtbarkeit sind, wie im Kap. 5 gezeigt wurde, maßgebend durch die Struktur des Systems bestimmt. Wenn die Struktur „stimmt“, führen nur ausgewählte Parameterkombinationen zum Verlust dieser Eigenschaften. Steuerbarkeit und Beobachtbarkeit sind also robust gegenüber Parameteränderungen.

**Aufgabe 7.3\*** Verhalten des Gleichstrommotors bei Störungen und Modellunbestimmtheiten

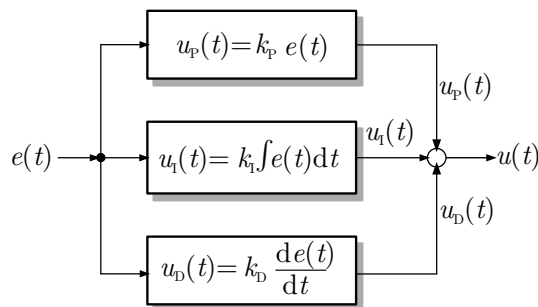
Betrachten Sie den Gleichstrommotor aus Beispiel 3.2 auf S. 90 unter der Wirkung eines zusätzlichen Bremsmomentes  $M_d(t)$  bzw. für eine Last, die ein mit der Drehzahl proportional steigendes Bremsmoment  $M_{Br}(t)$  auf den Motor ausübt. Welche Größen sind Störungen, welche wirken als Modellunbestimmtheiten? Zeichnen Sie für den Motor Blockschaltbilder, in denen entsprechend Abb. 7.13 das Fehlermodell bzw. das Störgrößenmodell zu erkennen ist. Wie verändern die betrachteten Größen das Modell des Motors? □

## 7.4 Reglertypen

### 7.4.1 PID-Regler

In diesem Abschnitt wird der PID-Regler eingeführt und es wird erläutert, welche Elemente dieses Reglers unter welchen Gegebenheiten eingesetzt werden müssen.

Im Prinzip kann man jedes beliebige dynamische System (7.5) als Regler verwenden. Häufig ist es aber gar nicht notwendig und nicht zweckmäßig, Regler sehr hoher dynamischer Ordnung einzusetzen. Man kann sogar sagen, dass sich das Know-how des Automatisierungstechnikers gerade darin ausdrückt, Regelungsaufgaben mit möglichst einfachen Reglern zu lösen. In den hier behandelten Beispielen reicht der in der Praxis häufig eingesetzte PID-Regler aus.



**Abb. 7.15:** Strukturbild eines PID-Reglers

Der PID-Regler besteht aus einer Parallelschaltung von drei Übertragungsgliedern, die drei unterschiedliche Regelungsstrategien realisieren (Abb. 7.15):



- Der **P-Anteil** (Proportionalanteil)

$$u_P(t) = k_P e(t)$$

realisiert die Strategie „Je größer die Regelabweichung ist, umso größer muss die Stellgröße sein.“

- Der **I-Anteil** (Integralanteil)

$$u_I(t) = k_I \int_0^t e(\tau) d\tau$$

arbeitet nach der Strategie „Solange eine Regelabweichung auftritt, muss die Stellgröße verändert werden“.

- Der **D-Anteil** (Differentialanteil)

$$u_D(t) = k_D \frac{de(t)}{dt}$$

berücksichtigt den Trend der Regelabweichung: „Je stärker sich die Regelabweichung verändert, umso stärker muss die Regelung reagieren“.

Die drei Anteile zusammen ergeben einen PID-Regler:

$$\begin{aligned} \text{PID-Regler: } u(t) &= k_P e(t) + k_I \int_0^t e(\tau) d\tau + k_D \frac{de(t)}{dt} \\ &= k_P \left( e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right). \end{aligned} \quad (7.15)$$

Die Gleichung gibt die beiden in der Literatur geläufigen Darstellungsweisen von PID-Reglern wieder, die sich mit den Beziehungen

$$k_I = \frac{k_P}{T_I} \quad \text{und} \quad k_D = k_P T_D$$

ineinander umrechnen lassen. In der zweiten Darstellung wird der Parameter  $T_I$  als *Nachstellzeit* und  $T_D$  als *Vorhaltezeit* bezeichnet. Lässt man einzelne Anteile des PID-Reglers weg, so erhält man einen PI-, PD-, P-, I- oder D-Regler.

Der P-Anteil arbeitet so, wie man eine Regelung ohne weitere Vorkenntnisse aufbauen würde, wobei die Stellgröße umso größer ist, je größer die Regelabweichung ist. Tatsächlich erfüllt eine solche Regelung in vielen Anwendungsfällen ihren Zweck, insbesondere wenn man impulsförmige Störungen hat und deshalb lediglich auf die Stabilität des Regelkreises achten muss.

Die Untersuchungen zur Sollwertfolge haben aber gezeigt, dass ein P-Regler bei sprungförmiger Führungsgröße oder Störgröße nicht ausreicht. Um eine bleibende Regelabweichung zu vermeiden, muss der Regler einen I-Anteil haben.

Andererseits reicht ein I-Regler i. Allg. nicht aus, um den Regelkreis schnell genug zu machen. Da die vom I-Anteil gelieferte Stellgröße proportional zum Integral der Regelabweichung

ist, reagiert sie auch auf große Regelabweichungen langsam, weil dafür erst das Integral über die Regelabweichung groß werden muss. Eine schnelle Reaktion erhält man nur durch einen P-Anteil, dessen Stellgröße proportional zur Regelabweichung selbst ist. P- und I-Anteil kombiniert ergibt einen PI-Regler, der sowohl schnell auf große Regelabweichungen reagiert als auch bleibende Regelabweichungen verhindert und der deshalb in der Praxis häufig eingesetzt wird.

Der D-Anteil ist zweckmäßig, wenn der Regler schnell auf die Regelabweichung reagieren soll. Er ist i. Allg. nicht zur Sicherung der Sollwertfolge erforderlich, sondern wird für die Erfüllung von Dynamikforderungen eingesetzt. Wie das folgende Beispiel zeigt, kann es aber auch für die Stabilisierung eines Systems notwendig sein, dass der Regler nicht nur auf die Regelabweichung, sondern auf die Geschwindigkeit reagiert, mit der sich die Regelabweichung verändert.

### Beispiel 7.6 Stabilisierung des invertierten Pendels

Das invertierte Pendel ist ein Beispiel für Systeme, für deren Stabilisierung eine P-Rückführung nicht ausreicht. Wie im Beispiel 2.10 auf S. 73 beschrieben wurde, besteht die Aufgabe der Regelung darin, das Pendel in der Gleichgewichtslage  $\bar{\phi} = 0$  zu stabilisieren. Aus Aufgabe 6.6 ist bekannt, dass diese Aufgabe nicht mit einer proportionalen Rückführung

$$u(t) = -k_P y(t)$$

gelöst werden kann, weil der Regelkreis für keinen Reglerparameter  $k_P$  asymptotisch stabil ist. Erst wenn man einen D-Anteil in den Regler hinzunimmt, findet man Reglerparameter, mit denen das geregelte Pendel stabil ist.

Durch den D-Anteil wird aus dem gemessenen Pendelwinkel  $\phi(t)$  die Winkelgeschwindigkeit  $\dot{\phi}(t)$  bestimmt, so dass man den PD-Regler in der Form

$$\text{PD-Regler: } u(t) = -k_P \phi(t) - k_D \dot{\phi}(t)$$

schreiben kann. Da der Sollwert für den Pendelwinkel gleich null ist, steht anstelle der Regelabweichung  $e(t)$  hier die Regelgröße  $\phi(t)$  im Reglergesetz.

Aus dem Zustandsraummodell (6.31) des Pendels

$$\text{Pendel: } \begin{cases} \begin{pmatrix} \dot{\phi}(t) \\ \ddot{\phi}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{3}{ml^2} \end{pmatrix} u(t) \\ y(t) = (1 \quad 0) \mathbf{x}(t) \end{cases}$$

erhält man mit dem PD-Regler das folgende Modell des geschlossenen Kreises:

$$\text{geregeltes Pendel: } \begin{pmatrix} \dot{\phi}(t) \\ \ddot{\phi}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} - k_P \frac{3}{ml^2} & -k_D \frac{3}{ml^2} \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}.$$

Wendet man das Hurwitzkriterium auf die charakteristische Gleichung

$$\lambda^2 + k_D \frac{3}{ml^2} \lambda - \frac{3g}{2l} + k_P \frac{3}{ml^2} = 0$$

an, so sieht man, dass die Ungleichungen

$$k_D > 0$$

$$k_P > \frac{mgl}{2}$$

notwendig und hinreichend für die Stabilität sind. Die erste Ungleichung zeigt, dass der D-Anteil im Regler notwendig für die Stabilisierung des Pendels ist.

**Diskussion.** Das invertierte Pendel zeigt auch, dass die Reglerparameter sehr unterschiedliche Maßeinheiten haben, auch wenn dies in den für die Berechnung verwendeten Zahlengleichungen nicht zum Ausdruck kommt, weil dort durch die Maßeinheiten dividiert wurde. Der Reglerparameter  $k_P$  des P-Anteils hat die Maßeinheit  $\frac{N}{\text{rad}}$ , wenn die Eingangsgröße  $u(t)$  in Newton und der Winkel  $\phi(t)$  im Bogenmaß gemessen wird. Das heißt, dass bei einem Winkel von 0,1 rad der P-Anteil des Reglers den Beitrag von  $k_P \cdot 0,1 \text{ N}$  zur Stellgröße  $u(t)$  beiträgt. Der Parameter  $k_D$  des D-Anteils hat die Maßeinheit  $\frac{Ns}{\text{rad}}$ , wenn die Winkelgeschwindigkeit  $\dot{\phi}(t)$  die Maßeinheit  $\frac{\text{rad}}{s}$  hat.  $\square$

### 7.4.2 Weitere Reglerkomponenten

Der Standardregelkreis aus Abb. 7.4 auf S. 214 enthält die für den Reglerentwurf und die Analyse der grundlegenden Eigenschaften von Regelkreisen notwendigen Komponenten. Er muss jedoch im praktischen Einsatz um weitere Komponenten erweitert werden, von denen in diesem Abschnitt einige beschrieben sind.

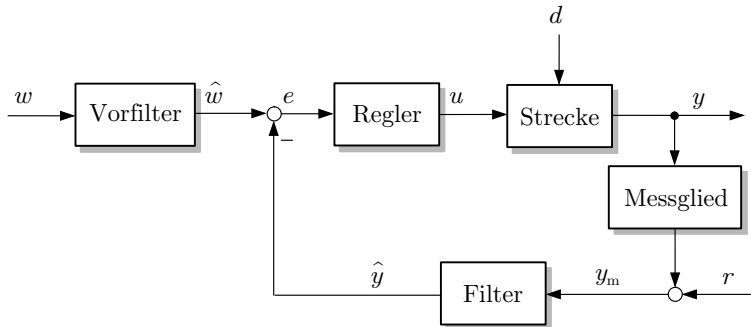


Abb. 7.16: Standardregelkreis mit zusätzlichen Komponenten

Der in Abb. 7.16 gezeigte Regelkreis hat gegenüber dem Standardregelkreis folgende zusätzliche Komponenten:

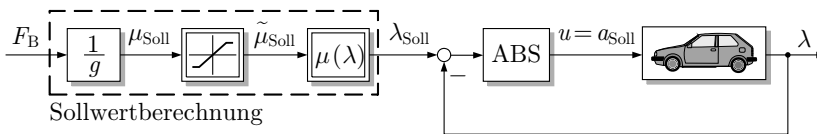
- **Messglied:** Das Messglied liefert den Messwert  $y_m(t)$  für die Regelgröße  $y(t)$ . Abgesehen von einer Verfälschung des Messwertes durch das Messrauschen  $r(t)$  oder durch systematische Messfehler spielt die zeitliche Verzögerung, mit der der Messwert der Regelgröße folgt, eine wichtige Rolle für das Regelkreisverhalten. Wenn die Regelgröße  $y(t)$  gar nicht messtechnisch erfasst werden kann, muss der Block „Messglied“ durch einen Block ersetzt werden, in dem die Regelgröße aus anderen messbaren Größen rekonstruiert wird.

- **Filter:** Um das Messrauschen zu beseitigen, wird die Messgröße  $y_m(t)$  gefiltert, wodurch der für die Regelung eingesetzte Wert  $\hat{y}(t)$  der Regelgröße entsteht. Dabei kann auf bekannte Filterverfahren zurückgegriffen werden. Für das Regelkreisverhalten ist neben der Filtergüte die Zeitverzögerung maßgebend, die durch das Filter in den Regelkreis hineingebracht wird.
- **Vorfilter:** Häufig ist es erforderlich, die gegebene Führungsgröße  $w(t)$  zu verformen und im Regelkreis mit der veränderten Führungsgröße  $\hat{w}(t)$  zu arbeiten. Beispielsweise müssen die Amplitude oder der Gradient der Führungsgröße so angepasst werden, dass der Regler technisch realisierbare bzw. technisch erlaubte Stellgrößen festlegt. Gegebenenfalls erscheint ein ähnlicher Block zwischen Regler und Strecke, um zu verhindern, dass die Stellgröße zu groß ist oder sich zu schnell ändert.

Das folgende Beispiel zeigt, dass sich diese zusätzlichen Blöcke in der Praxis (fast) zwangsläufig ergeben und dass dabei die für den Standardregelkreis untersuchten Eigenschaften von Regelungen im Wesentlichen bestehen bleiben.

#### Beispiel 7.7 Realisierung eines Antiblockiersystems

Wie in Abb. 6.13 auf S. 207 gezeigt ist, ist der Kern eines Antiblockiersystems ein Schlupfregler, der den zwischen Rad und Straße bestehenden Schlupf  $\lambda(t)$  einem Sollwert  $\lambda_{\text{Soll}}(t)$  anpasst, für den das Fahrzeug die gewünschte Bremsbeschleunigung erhält. In der technischen Realisierung muss dieser Regler um zusätzliche Komponenten erweitert werden.



**Abb. 7.17:** Erweiterung des ABS-Reglers um eine Komponente für die Sollwertberechnung

Die erste Komponente betrifft die Berechnung des Sollwertes  $\lambda_{\text{Soll}}(t)$  aus der vom Fahrer über das Bremspedal vorgegebenen Bremskraft  $F_B(t)$ . Aus  $F_B(t)$  wird zunächst die für diese Bremsbeschleunigung notwendige Kraftschlussbeanspruchung  $\mu_{\text{Soll}}(t) = \frac{F_B(t)}{mg}$  ermittelt. Liegt diese über dem Haftbeiwert  $\mu_H$  (vgl. Abb. 3.15 auf S. 108), so muss sie auf einen realisierbaren Wert  $\tilde{\mu}_{\text{Soll}}(t)$  verkleinert werden, was durch den linken nichtlinearen Block veranschaulicht wird (Abb. 7.19 (b)). Dann wird die  $\mu(\lambda)$ -Kennlinie verwendet, um den Schlupfsollwert  $\lambda_{\text{Soll}}(t)$  aus der Kraftschlussbeanspruchung  $\tilde{\mu}_{\text{Soll}}(t)$  zu ermitteln.

Für die Anwendung müssen jedoch noch weitere Umstände berücksichtigt werden. Da man den Schlupf  $\lambda(t)$  nicht messen kann, muss man ihn aus der Geschwindigkeit  $v(t)$  des Fahrzeugs und der Winkelgeschwindigkeit  $\omega(t)$  des Rades berechnen. Auch die Geschwindigkeitsmessung ist nicht ganz einfach, denn man kann  $v(t)$  nicht aus der Raddrehzahl entsprechend der Beziehung  $v(t) = r\omega(t)$  bestimmen, wie man es für die Anzeige im Auto tut, denn es kommt in dem hier betrachteten Regelkreis gerade auf den kleinen Unterschied zwischen  $v(t)$  und  $r\omega(t)$  an. Man verwendet häufig einen Schätzwert für  $v(t)$ , der aus den Winkelgeschwindigkeiten aller vier Räder berechnet wird, und ermittelt daraus den Schätzwert  $\hat{\lambda}$  für den Schlupf.

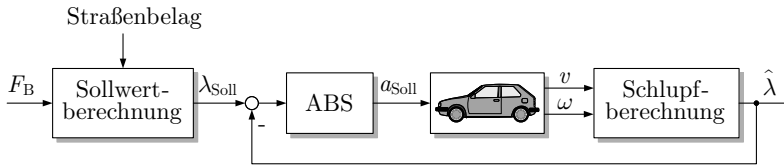


Abb. 7.18: Zweite Erweiterung des ABS-Reglers

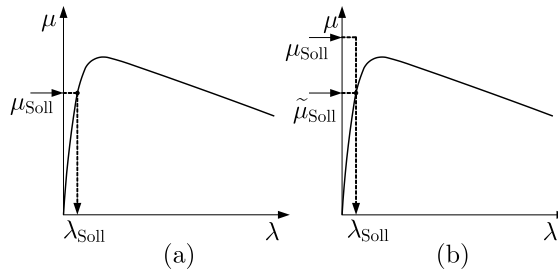


Abb. 7.19: Ermittlung des Sollwertes für den Schlupf: (a) bei ausreichendem Haftbeiwert, (b) durch Verkleinerung des Sollwertes für die Kraftschlussbeanspruchung

Für die Begrenzung der Kraftschlussbeanspruchung muss man den Haftbeiwert  $\mu_H$ , also die aktuellen Straßenverhältnisse kennen. Der Block „Sollwertberechnung“ hat deshalb eine zusätzliche Eingangsgröße, die diese Information liefert. Da man die Straßenverhältnisse nicht messen kann, müssen die entsprechenden Parameter der  $\mu(\lambda)$ -Kennlinie aus der Fahrzeugdynamik ermittelt werden (was in der Abbildung nicht dargestellt ist). Anhaltspunkte dafür liefert die Dynamik des ABS-Regelkreises, in die die zum aktuellen Straßenbelag gehörende  $\mu(\lambda)$ -Kennlinie eingeht (vgl. Abb. A.17 auf S. 587).

Wie das Blockschaltbild zeigt, ist beim ABS eine Steuerung im geschlossenen Wirkungskreis (Schlupfregelkreis) mit einer Steuerung in der offenen Wirkungskette (Sollwertberechnung) kombiniert. Störungen und Modellunbestimmtheiten, die sich z. B. aus unsicheren Straßenverhältnissen ergeben, können nur im Regelkreis ausgeglichen werden. Fehler in der Sollwertberechnung machen sich unmittelbar im Verhalten des Gesamtsystems bemerkbar.  $\square$

## Literaturhinweise

Von der vielfältigen Literatur zur robusten Regelung können die Monografien [3, 67, 78] als Einstiegslektüre empfohlen werden, in denen nicht nur die Theorie, sondern auch praktische Beispiele behandelt werden. Erweiterungen des Standardregelkreises durch eine Störgrößenaufschaltung, die Verwendung von Hilfsregelgrößen, Hilfsstellgrößen oder Kaskadenregelungen werden in [69] behandelt.



# 8

## Einstellregeln für PID-Regler

*Nach einer kurzen Erläuterung der Reglerentwurfsschritte behandelt dieses Kapitel Einstellregeln für PID-Regler, mit Hilfe derer man die Reglerparameter unter Verwendung von Experimenten mit dem Prozess festlegen kann.*

### 8.1 Entwurfsschritte

Um die im Abschn. 7.1 beschriebene Regelungsaufgabe zu lösen, müssen die folgenden drei Entscheidungen getroffen werden:

1. **Wahl der Regelkreisstruktur.** Es müssen die zu verwendende Stellgröße  $u(t)$  und die Regelgröße  $y(t)$  ausgewählt werden. Dieser Schritt wird häufig als Regelgrößen-Stellgrößen-Zuordnung bezeichnet. Meist ist die Regelgröße durch die Regelungsaufgabe vorgegeben. So ist bei einer Temperaturregelung in der Aufgabe festgelegt, welche Temperatur als Regelgröße fungieren soll. Bei der Wahl der Stellgröße gibt es jedoch i. Allg. mehr Freiheiten. Die Wahl der Stellgröße muss u. a. darauf Rücksicht nehmen, dass die Regelstrecke die im Kap. 5 angegebenen Steuerbarkeitsbedingungen erfüllt.
2. **Wahl der Reglerstruktur.** Es ist zu entscheiden, welcher Regler eingesetzt werden muss. Richtlinien hierfür sind im Kap. 7 behandelt worden. Im Folgenden werden meist PID-Regler verwendet, deren I-Anteil die Sollwertfolge und die Störkompensation bei sprungförmigen Führungs- und Störgrößen sichert.

3. **Wahl der Reglerparameter.** Die Reglerparameter sind so festzulegen, dass die an den Regelkreis gestellten Forderungen bezüglich des dynamischen Verhaltens erfüllt werden (vgl. S. 212). Mit dieser Aufgabe befassen sich die folgenden Abschnitte dieses Kapitels.

Das Ziel dieser Schritte ist es, ein Reglergesetz  $\Sigma_R$  aufzustellen, für das die im Abschn. 7.1 angegebenen Güteforderungen an den Regelkreis  $\bar{\Sigma}$  erfüllt werden. Die genannten Schritte werden i. Allg. in der angegebenen Reihenfolge ausgeführt. Typisch für den Reglerentwurf ist jedoch, dass die Entscheidungen mehrfach revidiert werden, bis ein befriedigender Regler gefunden ist.

Alle weiteren Betrachtungen dieses Kapitels gehen davon aus, dass die Regelkreisstruktur bereits festgelegt ist, also bekannt ist, welches Signal die Stellgröße  $u(t)$  und welches die Regelgröße  $y(t)$  darstellt. Festzulegen sind die Reglerstruktur und die Reglerparameter. Dabei sind die folgenden Schritte zu durchlaufen, die man zum Reglerentwurf (im weiteren Sinne) zusammenfasst:

---

### Entwurfsverfahren 8.1 *Reglerentwurf für kontinuierliche Regelstrecken*

---

**Gegeben:** Regelkreisstruktur  
Güteforderungen

1. Stellen Sie das Modell der Regelstrecke  $\Sigma_S$  auf.
2. Wählen Sie die Reglerstruktur.
3. Legen Sie die Reglerparameter fest.
4. Analysieren Sie das Verhalten des Regelkreises  $\bar{\Sigma}$ .
5. Bewerten Sie die Güte des Regelkreises: Sind die Güteforderungen erfüllt, ist das Entwurfsproblem gelöst. Andernfalls müssen Sie den Reglerentwurf ab Schritt 2 oder 3 wiederholen.

**Ergebnis:** Reglergesetz  $\Sigma_R$ .

---

Der Schritt der Festlegung der Reglerparameter wird auch als Reglerentwurf (im engeren Sinne) bezeichnet.

Die angegebenen Schritte werden i. Allg. mehrfach wiederholt, insbesondere der Reglerentwurf und die Simulationsuntersuchungen des Regelkreises. Da man beim Entwurf stets mit einem möglichst einfachen Reglergesetz beginnt, muss man nach einigen erfolglosen Entwurfschritten gegebenenfalls die Reglerstruktur allgemeiner ansetzen, wobei beispielsweise ein PI-Regler zu einem PID-Regler erweitert wird, um das dynamische Verhalten des Kreises schneller zu machen.

Da die Behandlung von Reglerentwurfsverfahren den Rahmen dieses Lehrbuchs sprengen würde, werden im Folgenden nur wesentlich einfachere Wege zur Ermittlung der Reglerparameter behandelt, bei denen die Reglerparameter mit Hilfe von Experimenten ermittelt werden. Man spricht dann nicht von einem Reglerentwurf, sondern von einer Reglereinstellung.



## 8.2 Einstellregeln von ZIEGLER und NICHOLS

Viele praktische Regelungsaufgaben sind dadurch gekennzeichnet, dass kein Modell der Regelstrecke verfügbar ist und ein Regler gesucht wird, der relativ schwache Güteforderungen zu erfüllen hat. Man fordert natürlich die Stabilität des Regelkreises und die Sollwertfolge für sprungförmige oder impulsförmige Führungs- und Störsignale, stellt aber keine weitergehenden Forderungen an das Übergangsverhalten des Regelkreises. Bei diesen relativ einfachen Regelungsaufgaben wird häufig so vorgegangen, dass der Regler an die Regelstrecke angeschlossen und mit Hilfe von Experimenten nach günstigen Reglerparametern gesucht wird. Die Experimente dienen einerseits zum Kennenlernen wichtiger dynamischer Eigenschaften der Regelstrecke und andererseits zum Beurteilen des Regelkreisverhaltens mit den gewählten Reglerparametern.

Da nur schwache Güteforderungen zu erfüllen sind, kann der aufwändige Weg des Entwurfsverfahrens 8.1 abgekürzt werden. Voraussetzung für dieses Vorgehen ist, dass die Regelstrecke stabil ist und dass mit der Regelstrecke experimentiert werden kann. Weiterhin muss bereits entschieden sein, welche Regelkreisstruktur verwendet wird und welcher Regler eingesetzt werden soll. Die Einstellregeln dienen lediglich zur Wahl der Reglerparameter.

Im Folgenden werden Regelkreise mit PID-Reglern (7.15)

$$\text{PID-Regler: } u(t) = k_P \left( e(t) + \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{de(t)}{dt} \right)$$

betrachtet, wobei man wahlweise einzelne Komponenten des PID-Reglers weglassen und damit zum P- bzw. PI-Regler übergehen kann (Abb. 8.1). Das hier beschriebene Vorgehen ist für Regelstrecken anwendbar, deren Übergangsfunktion näherungsweise den in Abb. 4.22 gezeigten Verlauf hat. Das heißt, bei einer Änderung der Eingangsgröße soll sich die Ausgangsgröße der Regelstrecke ohne oder mit geringem Überspringen dem statischen Endwert nähern. Durch ein Experiment werden Kenngrößen bestimmt, die die wichtigsten Regelstreckeneigenschaften beschreiben. Dann werden die Reglerparameter mit Hilfe empirisch ermittelter Einstellregeln festgelegt.

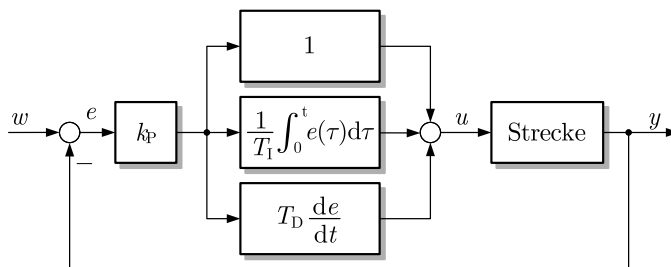
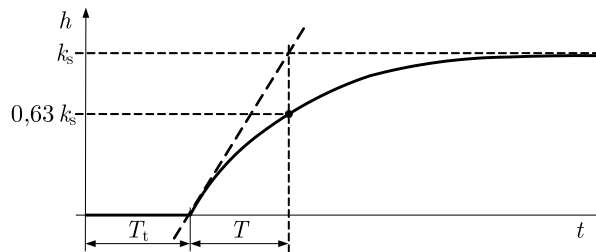


Abb. 8.1: Regelkreis mit PID-Regler

Zwei der bekanntesten Einstellregeln sind die von ZIEGLER und NICHOLS<sup>1</sup> aufgestellten, die in der Tabelle 8.1 zusammengefasst sind und jetzt näher betrachtet werden.

<sup>1</sup> NATHANIEL B. NICHOLS (1914 – 1997), amerikanischer Regelungstechniker



**Abb. 8.2:** Übergangsfunktion der Regelstrecke

**Erste Einstellregel.** Bei der ersten Einstellregel wird ein Experiment mit der Regelstrecke gemacht, bei dem die Strecke aus der Ruhelage  $x_0 = 0$  durch eine sprungförmige Eingangsgröße  $u(t) = \sigma(t)$  erregt wird. Dann werden die drei in Abb. 8.2 gekennzeichneten Parameter  $k_s$ ,  $T$  und  $T_t$  so bestimmt, dass die Messkurve durch die in der Abbildung gezeigte Kurve gut approximiert wird. Die drei Parameter charakterisieren die wichtigsten Eigenschaften der Regelstrecke:

- $T_t$  ist eine *Totzeit*, die vergeht, bevor sich die Eingangsgröße in einer deutlichen Veränderung der Ausgangsgröße bemerkbar macht.
- $T$  ist die Zeitkonstante der Regelstrecke, wenn man das dynamische Übergangsverhalten durch ein System erster Ordnung approximiert (vgl. Abschn. 4.4).
- $k_s$  ist die statische Verstärkung der Regelstrecke.

Da technische Systeme typischerweise eine Verzögerung höherer Ordnung aufweisen, entsteht bei der in der Abbildung gezeigten Approximation eine positive Totzeit  $T_t > 0$ .

Aus diesen Parametern der Regelstrecke werden die Reglerparameter nach folgendem Verfahren bestimmt:

---

### Einstellverfahren 8.1 *Reglereinstellung nach ZIEGLER und NICHOLS (I)*

---

**Gegeben:** Stabile Regelstrecke mit näherungsweise aperiodischem Übergangsverhalten, Güteforderungen bezüglich der Stabilität und der Sollwertfolge bei sprungförmigen Führungs- und Störsignalen

1. Bestimmen Sie durch ein Experiment die Übergangsfunktion der Regelstrecke.
2. Bestimmen Sie aus der Übergangsfunktion die statische Verstärkung  $k_s$ , die Totzeit  $T_t$  und die Zeitkonstante  $T$  entsprechend Abb. 8.2.
3. Legen Sie die Reglerparameter entsprechend Tabelle 8.1 (oben) fest.

**Ergebnis:** Reglerparameter für P-, PI- bzw. PID-Regler.

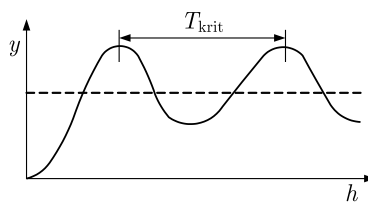
---

Die in der Tabelle angegebenen Einstellvorschriften wurden empirisch bzw. anhand von Simulationsuntersuchungen so festgelegt, dass der geschlossene Kreis mit mäßigem bis starkem Überschwingen den Sollwert erreicht.

**Tabelle 8.1.** Reglereinstellung nach ZIEGLER und NICHOLS

Voraussetzung	Regler	Reglerparameter
Die Übergangsfunktion der Regelstrecke ist durch $T_t$ , $T$ und $k_s$ beschrieben.	P	$k_P = \frac{1}{k_s} \frac{T}{T_t}$
	PI	$k_P = \frac{0,9}{k_s} \frac{T}{T_t}$ , $T_I = 3,33 T_t$
	PID	$k_P = \frac{1,2}{k_s} \frac{T}{T_t}$ , $T_I = 2 T_t$ , $T_D = 0,5 T_t$
Die kritische Verstärkung $k_{krit}$ und die Periodendauer $T_{krit}$ sind bekannt.	P	$k_P = 0,5 k_{krit}$
	PI	$k_P = 0,45 k_{krit}$ , $T_I = 0,85 T_{krit}$
	PID	$k_P = 0,6 k_{krit}$ , $T_I = 0,5 T_{krit}$ , $T_D = 0,12 T_{krit}$

**Zweite Einstellregel.** Ein anderer Weg der Reglereinstellung beruht darauf, dass zunächst die Stabilitätsgrenze des proportional geregelten Systems experimentell ermittelt wird. Diese Methode ist nur bei Prozessen einsetzbar, die bis an die Stabilitätsgrenze belastet werden können.



**Abb. 8.3:** Regelkreis mit schwingender Regelgröße

Man schließt den Regelkreis mit einem P-Regler und erhöht die Reglerverstärkung  $k_P$  so lange, bis der Regelkreis zu schwingen beginnt. Wenn man den Regelkreis durch eine Erhöhung des Sollwertes  $w(t)$  angeregt hat, erhält man den in Abb. 8.3 gezeigten Verlauf der Führungsübergangsfunktion. Die Reglerverstärkung, bei der das Schwingen eintritt, wird mit  $k_{krit}$  bezeichnet. Außerdem bestimmt man die Periodendauer  $T_{krit}$  der Schwingung, die man zwischen zwei Maximalwerten der Regelgröße ablesen kann.

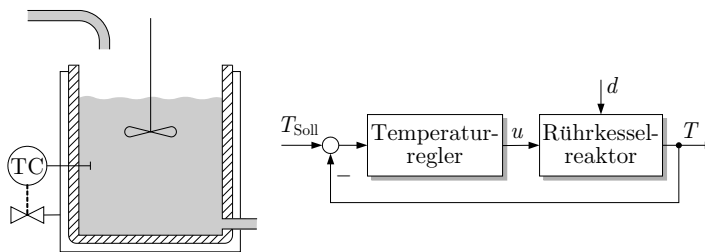
### Einstellverfahren 8.2 Reglereinstellung nach ZIEGLER und NICHOLS (II)

**Gegeben:** Stabile Regelstrecke, die zeitweise im grenzstabilen Bereich betrieben werden kann.

Güteforderungen bezüglich der Stabilität und der Sollwertfolge bei sprungförmigen Führungs- und Störsignalen

1. Schließen Sie den Regelkreis mit einem P-Regler.
2. Erhöhen Sie die Reglerverstärkung  $k_P$  solange, bis der geschlossene Kreis nach einer Sollwertänderung  $w(t) = \sigma(t)$  eine Dauerschwingung ausführt. Die dabei eingestellte Reglerverstärkung heißt  $k_{krit}$ , die Periodendauer der Schwingung  $T_{krit}$ .
3. Legen Sie die Reglerparameter entsprechend Tabelle 8.1 (unten) fest.

**Ergebnis:** Reglerparameter für P-, PI- und PID-Regler.

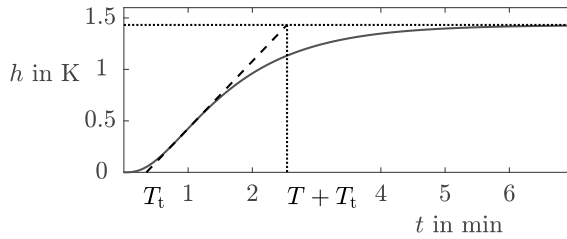
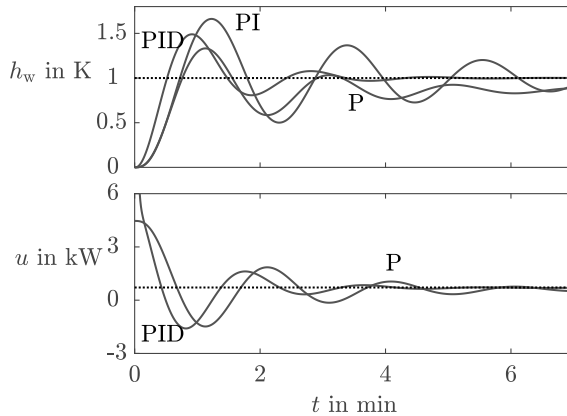


**Abb. 8.4:** Temperaturregelung eines kontinuierlich durchströmten Rührkesselreaktors mit einer Heizung in der Reaktorwand

#### Beispiel 8.1 Einstellung des Temperaturregelkreises eines Rührkesselreaktors

Die Einstellregeln sollen am Beispiel einer Temperaturregelung demonstriert werden. Die Regelstrecke besteht aus einem kontinuierlich durchströmten Reaktor, dessen Inhalt als homogen durchmischt angenommen wird. Regelgröße ist die Temperatur der aus dem Reaktor abgezogenen Flüssigkeit. Durch Veränderung der Heizleistung soll der Einfluss der Störung  $d(t)$ , die sich in Temperaturänderungen des Zulaufs bemerkbar macht, ausgeglichen werden.

Die Übergangsfunktion  $h(t)$  der Regelstrecke ist in Abb. 8.5 gezeigt. Sie beschreibt, wie sich eine sprungförmige Änderung  $u(t) = \sigma(t)$  der Heizleistung in der gemessenen Reaktortemperatur  $y(t)$  niederschlägt. Da die elektrische Heizung zunächst den Reaktormantel erwärmen muss, bevor sich die Erhöhung der elektrischen Wärmeleistung in einer Temperaturerhöhung bemerkbar machen kann, tritt die Wirkung mit einer erheblichen Verzögerung ein. Die Übergangsfunktion kann näherungsweise durch die Totzeit  $T_t = 0,4 \text{ min}$  und die Verzögerungszeit  $T = 2,5 \text{ min}$  beschrieben werden. Die statische Verstärkung hat den Wert  $k_s = 1,4 \frac{\text{K}}{\text{W}}$ .

**Abb. 8.5:** Übergangsfunktion des Rührkesselreaktors**Abb. 8.6:** Führungsübergangsfunktion des Temperaturregelkreises:  
Regelgröße  $y(t)$  (oben) und Stellgröße  $u(t)$  (unten)

Die Reglerparameter werden entsprechend Tabelle 8.1 (oben) eingestellt, wobei man folgende Werte erhält:

P-Regler:  $k_P = 4,46$

PI-Regler:  $k_P = 4,02$ ,  $T_I = 1,33$

PID-Regler:  $k_P = 5,38$ ,  $T_I = 0,80$ ,  $T_D = 0,20$ .

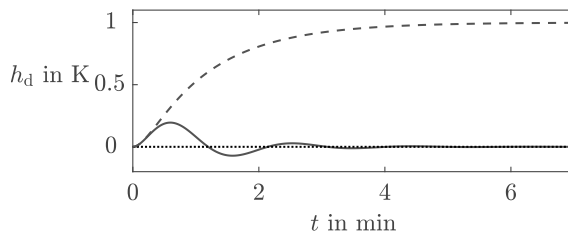
Die Führungsübergangsfunktionen  $h_w(t)$  für diese drei Regler sind in Abb. 8.6 (oben) dargestellt. Wie man sieht, reagieren alle drei Regelkreise auf die Sollwertänderung schneller als die Regelstrecke, wobei allerdings der P- und der PI-Regelkreis auf ein großes Überschwingen der Regelgröße führen und relativ schwach gedämpft sind. Dieser Verlauf der Regelgröße ist typisch für Regelstrecken mit großem Verhältnis von Zeitkonstante  $T$  und Totzeit  $T_t$ , denn für diese Regelstrecken führen die Einstellregeln auf eine große P-Verstärkung  $k_P$  und damit auf relativ schwach gedämpfte Regelkreise. Beim PI- und PID-Regler tritt keine bleibende Regelabweichung auf, beim P-Regler gilt entsprechend Gl. (7.10)

$$e(\infty) = \frac{1}{1 + k_0} \bar{w} = \frac{1}{1 + k_s k_P} = \frac{1}{1 + 1,4 \cdot 4,46} = 0,14.$$

Für den PI- und den PID-Regler hat die Stellgröße den Endwert

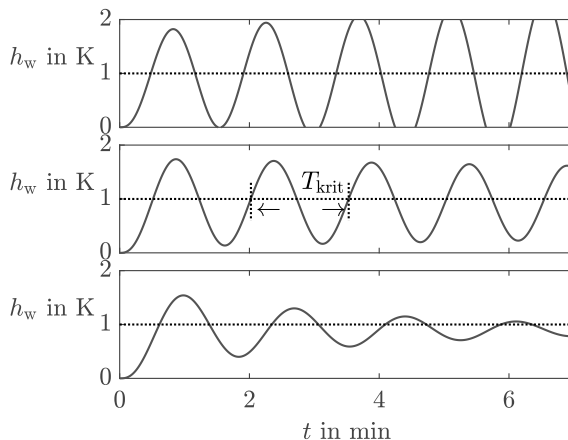
$$u(\infty) = \frac{1}{k_s} \bar{w} = \frac{1}{1,4} = 0,71,$$

der in Abb. 8.6 (unten) durch die gepunktete Linie gekennzeichnet ist. Für den P-Regler liegt der Endwert geringfügig darunter, was die bleibende Regelabweichung hervorruft. Aufgrund des D-Anteils ist die Stellgröße des PID-Reglers für kleine Zeit  $t$  sehr groß, während beim P-Regler die Kurve bei  $u(0) = k_P = 4,46$  beginnt.



**Abb. 8.7:** Störübergangsfunktion des Temperaturregelkreises mit PID-Regler im Vergleich zum Verhalten der Regelstrecke

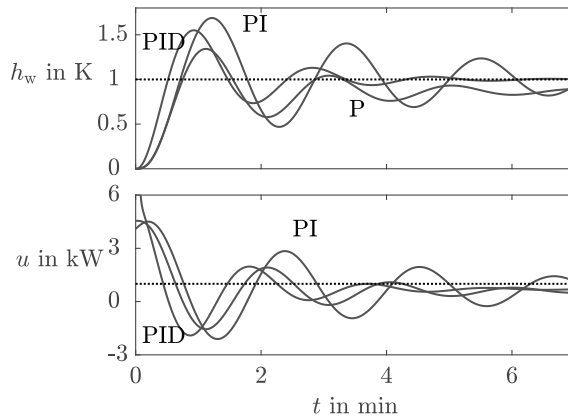
Abbildung 8.7 zeigt die Störübergangsfunktion  $h_d(t)$ , also die Reaktion des Regelkreises auf eine sprungförmige Erhöhung der Temperatur des Zulaufs zum Reaktor um 1 K. Wie man sieht, führt eine Temperaturerhöhung von einem Kelvin im Regelkreis auf eine maximale Anhebung der Reaktortemperatur um 0,2 K (durchgezogene Linie), während sie sich im unregulierten Reaktor in einer bleibenden Erhöhung von 1 K bemerkbar machen würde (gestrichelte Linie). Die schnelle Reaktion der Regelung führt dazu, dass die Wirkung der Störung schnell abgebaut wird.



**Abb. 8.8:** Verhalten des P-geregelten Reaktors bei unterschiedlichen Reglerverstärkungen  $k_P$

Für die Anwendung der zweiten Einstellregel wird der Regelkreis zunächst mit einem P-Regler geschlossen und die Reglerverstärkung  $k_P$  solange erhöht, bis der Regelkreis mit konstanter Amplitude schwingt. Abbildung 8.8 zeigt die Führungsübergangsfunktion des Regelkreises für die drei Reglerverstärkungen

$$k_P = 10,5 \text{ (oben)}, \quad k_P = 9,1 \text{ (Mitte)}, \quad k_P = 6,5 \text{ (unten)}.$$



**Abb. 8.9:** Führungsübergangsfunktion des Temperaturregelkreises mit der zweiten Reglereinstellung

Für die erste Reglerverstärkung ist der Regelkreis instabil, denn er schwingt mit steigender Amplitude. Für die dritte Reglerverstärkung nimmt die Schwingungsamplitude ab. Die mittlere Abbildung zeigt das Verhalten bei der kritischen Reglerverstärkung  $k_{\text{krit}} = 9,1$ . Für die Anwendung der Einstellregel wird aus dieser Abbildung die Periodendauer  $T_{\text{krit}} = 1,55$  abgelesen. Aus dem unteren Teil der Tabelle 8.1 erhält man dann folgende Reglerparameter:

P-Regler:  $k_P = 4,55$

PI-Regler:  $k_P = 4,10$ ,  $T_I = 1,32$

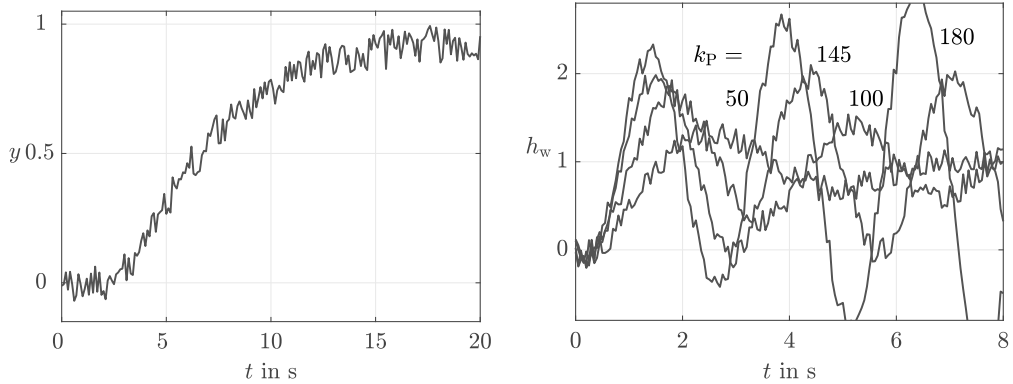
PID-Regler:  $k_P = 5,46$ ,  $T_I = 0,78$ ,  $T_D = 0,19$ .

Diese Parameterwerte unterscheiden sich nur geringfügig von den mit der ersten Methode erhaltenen. Sie führen deshalb auch zu sehr ähnlichen Führungsübergangsfunktionen, wie in Abb. 8.9 zu sehen ist.

**Diskussion.** Mit beiden Einstellregeln werden die Reglerparameter so ausgewählt, dass der Regelkreis relativ schwach gedämpft ist. Die Regelgröße erreicht deshalb relativ schnell den Sollwert, schwingt allerdings mit zunächst großer Überschwingweite über den Sollwert hinaus. Bei der P-Regelung tritt erwartungsgemäß eine bleibende Regelabweichung auf. Der PID-Regler führt bei beiden Einstellregeln zu einer schnelleren Beruhigung der Regelgröße im Sollwert als der PI-Regler. Bei ihm ist der schnelle Eingriff, der zu dem steilen Anstieg der Führungsübergangsfunktion führt, durch den D-Anteil begründet.  $\square$

#### Aufgabe 8.1\* Einstellung eines PID-Reglers

1. Für eine Regelstrecke wurde der im linken Teil von Abb. 8.10 gezeigte Verlauf der Ausgangsgröße gemessen, nachdem zur Zeit  $t = 2\text{ s}$  die Eingangsgröße von 0 auf 5 erhöht wurde. Ermitteln Sie die Parameter für einen P-, PI- und PID-Regler mit Hilfe der Einstellregeln.
2. Die Regelstrecke wurde mit einem P-Regler zum Regelkreis geschlossen. Für unterschiedliche Werte der Reglerverstärkung  $k_P$  entstanden die in Abb. 8.10 (rechts) gezeigten Führungsübergangsfunktionen. Ermitteln Sie die Parameter für einen P-, PI- bzw. PID-Regler mit Hilfe der Einstellregeln.  $\square$



**Abb. 8.10:** Ausgangsgröße des Reaktors (links) und Führungsübergangsfunktion des Regelkreises bei Experimenten mit verschiedenen Verstärkungen  $k_P$  des P-Reglers (rechts)

## 8.3 Robuste PI-Regelung

### 8.3.1 Gegenkopplungsbedingung für I-Regler

In diesem Abschnitt werden PI-Regler untersucht, die sich durch eine besonders große Robustheit auszeichnen. Dem Entwurf dieser Regler liegt folgende Idee zu Grunde: Wenn man das Regelstreckenverhalten sehr wenig kennt, jedoch weiß, dass die Regelstrecke stabil ist und sich näherungsweise linear verhält, so kann man mit sehr kleinen Reglerparametern Experimente mit dem Regelkreis durchführen und in Abhängigkeit von den Experimentergebnissen die Parametereinstellung des Reglers verbessern.

Um so vorzugehen, muss vorher untersucht werden, unter welchen Bedingungen der Regelkreis für eine hinreichend kleine Reglerverstärkung stabil ist. Dabei wird zunächst nur mit einem reinen I-Regler

$$\text{I-Regler : } \begin{cases} \dot{x}_r(t) = y(t) - w(t), & x_r(0) = x_{r0} \\ u(t) = -k_I x_r(t) \end{cases} \quad (8.1)$$

gearbeitet, der mit der Regelstrecke

$$\Sigma_S : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}d(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t) \end{cases} \quad (8.2)$$

zum Regelkreis

$$\bar{\Sigma} : \begin{cases} \begin{pmatrix} \dot{\mathbf{x}}(t) \\ \dot{x}_r(t) \end{pmatrix} = \begin{pmatrix} \mathbf{A} & -\mathbf{b}k_I \\ \mathbf{c}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x}(t) \\ x_r(t) \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} w(t) + \begin{pmatrix} \mathbf{g} \\ 0 \end{pmatrix} d(t) \\ y(t) = (\mathbf{c}^T \ 0) \begin{pmatrix} \mathbf{x}(t) \\ x_r(t) \end{pmatrix} \end{cases}$$



führt. Im Folgenden wird eine notwendige Stabilitätsbedingung abgeleitet, in die nicht das Zustandsraummodell der Regelstrecke als Ganzes, sondern nur dessen statische Verstärkung  $k_s$  eingeht.

Bei der Ableitung dieser Bedingung wird vorausgesetzt, dass der Regelkreis stabil ist, also alle Eigenwerte  $\bar{\lambda}_i$  der Systemmatrix

$$\bar{\mathbf{A}} = \begin{pmatrix} \mathbf{A} & -\mathbf{b}k_I \\ \mathbf{c}^T & 0 \end{pmatrix}$$

negativen Realteil haben und folglich die Beziehung

$$\det(-\bar{\mathbf{A}}) = \prod_{i=1}^{n+1} (-\bar{\lambda}_i) > 0$$

gilt. Da die Regelstrecke nach Voraussetzung asymptotisch stabil ist, gilt die Ungleichung

$$\det(-\mathbf{A}) = \prod_{i=1}^n (-\lambda_i) > 0$$

mit  $\lambda_i$  als Eigenwerte von  $\mathbf{A}$ . Aus beiden Bedingungen kann man nach Zerlegung der ersten Determinante

$$\det(-\bar{\mathbf{A}}) = \det \begin{pmatrix} -\mathbf{A} & \mathbf{b}k_I \\ -\mathbf{c}^T & 0 \end{pmatrix} = \det(-\mathbf{A}) \det(-\mathbf{c}^T \mathbf{A}^{-1} \mathbf{b}k_I) > 0$$

folgern, dass in einem stabilen Regelkreis die Beziehung

$$-\mathbf{c}^T \mathbf{A}^{-1} \mathbf{b}k_I > 0$$

erfüllt ist. In dieser Beziehung steckt die statische Verstärkung  $k_s$  nach Gl. (4.29), so dass sie in die folgende Ungleichung umgeschrieben werden kann, in der an Stelle der Modellparameter  $\mathbf{A}$ ,  $\mathbf{b}$  und  $\mathbf{c}^T$  nur  $k_s$  vorkommt:

**Satz 8.1 (Gegenkopplungsbedingung)**

*Wenn ein Regelkreis bestehend aus einer asymptotisch stabilen Regelstrecke (8.2) und einem I-Regler (8.1) asymptotisch stabil ist, so gilt die Ungleichung*

$$\text{Gegenkopplungsbedingung für I-Regler: } k_s k_I > 0. \quad (8.3)$$

Aus der Ableitung dieser Bedingung geht hervor, dass sie eine *notwendige* Stabilitätsbedingung für den Regelkreis ist. Das heißt, wenn diese Bedingung verletzt ist, so ist der Regelkreis instabil. Wenn sie erfüllt ist, muss der Regelkreis aber nicht stabil sein.

Die Gegenkopplungsbedingung schreibt vor, dass der Reglerparameter  $k_I$  dasselbe Vorzeichen haben muss wie die statische Verstärkung  $k_s$ . Da  $k_s$  häufig positiv ist, muss also auch  $k_I$

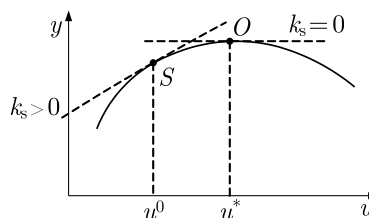
positiv sein, so dass das in jedem Regelkreis explizit eingetragene Minuszeichen in der Rückführung auf eine Gegenkopplung des Reglers führt und der Regler folglich versucht, der Ursache einer Regelabweichung entgegenzuwirken. Wenn beispielsweise die Regelgröße größer als die Führungsgröße ist, so erhält der Regler eine negative Regelabweichung  $w(t) - y(t)$  als Eingangsgröße. Wenn dies über eine hinreichend lange Zeit so ist, so ist auch die integrierte Regelabweichung negativ und auf Grund des positiven Reglerparameters auch die Stellgröße. Der Regler versucht also, mit einer negativen Stellgröße den zu großen Ausgang der Regelstrecke zu verkleinern und damit dem Sollwert anzupassen.

Die Gegenkopplungsbedingung hat zwei wichtige Konsequenzen:

- Damit man einen I-Regler verwenden kann, muss die statische Verstärkung der Regelstrecke die Bedingung
 
$$k_s \neq 0 \quad (8.4)$$
 erfüllen.
- Wenn sich die statische Verstärkung  $k_s$  der Regelstrecke während des Betriebs verändert, so darf sie nicht ihr Vorzeichen wechseln, wenn die Stabilität erhalten bleiben soll.

Die erste Bedingung ist eine *Existenzbedingung* für I-Regler. Sie besagt, dass man nur dann einen I-Regler einsetzen und dabei die Stabilität des Regelkreises durch geeignete Wahl des Reglerparameters  $k_I$  gewährleisten kann, wenn die Regelstrecke eine nicht verschwindende statische Verstärkung hat.

Wenn man das lineare Regelstreckenmodell durch Linearisierung aus einem nichtlinearen Modell gewinnt und dabei den Arbeitspunkt  $(\bar{u}, \bar{y})$  auf den Maximalwert (oder Minimalwert) einer nichtlinearen Kennlinie legt, so erhält man beispielsweise eine verschwindende statische Verstärkung und kann den gewählten Arbeitspunkt auf Grund der Bedingung (8.4) nicht durch einen I-Regler stabilisieren (Punkt  $O$  in Abb. 8.11). Obwohl es aus praktischen Gesichtspunkten sehr wünschenswert sein kann, den maximalen Wert der Ausgangsgröße dadurch zu erzeugen, dass man den Arbeitspunktwert  $\bar{u} = u^*$  wählt, ist dieser Arbeitspunkt aus regelungstechnischen Gründen also unzweckmäßig. Der Prozess kann nur solange im Arbeitspunkt  $O$  betrieben werden, bis eine Störung ihn aus diesem Punkt herausführt. Dann braucht man einen I-Regler, um die Wirkung der Störung auszugleichen. Mit einem solchen Regler entsteht aber ein instabiler Regelkreis, wie die Existenzbedingung aussagt.



**Abb. 8.11:** Interpretation der Existenzbedingung

Ein Ausweg besteht in der Verschiebung des Arbeitspunkts, beispielsweise in den Punkt  $S$ . Dort erreicht man zwar nicht den maximalen Wert der Ausgangsgröße, aber die Regelstrecke kann durch einen I-Regler sicher im Arbeitspunkt gehalten werden.

Die zweite Bedingung ist eine *Robustheitsbedingung*. Sie betrifft Veränderungen, die während des Betriebs der Regelung auftreten. Beispielsweise ruft eine Verschiebung des Arbeitspunkts eine Veränderung der Modellparameter hervor. Da trotz der Veränderungen der Regelstrecke der Reglerparameter  $k_I$  unverändert beibehalten wird, darf sich bei derartigen Parameterveränderungen die statische Verstärkung nicht im Vorzeichen ändern, weil andernfalls die notwendige Stabilitätsbedingung (8.4) verletzt würde und ein instabiler Regelkreis entstünde. Für Regelstrecken, deren statische Verstärkung ihr Vorzeichen wechselt, existiert demzufolge kein robuster I-Regler.

### 8.3.2 Reglereinstellung

Den Grundgedanken der in diesem Abschnitt behandelten Reglereinstellung kann man sich anhand von Abb. 8.12 klar machen, in dem die Eigenwerte der Matrix

$$\bar{A} = \begin{pmatrix} A & -bk_I \\ c^T & 0 \end{pmatrix}$$

für  $k_I = 0$  durch Kreuze markiert sind. Da die Regelstrecke stabil ist, liegen alle ihre Eigenwerte in der negativen komplexen Halbebene. Der Integrator des Reglers liefert in der offenen Kette einen Eigenwert im Ursprung der komplexen Ebene.

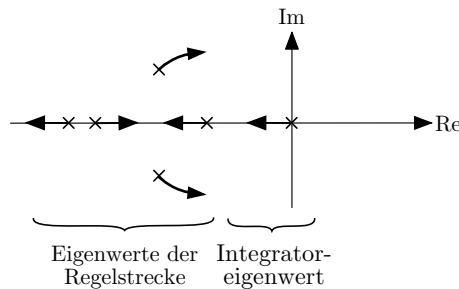


Abb. 8.12: Prinzip der Reglereinstellung

Wenn man nun den Reglerparameter  $k_I$  ein bisschen vergrößert, so wandern sämtliche Eigenwerte von den durch die Kreuze markierten Positionen weg. Dies geschieht stetig, d. h., die Eigenwerte springen nicht bei kleinen Reglerparameteränderungen. Die in der Abbildung angegebenen Pfeile zeigen, wie sich die Eigenwerte beispielsweise verändern können. Wenn man durch die Wahl eines hinreichend kleinen Wertes für  $k_I$  erreicht, dass der Integratoreigenwert vom Ursprung des Koordinatensystems in die linke Halbebene verschoben wird, während alle Streckeneigenwerte sich noch nicht so weit von ihren Ausgangspositionen entfernt haben,

dass sie die Imaginärachse überschritten und instabil werden könnten, so ist der geschlossene Regelkreis stabil.

Um dieses Prinzip anwenden zu können, muss man untersuchen, ob man bei Erfüllung der Gegenkopplungsbedingung (8.3) bei hinreichend kleiner Reglerverstärkung einen stabilen Regelkreis bekommt. Diese Frage kann man durch eine Betrachtung der Empfindlichkeit  $\frac{d\lambda}{dk_I}$  des im Ursprung des Koordinatensystems liegenden Eigenwertes bezüglich  $k_I$  beantworten und dabei zeigen, dass sich der Eigenwert in die negative komplexe Halbebene bewegt. Damit ist durch das Vorzeichen der statischen Verstärkung  $k_s$  vorgegeben, ob man mit einer positiven oder negativen Reglerverstärkung arbeiten muss: Das Vorzeichen von  $k_I$  muss mit dem der statischen Regelstreckenverstärkung  $k_s$  übereinstimmen.

Da sich die Eigenwerte des Regelkreises bei einer Veränderung von  $k_I$  stetig verändern, ist damit auch klar, dass der Regelkreis bei hinreichend kleinem Betrag der Reglerverstärkung  $k_I$  asymptotisch stabil ist. Auf Grund des I-Anteils des Reglers ist außerdem gewährleistet, dass der Regelkreis für sprungförmige Führungs- und Störsignale keine bleibende Regelabweichung hat.

---

### Einstellverfahren 8.3 *Einstellung robuster I-Regler*

---

**Gegeben:** Asymptotisch stabile Regelstrecke  $\Sigma_S$   
 Güteforderungen: Stabilität und Sollwertfolge bei sprungförmigen Führungs- und Störsignalen.

1. Bestimmen Sie experimentell die statische Verstärkung  $k_s$  der Regelstrecke.
2. Überprüfen Sie die Existenzbedingung (8.4). Ist diese Bedingung nicht erfüllt, so muss die Regelungsaufgabe mit einer anderen Stell- oder Regelgröße gelöst werden.
3. Wenn die statische Verstärkung positiv ist, legen Sie einen kleinen positiven Wert für den Reglerparameter  $k_I$  fest, andernfalls einen kleinen negativen Wert.
4. Schließen Sie den Regelkreis und bestimmen Sie experimentell die Führungsübergangsfunktion. Sie können den Betrag des Reglerparameters erhöhen, solange die Experimente mit dem geschlossenen Kreis zeigen, dass der Kreis eine genügend große Stabilitätsreserve aufweist.

**Ergebnis:** I-Regler, für den der Regelkreis stabil ist.

---

Um die statische Verstärkung  $k_s$  der Regelstrecke zu bestimmen, wird die Stellgröße sprungförmig verändert ( $u(t) = \bar{u}\sigma(t)$ ) und der Endwert der Systemantwort  $y(\infty)$  gemessen. Dann erhält man die statische Verstärkung aus der Beziehung

$$k_s = \frac{y(\infty)}{\bar{u}}.$$

Wenn man die Experimente mit dem Regelkreis mit einem betragsmäßig sehr kleinen Reglerparameter  $k_I$  beginnt, so wird sich die Regelgröße sehr langsam an den vorgegebenen Soll-

wert angleichen. Dies zeigt, dass alle Eigenwerte des Regelkreises weit in der linken komplexen Ebene liegen und eine kleine Veränderung des Reglerparameters die Stabilität des Regelkreises nicht gefährden kann. Man wird dann den Reglerparameter betragsmäßig vergrößern, bis der Regelkreis bei einem erneuten Experiment deutliche Schwingungen ausführt. Dann hängt es von der Regelungsaufgabe ab, ob man sich lieber für einen Regelkreis entscheidet, der langsam aber ohne Überspringen den Sollwert annimmt, oder ob man den Reglerparameter so wählt, dass ein Überspringen auftritt, die Regelgröße aber schneller in den Sollwert einschwingt.

### 8.3.3 Erweiterung auf PI-Regler

Mit dem I-Regler können die Güteforderungen nach Stabilität und Sollwertfolge erfüllt werden, aber der Regelkreis ist i. Allg. sehr langsam. Um ein schnelleres Übergangsverhalten zu erreichen, muss ein Proportionalanteil in den Regler eingeführt werden. Der PI-Regler ist durch

$$\text{PI-Regler : } \begin{cases} \dot{x}_r(t) = y(t) - w(t), & x_r(0) = x_{r0} \\ u(t) = -k_I x_r(t) - k_P (y(t) - w(t)) \end{cases} \quad (8.5)$$

beschrieben.

Bei der Suche nach einer geeigneten Proportionalverstärkung  $k_P$  kann man wie beim I-Anteil vorgehen und zunächst mit einem kleinen Wert beginnen. Dabei besteht im Gegensatz zum I-Anteil nicht die Gefahr, dass der Regelkreis bei falscher Vorzeichenwahl instabil wird, weil der P-Anteil keinen neuen Eigenwert an der Stabilitätsgrenze in den Regelkreis einführt. Für hinreichend kleinen Parameter  $k_P$  ist der Regelkreis stabil, unabhängig davon, ob der Parameter positiv oder negativ ist.

Da man mit dem P-Anteil den Regelkreis schneller machen will, ist die Vorzeichenwahl für den Reglerparameter  $k_P$  entscheidend. Auch hier gilt die Gegenkopplungsbedingung, d. h.,  $k_P$  soll dasselbe Vorzeichen wie die statische Verstärkung  $k_s$  haben. Die Wirkung einer schrittweisen betragsmäßigen Vergrößerung dieses Reglerparameters kann an den experimentell ermittelten Führungsübergangsfunktionen bewertet und auf diese Weise ein geeigneter Parameterwert festgelegt werden.

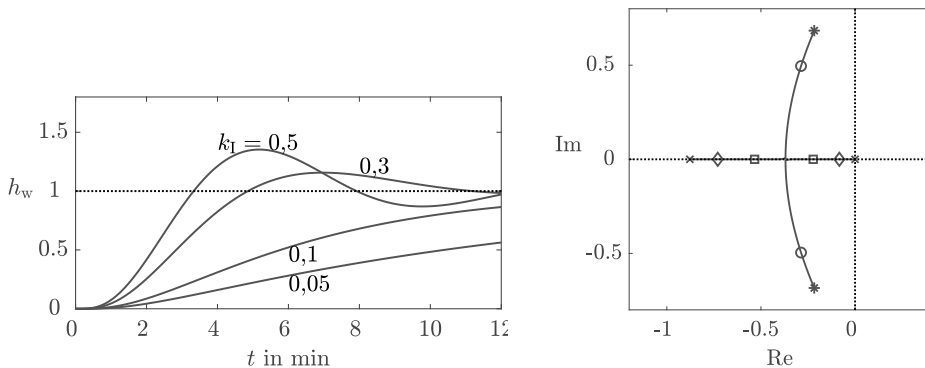
Um P- und I-Anteil in ein ausgewogenes Verhältnis zu bringen, kann man zwei Wege wählen. Auf dem ersten Weg wird zunächst der I-Anteil eingestellt und anschließend der P-Anteil hinzugefügt. Man kann aber auch P- und I-Anteil in ein zweckmäßiges Verhältnis bringen und dann beide Anteile gleichmäßig erhöhen. In diesem Fall ersetzt man die Ausgabegleichung des Reglergesetzes (8.5) durch die Gleichung

$$u(t) = -a(k_I x_r(t) + k_P (y(t) - w(t))), \quad (8.6)$$

in der  $a$  ein Einstellfaktor ist. Durch  $a$  wird der Einfluss des P- und des I-Anteiles auf das Regelkreisverhalten gleichzeitig erhöht, während durch die Faktoren  $k_P$  und  $k_I$  die Gewichte von P- und I-Anteil zueinander verändert werden. Das folgende Beispiel zeigt, dass dieses Vorgehen zweckmäßig ist.

**Beispiel 8.1 (Forts.)** *Einstellung des Temperaturregelkreises eines Rührkesselreaktors*

Der Temperaturregler soll jetzt mit den in diesem Abschnitt behandelten Vorgehen eingestellt werden. Wie aus Abb. 8.5 zu sehen ist, hat die Regelstrecke eine positive statische Verstärkung, so dass bei der Reglereinstellung mit positiven Reglerparametern  $k_I$  gearbeitet werden muss.



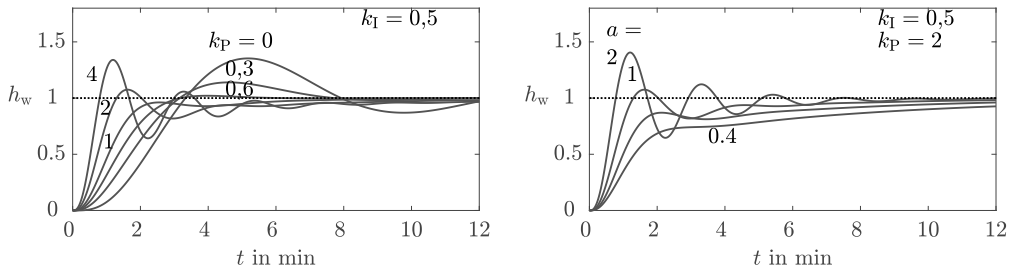
**Abb. 8.13:** Führungsübergangsfunktionen für den I-Regelkreis (links) und Lage der zwei größten Eigenwerte des I-Regelkreises bei unterschiedlicher Reglerverstärkung (rechts)

Wenn der Regelkreis mit einem reinen I-Regler geschlossen und  $k_I$  zwischen 0,05 und 0,5 verändert wird, erhält man aus Experimenten mit dem Regelkreis die in Abb. 8.13 links gezeigten Führungsübergangsfunktionen, bei denen die Ausgangsgröße in Kelvin gemessen wird. Erwartungsgemäß nähert sich die Übergangsfunktion für kleine Werte von  $k_I$  dem Endwert nur sehr langsam. Für größere Werte wird der Regelkreis schneller und zeigt bei  $k_I = 0,5$  ein schwingendes Verhalten.

Um zu untersuchen, wie sich die Eigenwerte des Regelkreises entsprechend Abb. 8.12 für dieses Beispiel verhalten, wurden mit dem bei den Simulationsuntersuchungen verwendeten Modell die Eigenwerte des Regelkreises ausgerechnet und die beiden größten Eigenwerte in der komplexen Ebene dargestellt (Abb. 8.13 (rechts)). Die Eigenwerte liegen bei  $k_I = 0$  in den durch die beiden Kreuze gekennzeichneten Stellen, wobei der Eigenwert bei null durch den I-Anteil des Reglers verursacht ist. Für  $k_I = 0,05$  erhält man die durch  $\diamond$  gekennzeichneten Werte, die zeigen, dass sich bei dieser Reglerverstärkung der Integratoreigenwert noch nicht weit von seiner ursprünglichen Lage im Ursprung entfernt hat. Dementsprechend langsam ist das Übergangsverhalten des Regelkreises. Bei einer Erhöhung der Reglerverstärkung auf  $k_I = 0,1$  und  $k_I = 0,3$  verschieben sich die Eigenwerte auf die durch  $\square$  bzw.  $\circ$  gekennzeichneten Punkte. Gleichzeitig wird das Übergangsverhalten schneller. Da das zweite Eigenwertpaar konjugiert komplex ist, schwingt die Übergangsfunktion. Das Schwingen wird größer, wenn die Reglerverstärkung auf  $k_I = 0,5$  erhöht wird, weil dann die durch die Sterne gekennzeichneten Eigenwerte entstehen. Man kann sich vorstellen, dass bei weiterer Erhöhung der Reglerverstärkung der Regelkreis instabil werden kann. In der Praxis steht das hier verwendete Modell der Regelstrecke nicht zur Verfügung, so dass dieses Verhalten der Eigenwerte nicht in der komplexen Ebene untersucht werden kann, sondern die Reglerverstärkung anhand der Übergangsfunktionen in Abb. 8.13 (links) festgelegt werden muss.

Verglichen mit dem Verhalten der Regelstrecke, deren Übergangsfunktion nach etwa 5 Minuten den statischen Endwert erreicht, ist der Regelkreis für alle Einstellwerte sehr langsam. Dies liegt daran, dass bisher nur ein I-Regler eingesetzt wurde. Eine weitere Verbesserung ist nur durch einen P-Anteil im Regler zu erreichen.

Wenn man bei vorgegebenem Integralanteil des Reglers (z. B.  $k_I = 0,5$ ) nun den Proportionalanteil  $k_P$  erhöht, so wird die Führungsübergangsfunktion im Anfangsteil schneller. Das ist der Teil, in dem der PI-Regler über seinen Proportionalanteil eine größere Stellgröße erzeugt als der reine I-Regler. Der linke Teil von Abb. 8.14 zeigt die dabei entstehenden Führungsübergangsfunktionen. Die Kurve für  $k_P = 4$  ähnelt der für den P-Regler mit  $k_P = 4,55$  in Abb. 8.6, nur dass hier auf Grund des vorhandenen I-Anteils keine bleibende Regelabweichung auftritt. Der Integralanteil ist allerdings langsam eingestellt, so dass die Regelabweichung erst nach Ablauf des hier dargestellten Zeitintervalls vollständig abgebaut wird. Die Kurve für  $k_P = 0$  ist dieselbe wie die mit  $k_I = 0,5$  gekennzeichnete Kurve im linken Teil von Abb. 8.13.



**Abb. 8.14:** Führungsübergangsfunktionen des Temperaturregelkreises mit  $k_I = 0,5$  und verändertem P-Anteil (links) bzw. für den Einstellfaktor  $a \in \{0,4, 0,6, 1, 2\}$  (rechts)

Die zweite Möglichkeit der Reglereinstellung besteht im gleichzeitigen Anheben der Reglerverstärkungen für den P- und den I-Anteil. Für den Regler (8.6) mit  $k_P = 2$  und  $k_I = 0,5$  entstehen die im rechten Teil von Abb. 8.14 gezeigten Übergangsfunktionen, wobei die Kurve für  $a = 1$  mit der Kurve für  $k_P = 2$  im linken Teil der Abbildung übereinstimmt. Bei einer Erhöhung des Einstellfaktors  $a$  wird der Kreis insgesamt schneller. Für  $a = 2$  ist das Ergebnis mit dem des PI-Reglers in Abb. 8.6 vergleichbar.

Im Vergleich zu den im Abschn. 8.2 angegebenen Einstellregeln hat die hier behandelte Vorgehensweise den Vorteil, dass die Abhängigkeit des Regelkreisverhaltens von den einzelnen Reglerparametern überschaubar ist. Während beim ersten Verfahren feste Reglerparameter aus einer Tabelle abgelesen wurden und es keine Richtwerte für deren Veränderungen gab, ist bei dem hier behandelten Vorgehen offenkundig, wie die Parameter zu korrigieren sind, damit sich das Regelkreisverhalten in einer gewünschten Weise verändert. Das Beispiel zeigt, dass auf diesem Wege auch dasselbe Regelkreisverhalten wie beim Ziegler-Nichols-Verfahren erzeugt werden kann, dass aber darüber hinaus während der Experimente offenkundig wird, wie bessere Einstellungen erzielt werden können, so dass insbesondere das unerwünscht große Überschwingen der Führungsübergangsfunktion bei der Ziegler-Nichols-Einstellung wesentlich gemindert werden konnte.

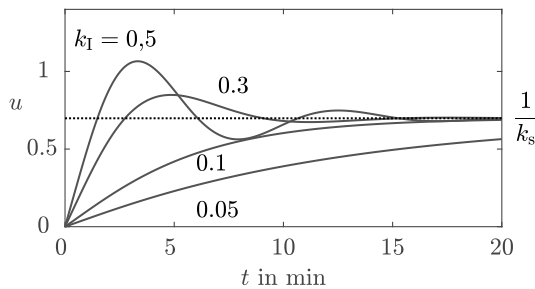
**Diskussion.** Die Wirkungsweise eines reinen I-Reglers kann man sich anhand von Abb. 8.15 erklären. Die Stellgröße  $u(t)$  ist proportional zur integrierten Regelabweichung  $e(t) = y(t) - 1$ . Bei kleiner Reglerverstärkung  $k_I = 0,05 \dots 0,3$  wird die Stellgröße langsam vergrößert, bis sie ihren statischen Endwert erreicht hat. Dieser ist für alle Reglerparameter derselbe, nämlich

$$u(\infty) = \frac{1}{k_s},$$

denn für diesen Wert gilt die Beziehung

$$y(\infty) = k_s \cdot u(\infty) = k_s \cdot \frac{1}{k_s} = 1,$$

so dass die Regelabweichung verschwindet ( $e(\infty) = y(\infty) - 1 = 0$ ) und der Integrator im Regler seinen Wert nicht mehr ändert.



**Abb. 8.15:** Stellgrößenverlauf beim I-Regler mit kleiner Reglerverstärkung  $k_I$  und  $w = 1$

Solange die Stellgröße langsam gesteigert wird, erreicht sie ohne Überspringen den statischen Endwert, der Regelkreis ist stabil und die Regelgröße nimmt nach einiger Zeit den Sollwert an. Wird jedoch die Reglerverstärkung erhöht, so beginnt das System zu schwingen, weil die Stellgröße zu schnell steigt und die Regelgröße auf Grund der zeitlichen Verzögerung in der Regelstrecke den Sollwert überschreitet. Die Schwingung klingt für hinreichend kleine Reglerverstärkung ab, kann aber für größere Reglerverstärkung aufklingen. Die Stabilität ist also für große Reglerverstärkungen nicht mehr gesichert.

Abbildung 8.15 zeigt auch, dass der I-Anteil im Regler erst sehr spät (bei  $t \approx 15 \dots 20$ ) den statischen Endwert annimmt. Dies erklärt, warum in den vorherigen Abbildungen mit wesentlich kürzerer Zeitachse die Regelabweichung noch nicht vollständig abgebaut wurde. Diese Erklärung gilt auch für PI- und PID-Regler, denn weder P- noch D-Anteil bauen die Regelabweichung für große Zeiten ab – dies kann allein der I-Anteil.

Die Reglereinstellung wird durch die Tatsache ermöglicht, dass man bei richtiger Wahl des Vorzeichens für hinreichend kleine Parameterwerte  $k_I$  einen stabilen Regelkreis erhält. Weil man dies weiß, muss man lediglich darauf achten, dass man die Experimente mit kleinen Reglerparametern beginnt. Man weiß darüber hinaus, dass man den Reglerparameter bis zu einem oberen Grenzwert  $\bar{k}_I$  erhöhen kann, bevor der Kreis instabil wird. Ohne Modell kann man die Stabilitätsgrenze  $\bar{k}_I$  allerdings nicht ausrechnen. Sie liegt aber i. Allg. deutlich über den Parameterwerten, bei denen der Regelkreis stark zu schwingen anfängt, so dass man bei den Experimenten nicht in die Nähe dieses Grenzwertes kommt.

In diesem Beispiel wurde mit demselben Modell gerechnet, für das in Aufgabe 6.7 als Stabilitätsgebiet die Ungleichung (A.30)

$$0 < k_I < 1,42$$

bestimmt wurde. Die Experimente haben deutlich gemacht, dass man den Reglerparameter nicht größer als ein Drittel der Stabilitätsgrenze  $\bar{k}_I = 1,42$  macht, um ein zu großes Schwingen des Regelkreises zu vermeiden. Es besteht also keine Gefahr, dass der Regelkreis bei den Experimenten instabil wird.  $\square$



**Aufgabe 8.2** *Einstellregeln für PID-Regler*

Auf der Homepage [www.atp.rub.de/Buch/AT](http://www.atp.rub.de/Buch/AT) erhalten Sie in der Datei *AufgabeZ8* einen Simulink-Block, der die in dieser Aufgabe zu betrachtende Regelstrecke enthält. Sie können den Block nicht im Detail einsehen und befinden sich deshalb in derselben Situation wie ein Anlagenfahrer, der ohne Modell seines Prozesses einen Regler einstellen soll. „Experimente“ mit dem Prozess können Sie dadurch durchführen, dass Sie die Eingangsgröße des Simulink-Blocks vorgeben und die Ausgangsgröße grafisch darstellen.

Stellen Sie P-, PI- und PID-Regler für diese Regelstrecke mit den in diesem Kapitel behandelten Methoden ein und analysieren Sie das Verhalten des Regelkreises. Die Führungsübergangsfunktion des Regelkreises soll möglichst schnell und mit möglichst kleinem Überspringen den Sollwert erreichen. Wie verändert sich das Regelkreisverhalten, wenn Sie das Regelstreckenverhalten beim Einstellverfahren 8.1 durch eine größere Totzeit  $T_t$  und kleinere Zeitkonstante  $T$  approximieren? Können Sie mit dem Einstellverfahren 8.3 die Reglerparameter so wählen, dass die Führungsübergangsfunktion gar nicht überschwingt?

Diskutieren Sie an diesem Beispiel die Vor- und Nachteile der behandelten Methoden. □

## 8.4 Ausblick: Verfahren für den Reglerentwurf

In diesem Kapitel wurden nur sehr einfache Verfahren für die Festlegung von Reglerparametern behandelt. Diese Verfahren sind nur dann anwendbar, wenn die folgenden Voraussetzungen erfüllt sind:

- Die Regelstrecke ist stabil.
- Es können Experimente mit der Regelstrecke durchgeführt werden.
- Es sind nur schwache Güteforderungen zu erfüllen.

Für das Verfahren von ZIEGLER und NICHOLS wurde darüberhinaus vorausgesetzt, dass die Übergangsfunktion der Regelstrecke ihren statischen Endwert ohne Überspringen annimmt. Die in Tabelle 8.1 angegebenen Beziehungen sind reine Faustformeln, die nicht berücksichtigen, an welcher Stelle des Regelkreises die Störung eingreift.

Ein wesentlicher Teil des Know-hows des Regelungstechnikers bezieht sich auf die geeignete Wahl der Reglerstruktur und die Festlegung der Reglerparameter, so dass der entstehende Regelkreis „strenge“ Güteforderungen erfüllt, also außer der Stabilität und Sollwertfolge auch Forderungen an das dynamische Übergangsverhalten genügt. So sollen Regelkreise in der Energietechnik die vorgegebenen Sollwerte ohne Überspringen erreichen, um auch unter ökonomischen Gesichtspunkten zweckmäßig zu arbeiten, während es bei der Bewegungssteuerung von Robotern vor allem auf die Zeit ankommt, in der der Greifer eine vorgegebene Position erreicht. Die Regelung, die die Bewegung der Magnetschwebbahn bei hoher Geschwindigkeit in vertikaler und horizontaler Richtung an den Verlauf des Fahrweges anpasst, muss so gut sein, dass die Position des Wagens trotz Störungen durch Wind, Änderungen der Fahrwegsrichtung und bei wechselnder Masse des Fahrzeugs in einem Band von wenigen Millimetern Breite gehalten wird.

Regelkreise, die derartige Güteforderungen erfüllen müssen, bestimmen die Qualität einer Technologie bzw. machen die Nutzung bestimmter Wirkungsprinzipien überhaupt erst möglich.

Ihre Regelungen können nur mit Methoden bestimmt werden, die von einer detaillierten Analyse der dynamischen Eigenschaften der Regelstrecke ausgehen, die die Art der Störungen und die Störeingriffspunkte berücksichtigen und die die Regelung an die Art der Führungsgrößen anpassen. Derartige Methoden sind z. B. in den Lehrbüchern [69, 70] beschrieben.

**Rechnergestützter Entwurf.** Die numerischen Berechnungsschritte, die beim Reglerentwurf wie auch bei den in den nachfolgenden Kapiteln behandelten Entwurfsproblemen für Zustandsbeobachter und Diagnosesysteme auftreten, werden heute mit Hilfe rechnergestützter Entwurfsprogramme durchgeführt. Diese Werkzeuge sind in kommerziellen Programmen für unterschiedliche Entwurfsaufgaben und innerhalb der Projektierungssoftware industrieller Automatisierungssysteme verfügbar. Sie erleichtern dem Automatisierungstechniker u. a. die folgenden Schritte:

- Ausführung numerischer Operationen,
- Verwaltung der Daten, die das Regelstreckenmodell, den Regler, den daraus entstehenden Regelkreis usw. beinhalten,
- grafische Aufbereitung der Entwurfsergebnisse, z. B. für die grafische Darstellung der Führungsübergangsfunktion.

Nicht rechnergestützt, sondern vom Ingenieur durchzuführen sind die folgenden Schritte:

- Auswahl der Reglerstruktur,
- Auswahl des im nächsten Entwurfsschritt anzuwendenden Verfahrens oder Algorithmus,
- Bewertung der Entwurfsergebnisse in Bezug auf die gestellten Güteforderungen,
- Entscheidung über den Fortgang des Entwurfs.

Um diese Schritte durchführen zu können, sind fundierte automatisierungstechnische Kenntnisse notwendig, von denen ein wichtiger Teil in diesem Buch behandelt wird.

## Literaturhinweise

Die Einstellregeln von ZIEGLER und NICHOLS sind bereits 1942 in [111] publiziert worden. Seit dieser Zeit wurden und werden weitere Einstellregeln erarbeitet und veröffentlicht, die für bestimmte Standardsituationen der Reglereinstellung eine schnelle und zweckmäßige Parameterwahl ermöglichen. Auf Robustheitsuntersuchungen gegründet sind die im Abschn. 8.3 beschriebenen Einstellregeln, die auch für Mehrgrößensysteme anwendbar sind und die ausführlich in [70, 78] nachgelesen werden können. In den genannten Büchern finden sich auch ausführliche Beispiele.

# 9

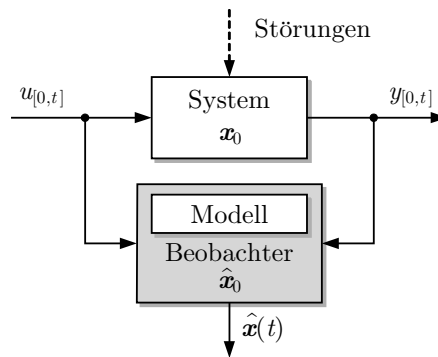
## Zustandsbeobachtung kontinuierlicher Systeme

*Der nicht messbare Zustand eines Systems kann aus dem Verlauf der Eingangs- und der Ausgangsgröße mit Hilfe eines Zustandsbeobachters ermittelt werden. Dieses Kapitel behandelt den Luenbergerbeobachter und seine wichtigsten Einsatzgebiete.*

### 9.1 Beobachtungsaufgabe

Eine wichtige Voraussetzung für die Lösung von Steuerungs- und Prozessüberwachungsaufgaben besteht in der Bereitstellung von Online-Informationen über den Prozess. Viele dieser Informationen können direkt über Messeinrichtungen beschafft, ausgewertet, angezeigt und für die Festlegung von Steuereingriffen eingesetzt werden. Wenn wichtige Prozessgrößen jedoch nicht messbar sind, ist es notwendig, diese Größen aus messbaren Größen unter Zuhilfenahme eines dynamischen Prozessmodells zu berechnen. Dieser Weg wird im Folgenden untersucht.

In einer Standardform wird die Aufgabe so formuliert, dass von einem dynamischen System der Verlauf  $u_{[0,t]}$  der Eingangsgröße und der Verlauf  $y_{[0,t]}$  der Ausgangsgröße im Zeitintervall  $[0, t]$  bekannt ist und mit Hilfe des Zustandsraummodells der aktuelle Zustand  $x(t)$  berechnet werden soll (Abb. 9.1). Man spricht dann davon, dass der Zustand *beobachtet* wird. In der Automatisierungstechnik macht man also einen Unterschied zwischen einer Messung und einer Beobachtung, wobei der zweite Begriff für Methoden verwendet wird, bei denen die gewünschten Signale nicht messbar sind und mit einem Modell aus anderen Messgrößen rekonstruiert werden.



**Abb. 9.1:** Idee des Zustandsbeobachters

Die Beobachtungsaufgabe lautet folgendermaßen:

**Beobachtungsaufgabe für kontinuierliche Systeme**

Gegeben: Zustandsraummodell des Systems  $\Sigma$   
 Verlauf der Eingangsgröße  $u_{[0,t]}$   
 Verlauf der Ausgangsgröße  $y_{[0,t]}$   
 Gesucht: aktueller Zustand  $x(t)$

Entsprechend Abb. 9.1 wird diese Aufgabe so gelöst, dass der in diesem Kapitel eingeführte Beobachter einen Näherungswert  $\hat{x}(t)$  erzeugt, der sich mit fortschreitender Zeit  $t$  immer besser dem wahren Systemzustand  $x(t)$  annähert.

In den meisten Anwendungen will man das Beobachtungsproblem nicht für einen fest vorgegebenen Zeithorizont  $t$ , sondern fortlaufend für ein immer größer werdendes Zeitintervall  $[0, t]$  lösen. Der Beobachter soll also ein dynamisches System mit den Eingängen  $u(t)$  und  $y(t)$  sein, das online den Ausgang  $\hat{x}(t)$  erzeugt. Es wird sich zeigen, dass der Zustandsbeobachter ein allgemeines Konzept darstellt, bei dem das Modell des zu beobachtenden Systems  $\Sigma$  verwendet wird, um die Näherung  $\hat{x}(t)$  zu erzeugen.

Die Beobachtungsaufgabe kann auch so formuliert werden, dass an Stelle des aktuellen Zustands  $x(t)$  der Anfangszustand  $x_0$  berechnet werden soll. Diese Aufgabe ist jedoch dieselbe wie die hier angegebene, denn aus  $x_0$  kann mit dem Modell und mit der bekannten Eingangsgröße  $u_{[0,t]}$  der aktuelle Zustand  $x(t)$  ermittelt werden und umgekehrt.

**Informationsgehalt von Messwerten und Messwertfolgen.** Zur Lösung der Beobachtungsaufgabe muss der Verlauf der Messgröße bekannt sein. Es reicht nicht, den Messwert nur zu einem bestimmten Zeitpunkt  $t$  zu kennen. Ein einfaches Beispiel soll der Unterschied zwischen den Informationen, die man durch eine einzelne Messung  $y(t)$  erhält bzw. aus dem zeitlichen Verlauf  $y_{[0,t]}$  der Messgröße ablesen kann, verdeutlichen.

**Beispiel 9.1** Beobachtung nicht messbarer Größen

Wenn die Messgröße  $y_m(t)$  der Ausgang eines Messumformers ist, dessen zeitliche Verzögerung im Vergleich zur Änderungsgeschwindigkeit der Messgröße vernachlässigbar ist, so kann der Messumformer als statisches System mit dem Übertragungsfaktor  $k_m$  aufgefasst und durch

$$y_m(t) = k_m y(t) \quad (9.1)$$

beschrieben werden, wobei  $y(t)$  die zu messende Größe und  $y_m(t)$  das Messergebnis bezeichnet. Durch Umstellen der Gleichung erhält man die Beziehung

$$y(t) = \frac{1}{k_m} y_m(t),$$

mit deren Hilfe man aus dem Messwert  $y_m(t)$  zum Zeitpunkt  $t$  das gewünschte Messsignal  $y(t)$  zum selben Zeitpunkt berechnen kann. Diese Berechnung kann zu jedem Zeitpunkt  $t$ , zu dem ein Messwert vorliegt, wiederholt werden. Eine Beobachtungsaufgabe muss dabei nicht gelöst werden. Diese Vorgehensweise ist in allen Situationen anwendbar, in denen zwischen dem gemessenen Signal  $y_m(t)$  und dem gesuchten Signal  $y(t)$  ein statischer Zusammenhang (9.1) besteht.

Anders sieht es aus, wenn zwischen dem gewünschten Signal und der tatsächlich gemessenen Größe ein dynamisches System liegt. Dieser Fall tritt beispielsweise auf, wenn man aus einer Füllstandsmessung  $h(t)$  den Zufluss  $q(t)$  in einen Reaktor ermitteln will. Dann gilt für den Füllstand  $h(t)$  die Differentialgleichung

$$\frac{dh(t)}{dt} = \frac{1}{A} q(t),$$

in der  $A$  den Querschnitt des zylindrischen Reaktors und  $q(t)$  den zu ermittelnden Volumenstrom bezeichnen. Durch Umstellung der Gleichung erhält man für den gesuchten Wert die Beziehung

$$q(t) = A \frac{dh(t)}{dt},$$

die besagt, dass man mit einer einzelnen Messung  $h(t)$  den Zufluss *nicht* bestimmen kann. Man muss den Verlauf des Signals  $h(t)$  kennen, um daraus die zeitliche Ableitung  $\frac{dh(t)}{dt}$  und aus dieser den Zufluss  $q(t)$  berechnen zu können.

Dieses einfache Beispiel verdeutlicht, dass man bei der Rekonstruktion nicht messbarer Signale aus messbaren Signalen nicht nur den Momentanwert der Messgrößen, sondern deren zeitlichen Verlauf kennen muss, wenn zwischen dem zu bestimmenden Signal und der Messgröße eine dynamische Komponente liegt.  $\square$

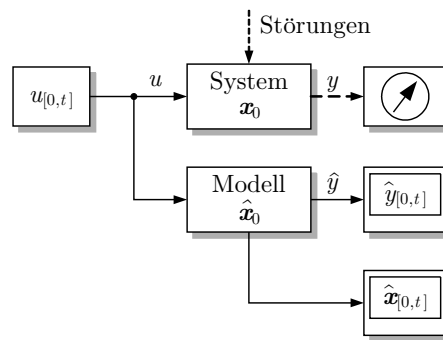
## 9.2 Luenbergerbeobachter

### 9.2.1 Grundidee

Bei der Analyse der Beobachtbarkeit linearer Systeme wurde im Abschn. 5.3 bereits gezeigt, dass man aus der Messung der Ausgangsgröße  $y(t)$  sowie deren ersten  $n - 1$  Ableitungen den aktuellen Zustand berechnen kann. Gleichung (5.15) auf S. 164 beweist, dass dies unter der im Beobachtbarkeitskriterium angegebenen Bedingung möglich ist. Technisch anwendbar ist dieser Berechnungsweg jedoch nicht, weil die zeitlichen Ableitungen der Ausgangsgröße i. Allg. nicht messbar sind und ihre Berechnung durch Differentiation technisch nicht durchführbar ist,

weil mit dem Signal auch das bei jeder Messung auftretende Messrauschen differenziert wird und dabei schon nach der ersten oder zweiten Ableitung das Rauschsignal das Berechnungsergebnis stärker bestimmt als das Nutzsignal.

Es wird deshalb im Folgenden ein anderer Lösungsweg betrachtet, bei dem das Modell parallel zum System mit der bekannten Eingangsgröße  $u(t)$  angesteuert wird, um aus dem auf einem Rechner realisierten Modell den Zustand auszulesen. Um den Modellzustand vom wahren Systemzustand  $x(t)$  unterscheiden zu können, wird der Modellzustand mit  $\hat{x}(t)$  bezeichnet. Dieser Lösungsweg für das Beobachtungsproblem ist in Abb. 9.2 gezeigt. Die Eingangsgröße  $u(t)$  des Systems wird von einem Automatisierungssystem oder einem Operator vorgegeben und gleichzeitig als Modelleingang verwendet. Am System kann man den Ausgang  $y(t)$  messen, während das Modell den Modellausgang liefert, der mit  $\hat{y}(t)$  bezeichnet wird.



**Abb. 9.2:** Parallelschaltung von System und Modell

Wenn das Modell den betrachteten Prozess hinreichend genau beschreibt, was im Folgenden vorausgesetzt wird, so stimmt die mit dem Modell berechnete erzwungene Bewegung  $\hat{y}_{\text{erzw}}(t)$  mit der erzwungenen Bewegung  $y_{\text{erzw}}(t)$  des Systems überein. Unterschiede gibt es bezüglich der Eigenbewegung, denn wenn der Systemzustand  $x(t)$  nicht messbar ist, kennt man auch den Anfangszustand  $x_0$  nicht und muss im Modell einen irgendwie gewählten Anfangszustand  $\hat{x}_0$  einsetzen. Allerdings klingt die Eigenbewegung ab, wenn das System stabil ist, und nach hinreichend langer Zeit gilt (näherungsweise) sowohl  $\hat{y}(t) = y(t)$  als auch  $\hat{x}(t) = x(t)$ . Aus dem Modell kann dann der unbekannte Systemzustand abgelesen werden.

Im Vergleich zu den im Abschn. 5.3 behandelten Methoden hat das in Abb. 9.2 gezeigte Vorgehen den Vorteil, dass es ohne Differentiation der Messgröße auskommt und den Zustand im Modell kontinuierlich dem des Systems nachführt. Der Zustand  $\hat{x}(t) = x(t)$  steht damit nach dem Abklingen der Eigenbewegung zu jedem Zeitpunkt  $t$  zur Verfügung. Das Schema kann also für einen sich mit der aktuellen Zeit  $t$  verändernden Zeithorizont  $[0, t]$  angewendet werden und passt online den Näherungswert  $\hat{x}(t)$  der Bewegung des Systems an.

Die beschriebene Vorgehensweise muss jedoch noch etwas erweitert werden, damit der Beobachtungsfehler  $e(t) = x(t) - \hat{x}(t)$  unter typischen praktischen Randbedingungen tatsächlich verschwindet bzw. hinreichend klein wird. Die Erweiterung soll sicherstellen, dass der Beobachter auch anwendbar ist,

- wenn das System instabil ist und dessen Eigenbewegung  $x_{\text{frei}}(t)$  folglich nicht abklingt und
- wenn nicht messbare Störungen auftreten, die das Systemverhalten beeinflussen, aber im Modell nicht berücksichtigt werden können.

Die Erweiterung der Struktur aus Abb. 9.2 betrifft die Nutzung der im gemessenen Ausgang  $y(t)$  erhaltenen Informationen für die Verbesserung des Beobachtungsergebnisses. Wie in Abb. 9.1 gezeigt ist, verwenden Beobachter nicht nur die Eingangsgröße  $u(t)$ , sondern auch die Ausgangsgröße  $y(t)$ . Der Systemzustand wird nicht mehr vorhergesagt, indem er mit dem Modell aus der bekannten Eingangsgröße  $u(t)$  berechnet wird, sondern *beobachtet*. Beide Vorgehensweisen unterscheiden sich, wie ein Vergleich der Abbildungen 9.1 und 9.2 zeigt, in den verwendeten Informationen, wobei die Beobachtung auch die gemessene Ausgangsgröße  $y(t)$  nutzt. Wie diese Information verarbeitet wird, um den Beobachtungsfehler  $e(t)$  möglichst klein zu machen, wird im folgenden Abschnitt behandelt.

### 9.2.2 Beobachterstruktur

Bei dem von LUENBERGER vorgeschlagenen Beobachter wird die Differenz  $y(t) - \hat{y}(t)$  zwischen dem gemessenen Systemausgang und dem Modellausgang auf das Modell zurückgeführt (Abb. 9.3). Dafür wird das Modell um den neuen Eingang  $u_B(t)$  erweitert. Die Wirkung dieser Rückführung wird im Folgenden untersucht und zwar zunächst für das ungestörte System.

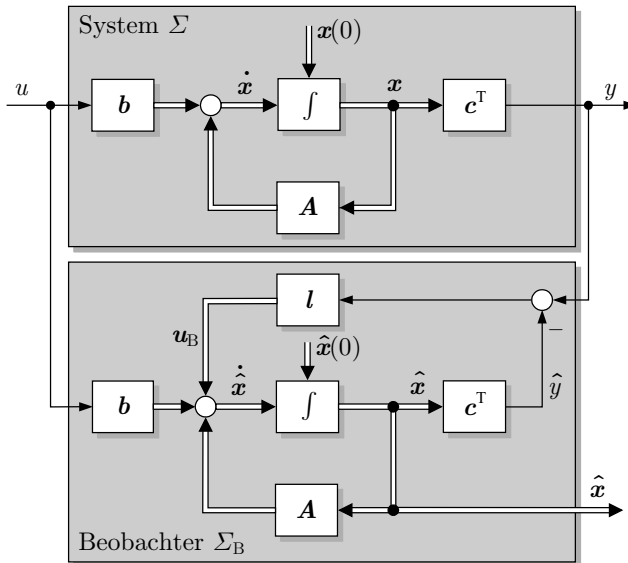


Abb. 9.3: Kopplung von System und Luenbergerbeobachter

Betrachtet wird ein nicht sprungfähiges System

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T \mathbf{x}(t). \end{cases} \quad (9.2)$$

Um das Modell als Beobachter einzusetzen, wird es um den  $n$ -dimensionalen Eingang  $\mathbf{u}_B(t)$  erweitert

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{b}u(t) + \mathbf{u}_B(t), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0 \quad (9.3)$$

$$\hat{y}(t) = \mathbf{c}^T \hat{\mathbf{x}}(t), \quad (9.4)$$

zu dem die Differenz zwischen der gemessenen Ausgangsgröße und der Ausgangsgröße des Modells zurückgeführt wird:

$$\mathbf{u}_B(t) = \mathbf{l}(y(t) - \hat{y}(t)). \quad (9.5)$$

Dabei ist  $\mathbf{l}$  ein  $n$ -dimensionaler Vektor, dessen  $i$ -tes Element bestimmt, mit welchem Gewicht die Differenz  $y(t) - \hat{y}(t)$  die Ableitung  $\dot{\hat{x}}_i(t)$  der  $i$ -ten Zustandsvariablen des Modells beeinflusst. Modell und Rückführung zusammen stellen den Luenbergerbeobachter dar (Abb. 9.3). Er wird durch die Gln. (9.2), (9.3) und (9.5) beschrieben, die man zu

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{b}u(t) + \mathbf{l}(y(t) - \mathbf{c}^T \hat{\mathbf{x}}(t)) \quad (9.6)$$

und

Beobachter  $\Sigma_B$ :  $\dot{\hat{\mathbf{x}}}(t) = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) \hat{\mathbf{x}}(t) + \mathbf{b}u(t) + \mathbf{l}y(t), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0$

(9.7)

umformen kann. Gleichung (9.7) zeigt, dass der Beobachter ein dynamisches System mit *zwei* Eingängen ist,  $u(t)$  und  $y(t)$ .

Solange der Beobachteraussgang  $\hat{y}(t)$  mit dem Systemausgang  $y(t)$  übereinstimmt ( $\hat{y}(t) = y(t)$ ), ist die Rückführung wirkungslos ( $\mathbf{u}_B(t) = \mathbf{0}$ ) und Gl. (9.7) reduziert sich auf

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{b}u(t),$$

womit die Beobachterstruktur nach Abb. 9.1 genauso wie die Parallelschaltung in Abb. 9.2 arbeitet. Tritt eine Differenz zwischen den Zustandsvektoren  $\mathbf{x}(t)$  und  $\hat{\mathbf{x}}(t)$  auf, die sich in einer Differenz der Ausgangsgrößen bemerkbar macht, wird der Beobachterzustand  $\hat{\mathbf{x}}(t)$  durch die Rückführung beeinflusst. Der Vektor  $\mathbf{l}$  muss nun so gewählt werden, dass der Term  $\mathbf{l}(y(t) - \mathbf{c}^T \hat{\mathbf{x}}(t))$  die Differenz  $\mathbf{x}(t) - \hat{\mathbf{x}}(t)$  verkleinert.



### 9.2.3 Wahl der Beobachterrückführung

Um einen geeigneten Vektor  $\mathbf{l}$  zu finden, wird der Beobachtungsfehler

$$\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t) \quad (9.8)$$

eingeführt. Mit Hilfe der Gleichungen (9.2) und (9.7) erhält man für diesen Fehler die Differentialgleichung

$$\begin{aligned} \dot{\mathbf{e}}(t) &= \dot{\mathbf{x}}(t) - \dot{\hat{\mathbf{x}}}(t) \\ &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) - (\mathbf{A} - \mathbf{l}\mathbf{c}^T)\hat{\mathbf{x}}(t) - \mathbf{b}u(t) - \mathbf{l}y(t) \\ &= (\mathbf{A} - \mathbf{l}\mathbf{c}^T)(\mathbf{x}(t) - \hat{\mathbf{x}}(t)) \end{aligned}$$

und daraus

$$\Sigma_e : \quad \dot{\mathbf{e}}(t) = (\mathbf{A} - \mathbf{l}\mathbf{c}^T)\mathbf{e}(t), \quad \mathbf{e}(0) = \mathbf{x}_0 - \hat{\mathbf{x}}_0. \quad (9.9)$$

Der Beobachtungsfehler kann also als Zustand eines autonomen Systems  $\Sigma_e$  gedeutet werden. Ist dieses System asymptotisch stabil, so klingt der Beobachtungsfehler ab und der Beobachterzustand  $\hat{\mathbf{x}}(t)$  nähert sich asymptotisch dem Systemzustand  $\mathbf{x}(t)$  an.

#### Satz 9.1 (Beobachter)

Für den Beobachtungsfehler

$$\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$$

eines Luenbergerbeobachters gilt die Beziehung

$$\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| = 0$$

für beliebige Anfangszustände des Systems und des Beobachters genau dann, wenn alle Eigenwerte der Matrix  $(\mathbf{A} - \mathbf{l}\mathbf{c}^T)$  negativen Realteil haben.

Die Aufgabe, der Matrix  $(\mathbf{A} - \mathbf{l}\mathbf{c}^T)$  durch die geeignete Wahl des Vektors  $\mathbf{l}$  Eigenwerte mit negativem Realteil zuzuweisen, ist eine Standardaufgabe der Regelungstechnik, deren Lösung hier für den Beobachterentwurf angegeben wird. Dabei ist zunächst zu erwähnen, dass man für vollständig beobachtbare Systeme der Matrix  $\mathbf{A} - \mathbf{l}\mathbf{c}^T$  nicht nur Eigenwerte mit negativem Realteil, sondern sogar beliebig ausgewählte Eigenwerte zuweisen kann. Nennt man diese Eigenwerte  $\lambda_{Bi}$  ( $i = 1, 2, \dots, n$ ), so kann man aus ihnen entsprechend

$$\begin{aligned} \prod_{i=1}^n (\lambda - \lambda_{Bi}) &= \lambda^n + a_{Bn-1}\lambda^{n-1} + \dots + a_{B1}\lambda + a_{B0} \\ &\stackrel{!}{=} \det(\lambda\mathbf{I} - (\mathbf{A} - \mathbf{l}\mathbf{c}^T)) \end{aligned} \quad (9.10)$$

das charakteristische Polynom ausrechnen, das die Matrix  $(\mathbf{A} - \mathbf{l}\mathbf{c}^T)$  haben soll. Die letzte Zeile ist eine Forderung, die durch die geeignete Wahl der Beobachterrückführung  $\mathbf{l}$  erfüllt werden soll.

Um die Rückführung  $\mathbf{l}$  zu ermitteln, mit der die mit dem Ausrufezeichen gekennzeichnete Forderung erfüllt ist, bildet man zunächst die letzte Spalte  $\mathbf{s}_B$  der inversen Beobachtbarkeitsmatrix

$$\mathbf{s}_B = \mathbf{S}_B^{-1} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \quad (9.11)$$

mit  $\mathbf{S}_B$  aus Gl. (5.13) auf S. 163. Den Vektor  $\mathbf{l}$  erhält man dann aus der Beziehung

Beobachterrückführung:  $\mathbf{l} = (\mathbf{s}_B \quad \mathbf{A}\mathbf{s}_B \quad \dots \quad \mathbf{A}^n \mathbf{s}_B) \begin{pmatrix} a_{B0} \\ a_{B1} \\ \vdots \\ a_{Bn-1} \\ 1 \end{pmatrix}.$

(9.12)

Der erste Faktor ist eine  $(n \times n + 1)$ -Matrix, der zweite ein  $(n + 1)$ -dimensionaler Vektor mit den Koeffizienten des charakteristischen Polynoms (9.10).

Für den Entwurf der Rückführung  $\mathbf{l}$  sind zwei Tatsachen wichtig:

- Den Eigenwerten der Matrix  $(\mathbf{A} - \mathbf{l}\mathbf{c}^T)$  können durch eine geeignete Wahl von  $\mathbf{l}$  genau dann beliebige Werte zugewiesen werden, wenn das System  $(\mathbf{A}, \mathbf{c}^T)$  vollständig beobachtbar ist, was beim Beobachterentwurf natürlich vorausgesetzt wird.
- Damit der Beobachtungsfehler schneller abklingt als die Eigenbewegung des zu beobachtenden Systems, müssen die Eigenwerte der Matrix  $\mathbf{A} - \mathbf{l}\mathbf{c}^T$  möglichst weit links in der komplexen Ebene platziert werden. Da die Elemente des Vektors  $\mathbf{l}$  umso größer werden, je weiter die Beobachtereigenwerte  $\lambda_{Bi}$  von den Eigenwerten  $\lambda_i$  der Matrix  $\mathbf{A}$  entfernt liegen, gibt man den Beobachtereigenwerten nicht einen beliebig kleinen negativen Realteil, sondern legt sie in der komplexen Ebene etwas links von den dominierenden (größten) Eigenwerten von  $\mathbf{A}$ .

**Berechnung der Beobachterrückführung aus der Beobachtungsnormform.** Auf eine besonders einfache Berechnungsvorschrift für die Beobachterrückführung kommt man, wenn das Modell des zu beobachtenden Systems  $\Sigma$  in Beobachtungsnormform

$$\Sigma : \begin{cases} \dot{\mathbf{x}}_B(t) = \mathbf{A}_B \mathbf{x}_B(t) + \mathbf{b}_B u(t), & \mathbf{x}_B(0) = \mathbf{x}_{B0} \\ y(t) = \mathbf{c}_B^T \mathbf{x}_B(t) \end{cases} \quad (9.13)$$

vorliegt, in der die Systemmatrix und der Ausgabevektor die spezielle Form

$$\mathbf{A}_B = \begin{pmatrix} 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & \cdots & 0 & -a_2 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -a_{n-1} \end{pmatrix} \quad \text{und} \quad \mathbf{c}_B^T = (0 \ 0 \ \dots \ 1) \quad (9.14)$$

haben. Der Vektor

$$\mathbf{l}_B = \begin{pmatrix} l_{B1} \\ l_{B2} \\ \vdots \\ l_{Bn} \end{pmatrix}$$

der für das Zustandsraummodell in Beobachtungsnormalform bestimmten Beobachterrückführung muss dann so gewählt werden, dass die Matrix

$$\mathbf{A}_B - \mathbf{l}_B \mathbf{c}_B^T = \begin{pmatrix} 0 & 0 & 0 & \cdots & -a_0 - l_{B1} \\ 1 & 0 & 0 & \cdots & -a_1 - l_{B2} \\ 0 & 0 & 1 & \cdots & -a_2 - l_{B3} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & -a_{n-1} - l_{Bn} \end{pmatrix}$$

die vorgegebenen Beobachtereigenwerte  $\lambda_{Bi}$ , ( $i = 1, \dots, n$ ) besitzt.

Bestimmt man entsprechend Gl. (9.10) aus den vorgegebenen Eigenwerten  $\lambda_{Bi}$  die Koeffizienten  $a_{Bi}$  des zugehörigen charakteristischen Polynoms, so muss dieses Polynom mit dem charakteristischen Polynom der hier angegebenen Matrix  $\mathbf{A}_B - \mathbf{l}_B \mathbf{c}_B^T$  übereinstimmen, für das man die Beziehung

$$\det(\lambda \mathbf{I} - (\mathbf{A}_B - \mathbf{l}_B \mathbf{c}_B^T)) = \lambda^n + (a_{n-1} + l_{Bn})\lambda^{n-1} + \dots + (a_1 + l_{B2})\lambda + (a_0 + l_{B1})$$

erhält. Aus der Gleichheit beider Polynome

$$\begin{aligned} & \lambda^n + a_{Bn-1}\lambda^{n-1} + \dots + a_{B1}\lambda + a_{B0} = \\ & \lambda^n + (a_{n-1} + l_{Bn})\lambda^{n-1} + \dots + (a_1 + l_{B2})\lambda + (a_0 + l_{B1}) \end{aligned}$$

folgt für den Vektor  $\mathbf{l}_B$  die folgende Beziehung:

Beobachterrückführung für Systeme in Beobachtungsnormalform:

$$\mathbf{l}_B^T = (a_{B0}, a_{B1}, \dots, a_{Bn-1}) - (a_0, a_1, \dots, a_{n-1}).$$

(9.15)

Die Beobachterrückführung erhält man also aus der Differenz der Koeffizienten des gewünschten charakteristischen Polynoms des Beobachters und des charakteristischen Polynoms des zu beobachtenden Systems. Diese einfache Berechnungsvorschrift ist der Grund für die Bezeichnung „Beobachtungsnormalform“ für Modelle, für die die Gl. (9.14) gilt.

Aus dieser Beziehung sieht man, dass die Elemente des Vektors  $\mathbf{l}_B$  umso größer werden, je mehr sich die Beobachtereigenwerte von den Eigenwerten der Matrix  $\mathbf{A}_B$  unterscheiden, denn damit steigt auch die Differenz der Koeffizienten beider charakteristischer Polynome.

Übrigens kann jedes vollständig beobachtbare System (9.2) durch die Zustandstransformation

$$\mathbf{x}_B(t) = \mathbf{T}_B^{-1} \mathbf{x}(t)$$

in die Beobachtungsnormalform (9.13) überführt werden. Die Transformationsmatrix  $\mathbf{T}_B$  bildet man aus dem in Gl. (9.11) eingeführten Vektor  $\mathbf{s}_B$  und der Systemmatrix  $\mathbf{A}$  folgendermaßen:

$$\mathbf{T}_B = (\mathbf{s}_B \quad \mathbf{A} \mathbf{s}_B \quad \dots \quad \mathbf{A}^{n-1} \mathbf{s}_B). \quad (9.16)$$

Die Elemente des Modells (9.13) erhält man dann aus den Beziehungen

$$\begin{aligned} \mathbf{A}_B &= \mathbf{T}_B^{-1} \mathbf{A} \mathbf{T}_B \\ \mathbf{b}_B &= \mathbf{T}_B^{-1} \mathbf{b} \\ \mathbf{c}_B^T &= \mathbf{c}^T \mathbf{T}_B \\ \mathbf{x}_{B0} &= \mathbf{T}_B^{-1} \mathbf{x}_0. \end{aligned}$$

**Entwurfsalgorithmus.** Der Beobachterentwurf lässt sich in folgendem Algorithmus zusammenfassen:

---

#### Algorithmus 9.1 Beobachterentwurf

---

**Gegeben:** Lineares System (9.2).

1. Überprüfen Sie mit Hilfe des im Satz 5.2 auf S. 163 beschriebenen Kriteriums die Beobachtbarkeit des Systems. Ein Beobachter kann nur für vollständig beobachtbare Systeme entworfen werden.
2. Wählen Sie  $n$  Beobachtereigenwerte  $\lambda_{Bi}$ .
3. Berechnen Sie die Beobachterrückführung  $\mathbf{l}$  aus Gl. (9.12) bzw. für Modelle in Beobachtungsnormalform aus Gl. (9.15).

**Ergebnis:** Beobachter (9.7).

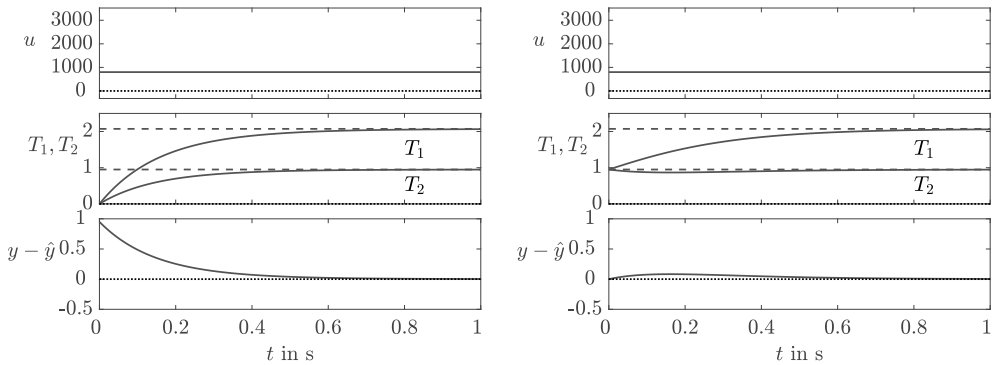
---

#### Beispiel 9.2 Beobachter für einen Industrieofen

Für den Industrieofen aus Beispiel 5.1 auf S. 158 soll ein Beobachter entworfen werden, mit dem aus der gemessenen Temperatur  $T_2(t)$  des Werkstückhalters die Werkstücktemperatur  $T_1(t)$  berechnet werden kann.

Die Eigenwerte der Matrix  $\mathbf{A}$  des Zustandsraummodells (5.11) auf S. 161 liegen bei  $-4,82$  und  $-0,92$ . Für den Beobachter werden deshalb die Eigenwerte  $-5$  und  $-7,5$  gewählt, die kleiner als die Ofeneigenwerte sind. Aus Gl. (9.7) erhält man dafür die Beobachterrückführung

$$\mathbf{l} = \begin{pmatrix} 10,09 \\ 6,75 \end{pmatrix}. \quad (9.17)$$



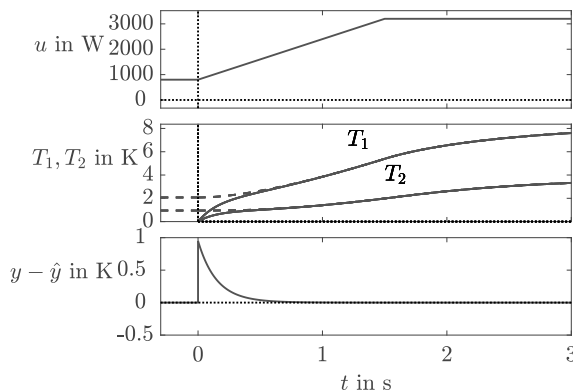
**Abb. 9.4:** Einschwingen des Beobachters bei konstanten Ofentemperaturen bei zwei unterschiedlichen Anfangszuständen des Beobachters:  $\hat{x}_0 = \mathbf{0}$  (links) und  $\hat{x}_0 = (y(0), y(0))^T$  (rechts); Temperaturen in Kelvin, Eingangsgröße in Watt

Abbildung 9.4 zeigt, wie der Beobachter auf die richtigen Temperaturen einschwingt. In diesem wie in den folgenden Abbildungen sind die Beobachtersignale durch die durchgezogenen Linien und die Signale am System durch gestrichelte Linien dargestellt. Der Ofen wird mit konstanter Eingangsgröße bei konstanten Temperaturen  $T_1(t)$  und  $T_2(t)$  betrieben. Im linken Teil der Abbildung ist das Verhalten des Beobachters für den Anfangszustand  $\hat{x}_0 = \mathbf{0}$  dargestellt. Der Übergangsvorgang nach etwa 0,5 Sekunden abgeschlossen und der Beobachter liefert danach (fast) die exakten Temperaturen.

Da der Beobachter die zweite Zustandsvariable als Messgröße erhält, ist es sinnvoll, den Anfangszustand des Beobachters aus dem Messwert abzuleiten:

$$\hat{x}_0 = \begin{pmatrix} y(0) \\ y(0) \end{pmatrix} = \begin{pmatrix} 0,95 \\ 0,95 \end{pmatrix}.$$

Der Beobachter schwingt wieder innerhalb von 0,5 Sekunden ein, wobei aber der Beobachtungsfehler  $y(t) - \hat{y}(t)$  viel kleiner ist.



**Abb. 9.5:** Beobachtung der Werkstücktemperatur bei zeitveränderlicher Eingangsgröße

Abbildung 9.5 zeigt das Verhalten des Beobachters, der zur Zeit  $t = 0$  mit verschwindendem Anfangszustand eingeschaltet wird, wenn der Ofen die oben dargestellte Eingangsgröße erhält. Nach etwa 0,5 Sekunden hat sich der Beobachterzustand an die wahren Temperaturen angeglichen. Das Beobachtungsergebnis folgt anschließend dem Temperaturverlauf exakt.

**Diskussion.** Die Dauer des Einschwingens des Beobachters ist in allen Fällen dieselbe, denn sie wird durch die gewählten Beobachtereigenwerte bestimmt. Zum größeren der beiden Eigenwerte gehört die Zeitkonstante  $T = \frac{1}{\delta s} = 0,2$  s. Die aus den Abbildungen ermittelte Einschwingzeit von 0,5 Sekunden entspricht also etwa dem Dreifachen dieser Zeitkonstanten.

Wenn sich bei konstanter Eingangsgröße  $u(t) = \bar{u}$  der statische Zustand eingestellt hat, braucht man keinen Beobachter, denn man kann alle Zustandsvariablen aus der Beziehung

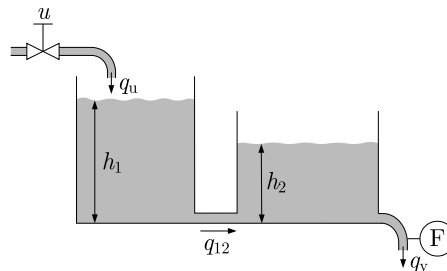
$$\bar{x} = -A^{-1}b\bar{u}$$

berechnen. Der eigentliche Wert eines Beobachters liegt darin, dass man mit ihm den Systemzustand während eines Übergangsvorgangs bestimmen kann, wie es Abb. 9.5 zeigt.  $\square$

#### Aufgabe 9.1\* Entwurf eines Beobachters für zwei gekoppelte Vorratsbehälter

Betrachten Sie das in Abb. 9.6 dargestellte System aus zwei gekoppelten Vorratsbehältern mit folgenden Werten für die Querschnitte der beiden Behälter sowie die Durchflusskoeffizienten der beiden Rohrleitungen:

$$\begin{aligned} A_1 &= 10 \text{ m}^2 & k_{12} &= 1 \frac{\text{m}^2}{\text{s}} \\ A_2 &= 12 \text{ m}^2 & k_y &= 0,5 \frac{\text{m}^2}{\text{s}}. \end{aligned}$$



**Abb. 9.6:** Behältersystem, für das ein Beobachter entworfen werden soll

Entwerfen Sie einen Beobachter in folgenden Schritten:

1. Stellen Sie ein lineares Zustandsraummodell für das System auf, bei dem Sie von einem linearen Zusammenhang zwischen dem Füllstand und dem Ausfluss ausgehen.
2. Untersuchen Sie, ob für dieses System ein Beobachter entworfen werden kann.
3. Geben Sie einen Beobachter an, der aus der Messung des Auslaufs des rechten Behälters die beiden Füllstände rekonstruiert.  $\square$

### 9.2.4 Verhalten des Beobachters bei Störungen und Modellunsicherheiten

Im Satz 9.1 ist festgehalten, dass für das ungestörte System der Beobachtungsfehler asymptotisch verschwindet, wenn die Beobachterrückführung  $l$  so gewählt wird, dass der Beobachter mit der Systemmatrix  $(A - lc^T)$  stabil ist. In diesem Abschnitt wird untersucht, inwieweit das Beobachtungsergebnis durch Störungen und Modellunsicherheiten beeinflusst wird.

Die Störgröße  $d(t)$  wirkt auf das zu beobachtende System und verändert dessen Zustand  $x(t)$ . Der Beobachterzustand  $\hat{x}(t)$  kann diesen Änderungen nur insofern folgen, als dass die Wirkung der Störung im Systemausgang  $y(t)$  erkennbar ist und über diesen das Verhalten des Beobachters beeinflusst.

Zur Beschreibung des Störverhaltens des zu beobachtenden Systems  $\Sigma$  wird das Modell (9.2) um die Störung erweitert:

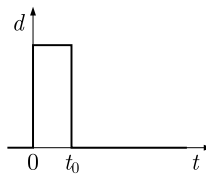
$$\Sigma : \begin{cases} \dot{x}(t) = Ax(t) + bu(t) + gd(t), & x(0) = x_0 \\ y(t) = c^T x(t). \end{cases}$$

Damit erweitert sich auch die Fehlergleichung (9.9) zu

$$\Sigma_e : \dot{e}(t) = (A - lc^T)e(t) + gd(t), \quad e(0) = x_0 - \hat{x}_0. \quad (9.18)$$

Bei der Analyse des Beobachtungsfehlers muss zwischen impulsförmigen und sprungförmigen Störungen unterschieden werden.

**Impulsförmige Störungen.** Wenn die Störung aus einem oder mehreren kurzen Impulsen besteht, so beeinflusst sie das Systemverhalten nur kurzzeitig und man kann erwarten, dass der Beobachtungsfehler zwischen den Impulsen abklingt, wenn diese weit genug auseinander liegen. Um diese Vermutung zu untersuchen, wird zunächst angenommen, dass die Störung nur aus einem kurzen Impuls besteht, der zur Zeit  $t = 0$  auftritt und bis zur Zeit  $t_0$  andauert (Abb. 9.7).



**Abb. 9.7:** Impulsförmige Störung

Für die Berechnung des Beobachtungsfehlers aus Gl. (9.18) kann man sich die Überlegung aus Beispiel 4.1 auf S. 135 zu Nutze machen und den Beobachtungsfehler  $e(t)$  für den Zeitraum  $t \geq t_0$  als eine Eigenbewegung des Systems (9.18) vom Zustand  $e(t_0)$  aus darstellen:

$$e(t) = e^{(A - lc^T)(t - t_0)} e(t_0) \quad \text{für } t \geq t_0.$$

Eine impulsförmige Störung kann also wie eine Veränderung der Anfangsbedingung von  $e(0)$  in  $e(t_0)$  interpretiert werden. Da ihr Einfluss auf das Verhalten des Beobachtungsfehlers nach

dem Zeitpunkt  $t_0$  durch  $e(t_0)$  beschrieben wird, sind für das Abklingen des Beobachtungsfehlers die Eigenwerte der Matrix  $(\mathbf{A} - \mathbf{l}\mathbf{c}^T)$  maßgebend. Unter den bekannten Bedingungen (Satz 9.1 auf S. 267) klingt der Beobachtungsfehler ab. Eine impulsförmige Störung hat also für  $t \gg t_0$  keinen Einfluss auf das Beobachtungsergebnis.

Diese Überlegungen können auch angewendet werden, wenn mehrere Störimpulse auftreten, diese aber zeitlich weit auseinander liegen. Dann ist der Einfluss des letzten Störimpulses bereits abgeklungen, bevor der nächste Störimpuls den Beobachtungsfehler wieder anhebt. Die Wirkung des Störimpulses wird durch die Veränderung des Systemzustands beschrieben, die der Störimpuls bis zum Ende seiner Impulsdauer hervorgerufen hat. Da der Beobachter diese Veränderung des Systemzustands nicht mitmacht und von ihr nur über eine veränderte Messgröße  $y(t)$  erfährt, steigt auch der Beobachtungsfehler an, klingt aber in der Zeit nach der Störung wieder ab.

Zusammenfassend zeigt diese Untersuchung, dass Beobachter impulsförmige Störungen tolerieren können, indem sie den dadurch entstehenden Beobachtungsfehler nach dem Einwirken eines Störimpulses wieder abbauen. Wie schnell dies geschieht, hängt von den gewählten Beobachtereigenwerten  $\lambda_{Bi}$  ab.

**Sprungförmige Störungen.** Es wird jetzt untersucht, was bei bleibenden Störungen passiert. Diese Störungen haben die Form

$$d(t) = \bar{d}\sigma(t),$$

wobei  $\sigma(t)$  die in Gl. (4.24) eingeführte Sprungfunktion ist. Die Fehlergleichung erhält damit eine konstante Eingangsgröße

$$\dot{e}(t) = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) e(t) + \mathbf{g}\bar{d},$$

auf Grund derer der Beobachtungsfehler nicht abklingt. Der bleibende Fehler

$$e(\infty) = \mathbf{x}(\infty) - \hat{\mathbf{x}}(\infty)$$

stellt sich bei  $\dot{e} = \mathbf{0}$  ein, wofür man

$$e(\infty) = -(\mathbf{A} - \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{g}\bar{d} \quad (9.19)$$

erhält. Da man  $\bar{d}$  nicht kennt kann man  $e(\infty)$  nicht berechnen. Allerdings kann man die bleibende Differenz  $y(\infty) - \hat{y}(\infty)$  der Ausgänge ermitteln, denn  $y(t)$  wird für alle Zeitpunkte  $t$  gemessen und  $\hat{y}(t)$  im Beobachter berechnet. Für diese Differenz gilt

$$y(\infty) - \hat{y}(\infty) = \mathbf{c}^T (\mathbf{x}(\infty) - \hat{\mathbf{x}}(\infty)) = \mathbf{c}^T e(\infty)$$

und

$$y(\infty) - \hat{y}(\infty) = -\mathbf{c}^T (\mathbf{A} - \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{g}\bar{d}.$$

Bezeichnet man den Faktor auf der rechten Seite mit  $k_{yd}$

$$k_{yd} = -\mathbf{c}^T (\mathbf{A} - \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{g}, \quad (9.20)$$

so kann man die letzte Gleichung einfacher als

$$y(\infty) - \hat{y}(\infty) = k_{yd}\bar{d} \quad (9.21)$$

schreiben. Diese Beziehung hat mehrere Konsequenzen:



- Eine sprungförmige Störung ruft typischerweise einen bleibenden Beobachtungsfehler hervor, der sich unabhängig von der Eingangsgröße  $u(t)$  asymptotisch dem in Gl. (9.19) angegebenen Wert  $e(\infty)$  nähert.
- Ein Maß für den (unbekannten) Beobachtungsfehler  $e(\infty)$  ist der bleibende Ausgangsfehler  $y(\infty) - \hat{y}(\infty)$ . Bei vielen Systemen gilt  $k_{yd} \neq 0$ , so dass der Beobachtungsfehler  $e(\infty)$  an einem bleibenden Ausgangsfehler erkennbar ist.
- Aus dem bleibenden Ausgangsfehler  $y(\infty) - \hat{y}(\infty)$  kann man mit Gl. (9.21) auf die Störamplitude  $\bar{d}$  schließen und daraus mit Gl. (9.19) den Beobachtungsfehler im eingeschwungenen Zustand berechnen:

$$e(\infty) = (A - lc^T)^{-1} g \frac{y(\infty) - \hat{y}(\infty)}{c^T (A - lc^T)^{-1} g}. \quad (9.22)$$

**Stochastische Störungen.** Störungen, die sich stochastisch um den Mittelwert null verändern, haben einen ständigen Einfluss auf den Beobachtungsfehler. Die Wirkung dieser Störungen hat jedoch auch den Mittelwert null, weil die Fehlergleichung (9.18) linear ist. Es ist offensichtlich, dass der durch die Störung hervorgerufene Fehler umso größer ist, je größer die Störamplitude (also die Varianz von  $d(t)$ ) ist.

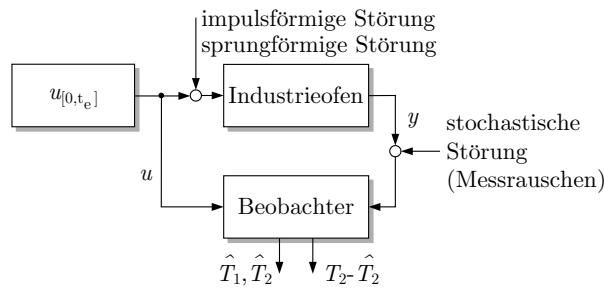
Hier soll keine genaue Analyse des Einflusses stochastischer Störungen vorgenommen werden, sondern nur auf die der Messung der Ausgangsgröße  $y(t)$  überlagerte Messstörung  $r(t)$  eingegangen werden. Diese Störung greift in Abb. 9.3 an der Summationsstelle an, an der die Differenz  $y(t) - \hat{y}(t)$  gebildet wird. Wie stark sie das Beobachtungsergebnis beeinflusst, hängt folglich von der Beobachterrückführung ab. Je größer die Elemente von  $l$  sind, desto stärker wirkt das Messrauschen auf den Zustand des Beobachters.

Da entsprechend Gl. (9.15) die Elemente des Vektors  $l$  umso größer werden, je mehr sich die Beobachtereigenwerte  $\lambda_{Bi}$  von denen des zu beobachtenden Systems unterscheiden, muss beim Beobachterentwurf ein Kompromiss zwischen der Einschwingzeit des Beobachters und der Störempfindlichkeit bezüglich des Messrauschens gefunden werden. Dabei muss man die Eigenwerte  $\lambda_{Bi}$  einerseits betragsmäßig möglichst groß machen, damit das Einschwingen im Vergleich zur Bewegung des zu beobachtenden Systems schnell genug ist. Andererseits dürfen die Eigenwerte nicht zu groß sein, um die Störempfindlichkeit zu begrenzen. Man begnügt sich deshalb häufig damit, die Eigenwerte  $\lambda_{Bi}$  betragsmäßig etwas größer als die dominierenden (d. h. betragsmäßig kleinsten) Eigenwerte der Matrix  $A$  zu wählen.

### Beispiel 9.3 Verhalten des Beobachters für den Industrieofen bei Störungen

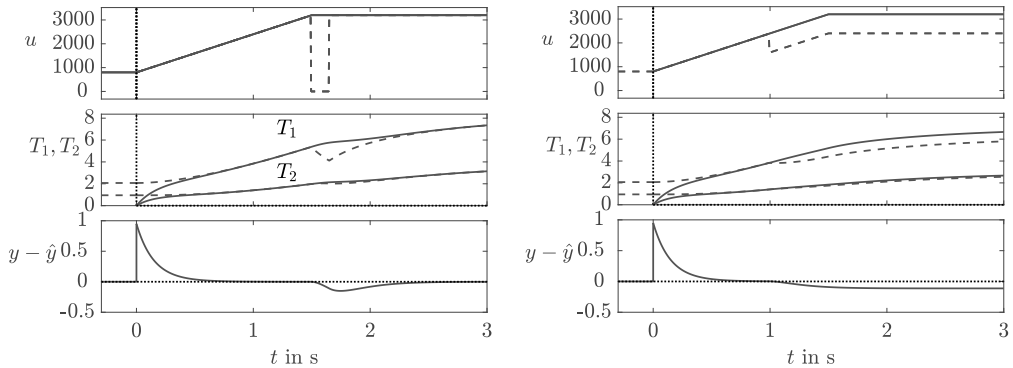
Der auf S. 270 beschriebene Beobachter wird jetzt unter der Wirkung von Störungen betrachtet (Abb. 9.8). Der linke Teil von Abb. 9.9 zeigt, wie sich eine kurzzeitige Unterbrechung der Heizspannung auf das Beobachtungsergebnis auswirkt. Der Beobachter arbeitet mit der für den Ofen vorgegebenen Eingangsgröße, die im obersten Abbildungsteil durch die durchgezogene Linie dargestellt ist. Die tatsächlich wirkende Heizleistung ist jedoch kurzzeitig gleich null, wie die gestrichelte Linie zeigt.

Auf Grund dieser Störung, die der Beobachter nicht kennt, fallen die beiden Temperaturen  $T_1(t)$  und  $T_2(t)$  und die Beobachtergrößen  $\hat{T}_1(t)$  und  $\hat{T}_2(t)$  liegen über den tatsächlichen Temperaturen. Deshalb steigt der Beobachtungsfehler  $|y(t) - \hat{y}(t)| = |T_2(t) - \hat{T}_2(t)|$  ab der Zeit  $t = 1,5$  s wieder an, nachdem



**Abb. 9.8:** Störungen bei der Beobachtung des Industrieofens

der Beobachter bereits eingeschwungen war. Da die Störung nach kurzer Zeit wieder verschwindet, kann der Beobachter wieder einschwingen, so dass der Beobachtungsfehler schnell wieder zu null wird.



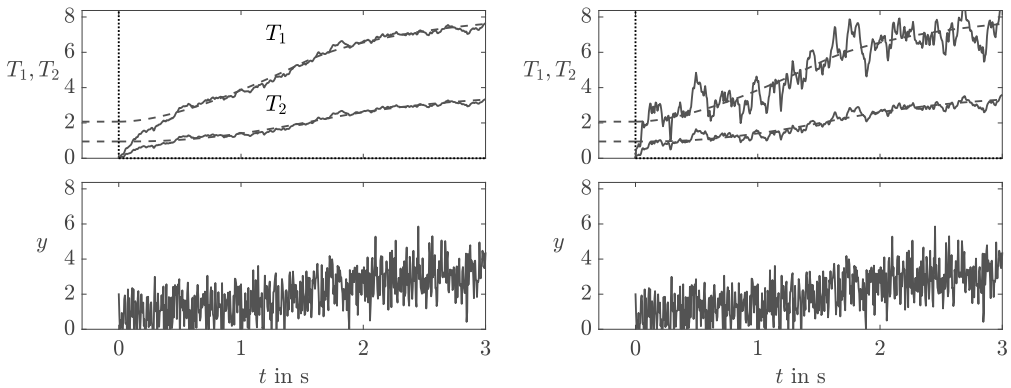
**Abb. 9.9:** Beobachtungsergebnis bei impulsförmiger Störung (links) und bei sprungförmiger Störung (rechts); Temperaturen in Kelvin, Eingangsgröße in Watt

Der rechte Teil von Abb. 9.9 zeigt das Verhalten des Beobachters bei einer sprungförmigen Störung, die zur Zeit  $t = 1$  s einsetzt und bestehen bleibt. Diese Störung bewirkt ein Absenken des von der Heizung ausgehenden Wärmestroms (gestrichelte Linie) gegenüber dem durch die Eingangsgröße  $u(t)$  geforderten Wert (durchgezogene Linie). Das Ergebnis ist ein bleibender Beobachtungsfehler, der sich in bleibenden Differenzen  $T_1(\infty) - \hat{T}_1(\infty)$  und  $T_2(\infty) - \hat{T}_2(\infty)$  zwischen den tatsächlichen und den mit dem Beobachter ermittelten Temperaturen äußert und den man an der bleibenden Differenz  $y(\infty) - \hat{y}(\infty)$  erkennen kann. Wie das Diagramm zeigt, ist die bleibende (unbekannte) Differenz  $T_1(\infty) - \hat{T}_1(\infty)$  der beobachteten Größe  $T_1(\infty)$  (obere Kurven im mittleren Teil der Abbildung) wesentlich größer als die bleibende Differenz der Ausgangsgrößen des Ofens und des Beobachters (untere Kurven).

Der bleibende Beobachtungsfehler kann mit Gl. (9.22) berechnet werden, für die hier aufgrund der am Eingang wirkenden Störung  $\mathbf{g} = \mathbf{b}$  gilt und die für die angegebenen Modellparameter auf

$$\begin{pmatrix} T_1(\infty) - \hat{T}_1(\infty) \\ T_2(\infty) - \hat{T}_2(\infty) \end{pmatrix} = \begin{pmatrix} 7,63 \\ 1 \end{pmatrix} 0,113 = \begin{pmatrix} 0,862 \\ 0,113 \end{pmatrix}$$

führt.



**Abb. 9.10:** Beobachtungsergebnis bei stochastischer Störung mit langsam (links) bzw. schnell (rechts) eingestelltem Beobachter; alle Temperaturen in Kelvin

Den Einfluss einer stochastischen Messstörung zeigt Abb. 9.10. Im unteren Abbildungsteil ist die verrauschte Messgröße zu sehen, die für die zwei betrachteten Beobachter dieselbe ist. Trotz dieser erheblichen Störung folgen die von den Beobachtern berechneten Temperaturen recht genau den tatsächlichen Temperaturen. Bemerkenswerterweise ist der Schätzwert  $\hat{T}_2(t)$  viel besser als der mit der Messstörung behaftete Messwert  $T_2(t)$ , wie man aus einem Vergleich der in gleichen Maßstäben dargestellten Kurven im oberen und unteren Teil von Abb. 9.10 erkennt. Der Beobachter wirkt bezüglich  $T_2(t)$  wie ein Filter, das die stochastischen Störungen mindert.

Eine Analyse des stochastisch gestörten Beobachters zeigt, dass man die Beobachtereigenwerte betragsmäßig nicht zu groß machen darf, weil der Beobachter sonst zu empfindlich auf Messstörungen reagiert. Um dies zu zeigen, wird im rechten Teil der Abbildung ein zweiter Beobachter betrachtet, dessen Eigenwerte mit  $-10$  und  $-15$  betragsmäßig größer als die des ersten Beobachters sind. Er hat die Beobachterrückführung

$$\mathbf{l} = \begin{pmatrix} 70,2 \\ 19,3 \end{pmatrix},$$

deren Elemente wesentlich größer als die des ersten Beobachters sind (vgl. Gl. (9.17)). Durch die neue Rückführung schwingt der Beobachter schneller ein, aber die Vergrößerung der Elemente des Vektors  $\mathbf{l}$  hat zur Folge, dass die Messstörung mit größerer Verstärkung auf den Beobachter aufgeschaltet wird (vgl. Signalweg von  $y(t)$  nach  $\mathbf{u}_B(t)$  in Abb. 9.3 auf S. 265). Deshalb beeinflusst die stochastische Störung das Beobachtungsergebnis jetzt viel stärker, wie ein Vergleich der beiden Teile von Abb. 9.10 zeigt. Die Beobachtungsergebnisse sind jetzt deutlich schlechter als beim ersten Beobachter. Es muss beim Beobachterentwurf also immer ein Kompromiss zwischen der Einschwingzeit des Beobachters und der Störempfindlichkeit gefunden werden.  $\square$

**Modellunbestimmtheiten.** Bisher wurde angenommen, dass der Beobachter mit einem Modell arbeitet, das den zu beobachtenden Prozess exakt beschreibt. Wenn jedoch für den durch die Gl. (9.2) beschriebenen Prozess ein Beobachter

$$\Sigma_B : \begin{cases} \dot{\hat{\mathbf{x}}}(t) = \hat{\mathbf{A}}\hat{\mathbf{x}}(t) + \hat{\mathbf{b}}u(t) + \mathbf{l}(y(t) - \hat{y}(t)), & \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0 \\ \hat{y}(t) = \hat{\mathbf{c}}^T \hat{\mathbf{x}}(t) \end{cases}$$

verwendet wird, dessen Modellparameter sich um

$$\begin{aligned} \delta \mathbf{A} &= \mathbf{A} - \hat{\mathbf{A}} \\ \delta \mathbf{b} &= \mathbf{b} - \hat{\mathbf{b}} \\ \delta \mathbf{c}^T &= \mathbf{c}^T - \hat{\mathbf{c}}^T \end{aligned}$$

von den Parametern des Prozesses unterscheiden, so erhält man an Stelle der Fehlergleichung (9.9) die Beziehung

$$\dot{e}(t) = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) e(t) + \delta \mathbf{b} u(t) + (\delta \mathbf{A} - \mathbf{l} \delta \mathbf{c}^T) \hat{\mathbf{x}}(t), \quad e(0) = \mathbf{x}_0 - \hat{\mathbf{x}}_0. \quad (9.23)$$

Der Beobachtungsfehler  $e(t)$  wird also nicht mehr als Zustand eines autonomen Systems beschrieben, sondern durch ein System mit den Eingängen  $u(t)$  und  $\hat{\mathbf{x}}(t)$ . Der Fehler  $e(t)$  kann nur verschwinden, wenn das beobachtete System nicht erregt wird ( $u(t) = 0$ ) und wenn es stabil ist, so dass  $\mathbf{x}(t) \rightarrow \mathbf{0}$  und somit  $\hat{\mathbf{x}}(t) \rightarrow \mathbf{0}$  gilt. In allen anderen Fällen tritt ein bleibender Beobachtungsfehler auf, der sich im Ausgangsfehler  $y(\infty) - \hat{y}(\infty)$  äußert.

### 9.3 Beobachter für nichtlineare Systeme

Die Idee des Luenbergerbeobachters kann für nichtlineare Systeme

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t)), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = h(\mathbf{x}(t)) \end{cases}$$

erweitert werden. Auch hier schaltet man das Modell parallel zum Prozess und führt die Differenz zwischen dem gemessenen Ausgang  $y(t)$  und dem Beobachteraussgang  $\hat{y}(t)$  auf das Modell zurück (vgl. Gln. (9.3) – (9.5) auf S. 266), so dass

$$\begin{aligned} \dot{\hat{\mathbf{x}}}(t) &= \mathbf{g}(\hat{\mathbf{x}}(t), u(t)) + \mathbf{u}_B(t), & \hat{\mathbf{x}}(0) &= \hat{\mathbf{x}}_0 \\ \hat{y}(t) &= h(\hat{\mathbf{x}}(t)), \end{aligned}$$

mit

$$\mathbf{u}_B(t) = \tilde{\mathbf{l}}(y(t) - \hat{y}(t))$$

gilt. In der letzten Gleichung bedeutet der Ausdruck  $\tilde{\mathbf{l}}(y(t) - \hat{y}(t))$  nicht wie bisher eine Multiplikation des Vektors  $\tilde{\mathbf{l}}$  mit dem Skalar  $y(t) - \hat{y}(t)$ , sondern  $\tilde{\mathbf{l}}(\cdot)$  ist eine nichtlineare Funktion, die dem Fehler  $y(t) - \hat{y}(t)$  zu jedem Zeitpunkt  $t$  einen  $n$ -dimensionalen Vektor  $\tilde{\mathbf{l}}(y(t) - \hat{y}(t))$  zuordnet. Die in Abb. 9.11 dargestellte Struktur ist deshalb der in Abb. 9.3 auf S. 265 gezeigten Struktur sehr ähnlich. Die linearen Zusammenhänge zwischen  $\mathbf{x}(t)$  und  $u(t)$  sowie  $\dot{\mathbf{x}}(t)$  und  $y(t)$  sind durch die entsprechenden nichtlinearen Beziehungen ersetzt. Für den Beobachter erhält man daraus die Gleichung

$$\Sigma_B : \dot{\hat{\mathbf{x}}}(t) = \mathbf{g}(\hat{\mathbf{x}}(t), u(t)) + \tilde{\mathbf{l}}(y(t) - h(\hat{\mathbf{x}}(t))), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0$$

mit der aus  $y(t)$  und  $u(t)$  der Näherungswert  $\hat{\mathbf{x}}(t)$  für den Systemzustand  $\mathbf{x}(t)$  berechnet wird.

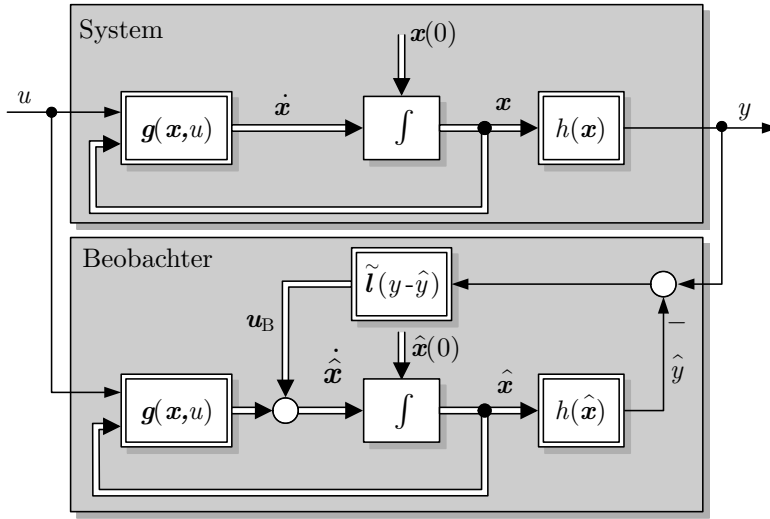


Abb. 9.11: Beobachter für ein nichtlineares System

Für den Beobachtungsfehler  $e(t) = x(t) - \hat{x}(t)$  gilt die nichtlineare Differentialgleichung

$$\Sigma_e : \begin{cases} \dot{e}(t) = g(e(t) + \hat{x}(t), u(t)) - g(\hat{x}(t), u(t)) + \tilde{l}(h(x(t)) - h(e(t) + \hat{x}(t))) \\ e(0) = x_0 - \hat{x}_0, \end{cases} \quad (9.24)$$

in der der Eingang  $u(t)$  und der Beobachterzustand  $\hat{x}(t)$  vorkommen. Es ist deshalb wesentlich schwieriger als beim linearen Beobachter, das Verhalten des Beobachtungsfehlers zu analysieren und Entwurfsrichtlinien abzuleiten, nach denen die nichtlineare Funktion  $\tilde{l}$  gewählt werden muss, damit der Beobachtungsfehler wie im linearen Fall asymptotisch verschwindet:  $\lim_{t \rightarrow \infty} \|e(t)\| \stackrel{!}{=} 0$ .

Die meisten Verfahren für den Entwurf nichtlinearer Beobachter verwenden deshalb eine Linearisierung. Wendet man das im Abschn. 3.5 behandelte Linearisierungsverfahren auf das Fehlersystem (9.24) an, so erhält man nach kurzer Rechnung die Beziehung

$$\Sigma_e : \dot{\hat{x}}(t) \approx \left( \frac{dg(x(t), u(t))}{dx} - \frac{d\tilde{l}(y)}{dy} \cdot \frac{dh(x, u)}{dx} \right) e(t), \quad e(0) = x_0 - \hat{x}_0. \quad (9.25)$$

Wenn man als neue Bezeichnungen

$$A = \frac{dg(x, u)}{dx}, \quad l = \frac{d\tilde{l}(y)}{dy}, \quad c^T = \frac{dh(x, u)}{dx}$$

eingführt, hat dieses Fehlersystem  $\Sigma_e$  dieselbe Gestalt wie Gl. (9.9).

Ein wichtiger Unterschied besteht jedoch darin, dass die Matrix  $A$  sowie die Vektoren  $l$  und  $c^T$  jetzt von der Zeit abhängen, denn der „Arbeitspunkt“  $e(t) = 0$  bedeutet, dass der Beobachterzustand dem Systemzustand folgt, also  $\hat{x}(t) = x(t)$  ist. Damit ändern sich die Argumente,

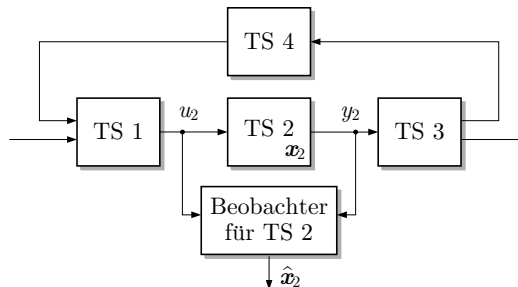
die in die Differentialquotienten einzusetzen sind und folglich der in der Klammer auf der rechten Seite der linearisierten Fehlergleichung stehende Ausdruck. Es ist deshalb schwierig, für nichtlineare Beobachter die Konvergenz der Fehlergleichung nachzuweisen und Kriterien für die Wahl der Beobachterrückführung anzugeben.

Die wichtigste Information dieses Abschnittes, der diese Analyse nicht im Einzelnen behandeln kann, ist also der Hinweis darauf, dass das Beobachterprinzip auch auf nichtlineare Systeme angewendet werden kann, dort aber auf schwierigere Entwurfsaufgaben führt.

## 9.4 Anwendungen der Zustandsbeobachtung

### 9.4.1 Beobachtung eines Teilsystems

In vielen praktischen Aufgabenstellungen bezieht sich die Zustandsbeobachtung nicht auf das gesamte betrachtete System, sondern auf ein Teilsystem. Abbildung 9.12 zeigt vier verkoppelte Teilsysteme, von denen der Beobachter nur den Zustand  $x_2(t)$  des Teilsystems 2 bestimmen soll. Obwohl die Teilsysteme stark zusammenhängend sind (vgl. Abschn. 2.4) kann das Beobachterprinzip allein auf das Teilsystem 2 angewendet werden, denn der in der Abbildung eingetragene Beobachter erhält die Eingangsgröße  $u_2(t)$  und die Ausgangsgröße  $y_2(t)$  dieses Teilsystems als Eingänge.



**Abb. 9.12:** Beobachter für das Teilsystem 2

Wie die Ableitung der Beobachtergleichungen gezeigt hat, muss für den Einsatz eines Beobachters für ein Teilsystem sowohl der Eingang als auch der Ausgang gemessen werden können. Ob diese Größen über weitere Teilsysteme miteinander verknüpft sind, ist dabei gleichgültig. Bei gekoppelten Systemen treten möglicherweise mehrere Teilsystemeingänge und -ausgänge auf. Die Beobachtergleichungen müssen dann auf die zu Vektoren zusammengefassten Größen  $u(t)$  und  $y(t)$  erweitert werden. Daraus erkennt man die folgenden Bedingungen, die für den Beobachtereinsatz an einem Teilsystem erfüllt sein müssen:

- Es müssen alle Eingangsgrößen des Teilsystems als Informationen für den Beobachter zur Verfügung stehen. Nicht gemessene Eingangsgrößen wirken wie Störungen und verfälschen das Beobachtungsergebnis.

- Es müssen so viele Ausgangsgrößen des Teilsystems zur Verfügung stehen, dass das Teilsystem durch diese Ausgangsgrößen vollständig beobachtbar ist.

Die zweite Forderung zeigt, dass bei Teilsystemen mit mehreren Ausgangsgrößen nicht alle Ausgangsgrößen dem Beobachter zur Verfügung stehen müssen. Wenn das Teilsystem 2 in Abb. 9.12 über zusätzliche Ausgänge mit den Teilsystemen 1 und 4 verbunden wäre, so müssten diese Ausgänge nicht für die Zustandsbeobachtung verwendet werden, wenn das Teilsystem 2 bereits über den Ausgang  $y_2(t)$  vollständig beobachtbar ist. Allerdings ist intuitiv klar, dass das Beobachtungsergebnis umso besser sein wird, je mehr Messinformationen der Beobachter erhält.

Wenn das Gesamtsystem nicht vollständig beobachtbar ist, kann man es in Teilsysteme zerlegen, von denen mindestens eines vollständig beobachtbar ist (Abschn. 5.5). Der Luenbergerbeobachter kann dann auf das vollständig beobachtbare Teilsystem angewendet werden, um dessen Zustand zu rekonstruieren.

#### 9.4.2 Online-Vorhersage des Systemverhaltens

Im Folgenden werden Automatisierungsaufgaben behandelt, in denen der von einem Beobachter gelieferte Näherungswert  $\hat{x}(t)$  für den Systemzustand weiter verarbeitet wird. Neben den dabei behandelten Aufgaben der Online-Prädiktion und der Regelung unter Verwendung eines Beobachters ist die Prozessdiagnose ein wichtiges Gebiet, dem nachfolgend ein eigenes Kapitel gewidmet ist.

Im Kap. 4 wurde gezeigt, wie man bei Kenntnis des Anfangszustands  $x_0$  und des Verlaufs der Eingangsgröße  $u_{[0,t_e]}$  den Verlauf der Ausgangsgröße  $y_{[0,t_e]}$  vorhersagen kann. Diese Aufgabe wird jetzt unter der Bedingung untersucht, dass sie online, also während der Prozess läuft, gelöst werden soll. Neben der offensichtlichen Tatsache, dass man bei der Vorhersage ein genügend schnelles Simulationssystem benötigt, um die Ergebnisse unter Echtzeitbedingungen rechtzeitig zu erhalten, verändert sich die Aufgabenstellung bezüglich der im Kap. 4 behandelten in drei Aspekten:

- Erstens ist nicht mehr der Anfangszustand  $x_0$  bekannt, denn man kann nur den Systemausgang messen.
- Zweitens unterliegt der Prozess Störungen, die nicht messbar sind, trotzdem aber bei der Vorhersage berücksichtigt werden müssen.
- Drittens ist die Vorhersageaufgabe von einem sich ständig verändernden aktuellen Zeitpunkt  $t$  ausgehend für einen konstanten Vorhersagehorizont  $h$  zu lösen, wobei sich beide Grenzen des Vorhersagezeitraumes  $[t, t + h]$  mit der aktuellen Zeit  $t$  verschieben.

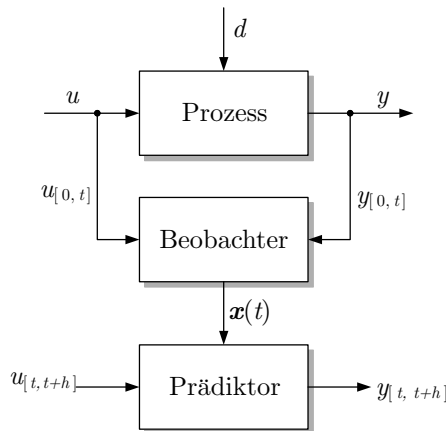
Bekannt ist der Verlauf des Eingangs  $u(t)$  und des Ausgangs  $y(t)$  seit dem Zeitpunkt 0. Für die Vorhersage muss dann noch der zukünftige Verlauf  $u_{[t,t_e]}$  der Eingangsgröße vorgegeben werden. Wenn dieser nicht exakt bekannt ist, können auch mehrere mögliche Verläufe für die Vorhersage genutzt werden. Man spricht in diesem Falle dann von einer Szenarienanalyse.

### Online-Vorhersage des Systemverhaltens

Gegeben: Zustandsraummodell des Systems  $\Sigma$   
 Verlauf der Eingangs- und Ausgangsgröße  $u_{[0,t]}, y_{[0,t]}$   
 Zukünftiger Verlauf der Eingangsgröße  $u_{[t,t+h]}$   
 Gesucht: Zukünftiger Verlauf der Ausgangsgröße  $y_{[t,t+h]}$

Unter den angegebenen Randbedingungen kann man die Vorhersageaufgabe in zwei Schritten lösen (Abb. 9.13):

1. **Beobachtung des Zustands  $x(t)$ :** Die Informationen  $u_{[0,t]}$  und  $y_{[0,t]}$  über den vergangenen Prozessablauf werden genutzt, um einen Näherungswert  $\hat{x}(t)$  des aktuellen Zustands  $x(t)$  zu bestimmen.
2. **Vorhersage des zukünftigen Systemverhaltens:** Vom aktuellen Zustand  $\hat{x}(t)$  ausgehend wird mit Hilfe der zukünftigen Eingangsgröße  $u_{[t,t+h]}$  der zukünftige Verlauf  $y_{[t,t+h]}$  der Ausgangsgröße berechnet.



**Abb. 9.13:** Online-Prädiktion mit Beobachter

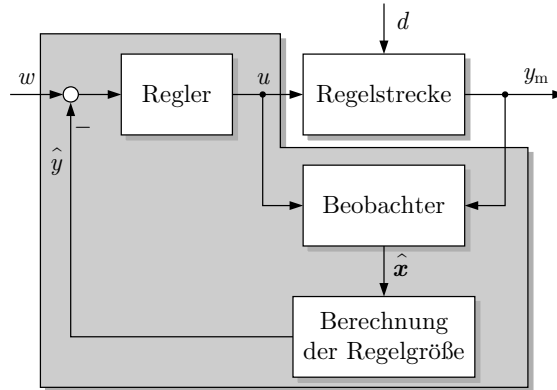
Die Verwendung des Beobachters zur Berechnung des aktuellen Zustands  $x(t)$  ist nicht nur deshalb notwendig, weil der aktuelle Zustand nicht messbar ist, sondern auch, um die Wirkung der unbekannten Störung  $d(t)$  zu berücksichtigen. Wenn man, wie es im Kap. 4 beschrieben wurde, von einem Anfangszustand  $x_0$  ausgehend das Systemverhalten aus der vorgegebenen Eingangsgröße  $u_{[0,t_e]}$  berechnet, so werden bei einem großen Zeitintervall  $[0, t_e]$  die Störung  $d(t)$  wie auch Modellunbestimmtheiten zu einem erheblichen Vorhersagefehler führen. Durch den Einsatz des Beobachters wird bei der Online-Prädiktion die in der Ausgangsgröße  $y_{[0,t]}$  steckende Information über den tatsächlichen Prozessablauf genutzt, um den aktuellen Zustand  $x(t)$ , der als Ausgangspunkt für die Vorhersage für den Zeitabschnitt  $[t, t+h]$  dient, dem tatsächlichen Systemzustand nachzuführen. Obwohl im Abschn. 9.2.4 gezeigt wurde, dass



bleibende Störungen und Modellunbestimmtheiten zu einem bleibenden Beobachtungsfehler  $\mathbf{x}(t) - \hat{\mathbf{x}}(t)$  führen, liegt der vom Beobachter unter diesen Bedingungen berechnete Zustand  $\hat{\mathbf{x}}(t)$  i. Allg. wesentlich näher am wirklichen Systemzustand  $\mathbf{x}(t)$  als das Ergebnis einer beim Zeitpunkt  $t = 0$  begonnenen Vorhersage, die die in  $y_{[0,t]}$  enthaltenen Informationen nicht nutzt.

### 9.4.3 Regelung unter Verwendung einer beobachteten Regelgröße

Im Standardregelkreis wird angenommen, dass die Regelgröße  $y(t)$  messbar ist, so dass die Regelabweichung  $e(t) = w(t) - y(t)$  aus einem Vergleich der Sollgröße  $w(t)$  und der Istgröße  $y(t)$  gebildet werden kann. Es gibt jedoch eine Reihe von Regelungsaufgaben, bei denen aus unterschiedlichen Gründen die Regelgröße  $y(t)$  nicht gemessen werden kann. So kann bei dem im Beispiel 9.2 betrachteten Industrieofen die Temperatur des Werkstücks nicht gemessen werden, obwohl sie als Regelgröße in einer Temperaturregelung für das Werkstück gebraucht wird.



**Abb. 9.14:** Realisierung einer Regelung mit Hilfe eines Beobachters für die nicht messbare Regelgröße  $y$

In diesen Fällen kann man einen Zustandsbeobachter einsetzen, um aus dem Verlauf der Stellgröße  $u(t)$  und einer messbaren Ausgangsgröße  $y_m(t)$  einen Näherungswert  $\hat{\mathbf{x}}(t)$  für den Systemzustand zu bestimmen und daraus mit der Ausgabegleichung

$$\hat{y}(t) = \mathbf{c}^T \hat{\mathbf{x}}(t) \quad (9.26)$$

einen Näherungswert für die Regelgröße  $y(t)$  zu berechnen. Der Beobachter muss dabei mit der Ausgabegleichung

$$y_m(t) = \mathbf{c}_m^T \mathbf{x}(t)$$

der messbaren Ausgangsgröße  $y_m$  gebildet werden.

Wie Abb. 9.14 zeigt, besteht der Regler dann aus drei Blöcken:

- dem Regler mit dem Eingang  $e(t) = w(t) - \hat{y}(t)$  und dem Ausgang  $u(t)$ ,
- dem Beobachter mit den beiden Eingängen  $u(t)$  und  $y_m(t)$  und dem Ausgang  $\hat{x}(t)$  sowie
- einem Block, in dem die Regelgröße entsprechend Gl. (9.26) aus dem Beobachtungsergebnis  $\hat{x}(t)$  berechnet wird.

Dies ist in jedem Fall ein dynamischer Regler, denn er enthält mit dem Beobachter eine Dynamik  $n$ -ter Ordnung. Der in der Abbildung grau unterlegte Teil hat wie jeder Regler die beiden Eingangsgrößen  $w(t)$  und  $y_m(t)$  sowie die Ausgangsgröße  $u(t)$ . Im Unterschied zu den bisher betrachteten Reglern tritt hier jedoch an die Stelle der Regelgröße  $y(t)$  die messbare Ersatzgröße  $y_m(t)$ .

### Aufgabe 9.2 Temperaturregelung eines Industrieofens

Bei dem im Beispiel 9.2 betrachteten Industrieofen soll die Werkstücktemperatur  $T_1(t)$  durch einen Regler auf einen vorgegebenen Sollwert  $w$  gebracht werden. Da die Werkstücktemperatur nicht messbar ist, muss sie unter Verwendung der gemessenen Temperatur  $T_2(t)$  des Werkstückhalters mit Hilfe eines Beobachters bestimmt werden. Um Sollwertfolge zu erreichen, soll ein I-Regler eingesetzt werden.

1. Nehmen Sie zunächst an, dass die Temperatur  $T_1(t)$  messbar ist, und stellen Sie das Zustandsraummodell des Regelkreises mit I-Regler auf. Zeigen Sie, dass die Regelgröße  $T_1$  bei diesem Regelkreis für beliebige Anfangszustände der Regelstrecke und des Reglers und beliebigem Sollwert  $w$  asymptotisch den Sollwert annimmt.
2. Erweitern Sie den Regelkreis um einen Beobachter, der die Regelgröße  $T_1(t)$  aus der Messgröße  $T_2(t)$  und der im Regler bekannten Stellgröße  $u(t)$  berechnet. Der Regler führt anstelle des wahren Wertes  $T_1(t)$  den vom Beobachter gelieferten Wert  $\hat{T}_1(t)$  zurück.
3. Untersuchen Sie, ob auch der Regelkreis mit Beobachter die Eigenschaft der Sollwertfolge besitzt? □

### Aufgabe 9.3 Zustandsbeobachtung eines Gleichstrommotors



Bei dem in Abb. 3.2 auf S. 87 gezeigten Gleichstrommotor wird der Strom  $i(t)$ , der im Zustandsraummodell (3.16) auf S. 90 als erste Zustandsvariable auftritt, typischerweise nicht gemessen. Entwerfen Sie einen Beobachter, der diesen Strom aus der bekannten Eingangsspannung  $u(t)$  und der gemessenen Drehfrequenz  $y(t)$  bestimmt.

Simulieren Sie das Verhalten des Motors für unterschiedliche Anfangszustände und Verläufe der Eingangsspannung und vergleichen Sie den tatsächlichen Verlauf des Stroms mit dem vom Zustandsbeobachter gelieferten Verlauf, wobei Sie die im Beispiel 3.4 auf S. 96 angegebenen Motorparameter verwenden. Wie gut ist das Beobachtungsergebnis, wenn der Motor ein zusätzliches, im Modell des Beobachters nicht berücksichtigtes Drehmoment (Störung) erhält. Wie beeinflussen Veränderungen der Motorparameter, beispielsweise die Veränderung des Trägheitsmoments  $J$ , das Beobachtungsergebnis? □

## Literaturhinweise

Die Idee der Zustandsbeobachtung geht auf LUENBERGER zurück, der den hier behandelten Beobachter in [66] eingeführt hat. Beobachter werden in Lehrbüchern wie z. B. [70] heute meist im Zusammenhang mit Regelungen betrachtet, die nicht nur die Regelgröße  $y$ , sondern den gesamten Zustandsvektor  $\mathbf{x}$  auf die Stellgröße  $u$  zurückführen und sich bei der Ermittlung des aktuellen Zustands eines Beobachters bedienen.

Eine wichtige Erweiterung der hier behandelten Grundidee der Zustandsbeobachtung führt auf reduzierte Beobachter, deren dynamische Ordnung  $n - p$  um die Anzahl  $p$  der Ausgangsgrößen kleiner ist als die dynamische Ordnung des zu beobachtenden Systems [70].

Eine kurze Übersicht über Beobachter für nichtlineare Systeme und einige Entwurfsverfahren findet man in [11] und [90].



# 10

## Diagnose kontinuierlicher Systeme

*Bei der modellbasierten Diagnose werden die gemessenen Eingangs- und Ausgangsgrößen mit einem dynamischen Modell verglichen und es wird aus Abweichungen zwischen dem Modell- und dem Systemverhalten auf Fehler geschlossen. Dieses Kapitel zeigt, wie man diese Aufgabe mit Methoden der Zustandsbeobachtung lösen kann.*

### 10.1 Diagnoseaufgabe und Lösungswege

#### 10.1.1 Diagnoseaufgabe

Durch die Diagnose sollen die in einem Prozess möglicherweise aufgetretenen Fehler gefunden werden. Dafür werden die Messgrößen  $u(t)$  und  $y(t)$  mit einem Prozessmodell verglichen und Abweichungen für die Fehlererkennung genutzt.

Die im Folgenden beschriebene Aufgabe und der prinzipielle Lösungsweg sind nicht für kontinuierliche Systeme spezifisch, sondern können auch für diskrete Systeme eingesetzt werden. Die Diagnosemethoden für beide Systemklassen unterscheiden sich aber in den mathematischen Methoden, mit denen dieser Lösungsweg realisiert wird (vgl. Kap. 16).

Diagnoseaufgaben können unter sehr vielfältigen Randbedingungen gestellt werden, wobei man Tests mit der betrachteten Anlage durchführen oder die Anlage zum Zweck der Fehlerdiagnose sogar demontieren kann. Im Gebiet der Automatisierungstechnik beziehen sich die Betrachtungen zur Diagnose jedoch vorrangig auf Aufgaben, bei denen Fehler während des

Betriebs der Anlage mit Hilfe der gemessenen Signale ermittelt werden sollen. Wenn man betonen will, dass weder spezielle Tests noch die Veränderung der Anlage zum Zweck der Diagnose zugelassen sind und dass die Diagnose unter Echtzeitbedingungen mit den zur Verfügung stehenden Messsignalen durchgeführt werden muss, verwendet man auch den Begriff *Prozessdiagnose*.

#### Beispiel 10.1 Diagnose von Kraftfahrzeugen

Bei modernen Fahrzeugen werden viele Komponenten während des Betriebs überwacht. In der Fahrzeugindustrie spricht man dabei von einer Onboard-Diagnose, weil die Diagnose „an Bord“ des Fahrzeugs erfolgt. Die Onboard-Diagnose wertet die gemessenen Signale unter Verwendung eines Modells aus. Dafür können die hier behandelten Methoden der Prozessdiagnose eingesetzt werden.

Im Gegensatz dazu greift man bei der Werkstattdiagnose auch auf Tests zurück, durch die man Vermutungen über Fehler bestätigen oder widerlegen kann. Wenn diese Tests nicht zum Erfolg führen, werden Komponenten gegen fehlerfreie getauscht oder zusätzliche Informationen durch eine direkte Inspektion der Komponenten beschafft. Diese Vorgehensweise stützt sich nicht ausschließlich auf Online-Messwerte und gehört deshalb nicht zur Prozessdiagnose.

Die Fahrzeugdiagnose greift auf beide Formen der Diagnose zurück, wobei die Onboard-Diagnose einen aufgetretenen Fehler soweit einschränkt, dass die Werkstattdiagnose mit diesen Vorinformationen schneller die auszutauschende Komponente findet. □

**Fehler in technischen Systemen.** Was ein Fehler in einer technischen Anlage ist, kann in Abhängigkeit vom Anwendungsfall sehr verschieden sein. Eine sehr allgemeine Definition heißt:

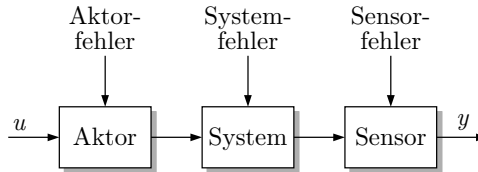
|| Ein *Fehler* ist eine unzulässige Abweichung eines oder mehrerer Merkmale einer technischen Anlage von den Nominalwerten.

Beispiele sind der Ausfall einer Versorgungsspannung, das Verklemmen eines Ventils oder das Zusetzen einer Rohrleitung. In allen diesen Fällen erfüllt eine Komponente der technischen Anlage nicht mehr die ihr zugewiesene Funktion.

Aus praktischer Sicht unterscheidet man üblicherweise zwischen externen und internen Fehlern:

- *Externe Fehler* betreffen Veränderungen in den Umweltbedingungen eines Systems, auf Grund dessen sich das System anders verhält als gefordert. So ist der Zufluss in einen Reaktor nicht nur von der Ventilöffnung, sondern auch vom Vordruck vor dem Ventil abhängig. Wenn dieser Druck abfällt, kann eine Füllstandsregelung, die die Ventilstellung als Stellgröße verwendet, den Füllstand im Behälter gegebenenfalls nicht mehr auf dem Sollwert halten, weil selbst bei vollständig geöffnetem Ventil der Zufluss den Abfluss aus dem Reaktor nicht kompensiert. Man spricht hier von einem externen Fehler, weil nicht die Regelung fehlerhaft ist, sondern die Funktionsfähigkeit der Regelung durch Einflüsse aus der Umgebung des Füllstandsregelkreises beeinträchtigt ist.
- *Interne Fehler* betreffen die Systemkomponenten selbst, also im Reaktorbeispiel den Ausfall des Füllstandssensors, auf Grund dessen die Füllstandsregelung nicht bestimmungsgemäß funktioniert. Interne Fehler werden in der Praxis häufig in Abhängigkeit von ihrem

Eingriffsort in *Aktorfehler*, *Systemfehler* und *Sensorfehler* unterteilt. Diese Einteilung ist insofern wichtig, als dass es für Aktor- und Sensorfehler spezielle Diagnoseverfahren gibt.



**Abb. 10.1:** Klassifikation interner Fehler

Fehler führen auf eine Veränderung der Eigenschaften der Anlage, die sich in einer Veränderung der gemessenen Signale äußert. Man unterscheidet dabei zwischen dem Fehler, der die primäre Ursache für die Veränderungen darstellt, und den Störungen, die der Fehler hervorruft. Beispielsweise verursacht der Fehler „Das Ventil ist blockiert.“ Störungen im Durchfluss und im Füllstand eines nachgeschalteten Behälters. Wichtig ist, dass in dieser Situation die Abweichungen des Durchflusses und des Füllstands vom Nominalverhalten nicht als Fehler, sondern als Störung bezeichnet werden. Da der Begriff der Störung in der Automatisierungstechnik vor allem im Zusammenhang mit nicht beeinflussbaren äußeren Signalen gebraucht wird, sei besonders darauf hingewiesen, dass dieser Begriff bei der Fehlerdiagnose in einem anderen Sinne, nämlich dem eines Fehlverhaltens, gebraucht wird.

**Diagnoseaufgabe.** Die Aufgabe der Prozessdiagnose kann für kontinuierliche Systeme  $\Sigma$  folgendermaßen beschrieben werden (vgl. Abb. 1.16 auf S. 31):

**Diagnoseaufgabe für kontinuierliche Systeme**

Gegeben: Zustandsraummodell des Systems  $\Sigma$   
 Verlauf der Eingangsgröße  $u_{[0, t_e]}$   
 Verlauf der Ausgangsgröße  $y_{[0, t_e]}$   
 Gesucht: Fehler  $f$

Zur Vereinfachung der Darstellung wird diese Aufgabe hier für Fehler  $f$  untersucht, die sich im Zeitintervall  $[0, t_e]$  nicht verändern. Wie die Beispiele zeigen, können die behandelten Methoden aber auch für zeitabhängige Fehler angewendet werden, die durch ein Signal  $f(t)$  repräsentiert sind.

### 10.1.2 Diagnoseschritte

Bezüglich der Genauigkeit, mit der ein Fehler diagnostiziert wird, unterscheidet man drei Teilaufgaben der Prozessdiagnose:

- **Fehlererkennung**<sup>1</sup>: Es wird untersucht, ob in der Anlage zum gegenwärtigen Zeitpunkt ein Fehler aufgetreten ist.
- **Fehlerlokalisierung**<sup>2</sup>: Wenn ein Fehler aufgetreten ist, so wird die fehlerhafte Komponente der Anlage ermittelt.
- **Fehleridentifikation**<sup>3</sup>: Es wird die Art und die Größe des Fehlers bestimmt.

Die drei angegebenen Aufgaben beziehen sich mit der Eintrittszeit eines Fehlers, dem Fehlerort und der Fehlerart auf drei verschiedene Aspekte des Fehlverhaltens eines Systems. Sie werden häufig in der angegebenen Reihenfolge gelöst. Die Bezeichnungen Prozessdiagnose und Fehlerdiagnose werden als Oberbegriffe dieser drei Aufgaben verwendet.

Wie genau der Fehler ermittelt werden muss, hängt vom Anwendungsfall ab. Bei sicherheitsrelevanten Anlagenteilen ist es oft ausreichend, das Auftreten eines Fehlers zu erkennen, weil man die fehlerhafte Anlage unter keinen Umständen weiter betreiben will und nach dem Erkennen eines Fehlers abschaltet. Andererseits können viele Geräte auch über eine gewisse Zeit im fehlerhaften Zustand arbeiten, bis der Fehler durch eine Reparatur beseitigt ist. Die Fehlerdiagnose soll dann den Fehler möglichst genau charakterisieren, um die Reparaturzeit zu verkürzen. Dabei genügt es, den Fehler der kleinsten austauschbaren Einheit zuzuordnen. Dabei muss man in der Fehlerlokalisierung erkennen, welche Komponente fehlerbehaftet ist, aber man muss nicht die Art und Größe des Fehlers ermitteln.

### 10.1.3 Signalbasierte und modellbasierte Diagnose

Für die Prozessdiagnose sind eine Reihe unterschiedlicher Verfahren entwickelt worden, deren Einsatzgebiete vor allem durch die Betriebsweise des zu diagnostizierenden Systems sowie durch die verfügbaren Messinformationen charakterisiert werden können.

**Signalbasierte Diagnose.** Eine einfache und deshalb häufig eingesetzte Methode der Fehlererkennung besteht in der Grenzwertüberwachung eines Signals. Wenn man beispielsweise durch eine Temperaturregelung die Motortemperatur unabhängig von der Belastung auf einem konstanten Wert halten will oder durch eine Druckregelung den Öldruck in einem Hydrauliksystem einem vorgegebenen Wert anpasst, braucht man nur die Regelgröße „Temperatur“ bzw. „Druck“ zu messen und mit einem vorgegebenen Grenzwert zu vergleichen, um Fehler erkennen zu können, die zu einer Überhitzung des Motors oder zu einem Abfall des Öldrucks führen.

<sup>1</sup> An Stelle von Fehlererkennung spricht man in Anlehnung an den englischen Begriff *fault detection* auch von einer Fehlerdetektion.

<sup>2</sup> Man spricht in direkter Übersetzung des in der englischen Literatur üblichen Begriffs *fault isolation* auch von Fehlerisolation.

<sup>3</sup> Fehlererkennung und Fehleridentifikation werden in der Literatur häufig durch FDI abgekürzt.



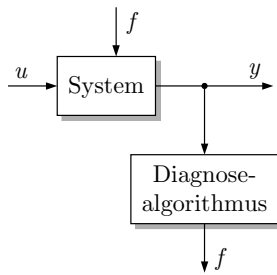


Abb. 10.2: Signalbasierte Diagnose

Der Diagnosealgorithmus wertet dabei ein Messsignal aus und zeigt bei Abweichungen des Messsignals von einem vorgegebenen Wert das Auftreten eines Fehlers an (Abb. 10.2).

Man spricht bei dieser Art der Diagnose von einer signalbasierten Diagnose, weil das Diagnosesystem ein einzelnes Signal verarbeitet. Außer der Grenzwertüberwachung können signalbasierte Diagnoseverfahren auch Klassifikationsmethoden nutzen. So kann man das Messsignal bei rotierenden Achsen einer Spektralanalyse unterziehen, um zu überprüfen, ob das System mit der richtigen Frequenz rotiert und ob große Oberwellen auf Fehler in der Lagerung einer Welle schließen lassen.

Bei der signalbasierten Diagnose ist die verwendete Art der Auswertung auf den betrachteten Prozess zugeschnitten, wobei das Wissen über das Nominalverhalten des Prozesses implizit in der Art und Weise enthalten ist, wie das gemessene Signal verarbeitet wird und welche Grenzwerte für die berechneten Merkmale vorgegeben werden. Diese Vorgehensweise ist einfach und überschaubar und vor allem dann ausreichend, wenn man nach dem Erkennen eines Fehlers die Anlage aus Sicherheitsgründen abschalten bzw. in einen sicheren Zustand überführen will.

Zwei Randbedingungen beschränken jedoch den Einsatz signalbasierter Verfahren. Erstens sind diese Verfahren im Wesentlichen auf stationär betriebene Systeme beschränkt, bei denen festgelegte Grenzwerte überwacht werden können. Bei Systemen, deren Arbeitspunkte wechseln, müssen die Grenzwerte dem aktuellen Arbeitspunkt angepasst und für die Übergangsvorgänge zwischen den Arbeitspunkten gesondert festgelegt werden, wofür der Diagnosealgorithmus auch die Eingangsgröße  $u(t)$  verarbeiten muss.

Zweitens kann man mit den signalbasierten Diagnoseverfahren i. Allg. nur den Schritt der Fehlererkennung durchführen, die Fehler aber nicht identifizieren. Eine zu hohe Motortemperatur zeigt nur an, dass irgendein Fehler eingetreten ist, lässt aber keine Schlüsse auf den Fehler selbst zu.

**Modellbasierte Diagnose.** Wenn man Fehler während eines dynamischen Übergangsvorgangs erkennen will oder wenn man den Fehler identifizieren will, reicht es nicht aus, einzelne Signale separat voneinander zu betrachten, sondern man muss den Zusammenhang mehrerer Signale für die Diagnose ausnutzen. Die Diagnose erfolgt dann entsprechend dem in Abb. 1.16 auf S. 31 gezeigten Schema, bei dem der Diagnosealgorithmus nicht nur die Ausgangsgröße  $y(t)$ , sondern auch die Eingangsgröße  $u(t)$  als Online-Informationen erhält und diese Signale mit Hilfe

eines Prozessmodells auswertet. Da der Diagnosealgorithmus direkt mit einem Prozessmodell arbeitet, spricht man von einer *modellbasierten Diagnose*.

Dass bei der modellbasierten Diagnose das Prozessmodell einen eigenständigen „Baustein“ des Algorithmus darstellt, hat für den praktischen Einsatz wichtige Konsequenzen. Einerseits kann man den Diagnosealgorithmus für einen bestimmten Prozess dadurch entwickeln, dass man eine allgemeingültige Diagnosemethode mit dem Modell des zu diagnostizierenden Prozesses kombiniert. Dieses Vorgehen wird in diesem Kapitel an mehreren Beispielen demonstriert. Der Zuschnitt des Diagnosesystems auf den konkreten Anwendungsfall erfolgt also durch das Modell. Andererseits kann man modellbasierte Diagnoseverfahren an Veränderungen im Prozess, die sich während des Betriebs oder in einer Massenfertigung auf Grund der Variantenvielfalt des Produkts ergeben, einfach dadurch anpassen, dass man das Prozessmodell dementsprechend modifiziert.

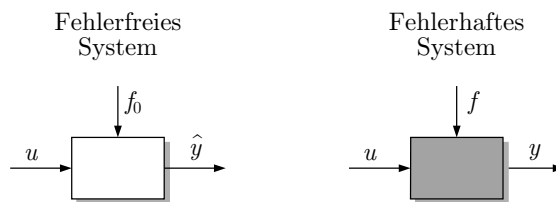
Die gemeinsame Auswertung des Eingangssignals und des Ausgangssignals begründet das breite Anwendungsfeld modellbasierter Verfahren. Fehler können auf diese Weise auch bei stationären Betriebsweisen erkannt werden. Auch ist es möglich, Fehler in geschlossenen Regelkreisen zu identifizieren, was bei signalbasierten Verfahren schwierig ist, weil die Regelung die Wirkung von Fehlern zumindest teilweise kompensiert und Fehler damit nicht im Verhalten des Regelkreises „sichtbar“ sind. Diese Eigenschaft modellbasierter Diagnoseverfahren wird u. a. im Beispiel 10.4 demonstriert, in dem ein Fehler im Servoventil trotz des den Fehler korrigierenden Eingriffs eines Reglers erkannt wird.

#### 10.1.4 Modelle des fehlerfreien und des fehlerhaften Systems

Das zu überwachende System wird für unterschiedliche Fehlerfälle  $f$  betrachtet, die zu einer Menge

$$\mathcal{F} = \{f_0, f_1, \dots, f_q\}$$

möglicher Fehler  $f_i$  gehören. Dabei wird der fehlerfreie Betriebszustand mit  $f_0$  bezeichnet und zur Vereinfachung der Betrachtungen mit zur Fehlermenge gezählt. Ein Fehlersymbol  $f_i$  kann auch einen Mehrfachfehler bezeichnen.



**Abb. 10.3:** Fehlerfreies und fehlerhaftes System

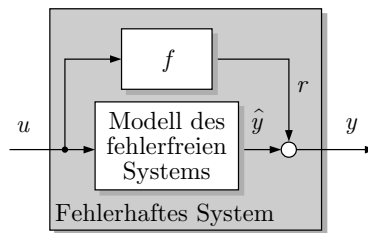
In der hier untersuchten Situation, in der ein Fehler im betrachteten Zeitintervall unverändert vorhanden oder nicht vorhanden ist, führt jeder Fehler  $f \neq f_0$  auf eine bleibende Veränderung des Zusammenhangs zwischen der Eingangsgröße  $u(t)$  und der Ausgangsgröße  $y(t)$

(Abb. 10.3). Für eine vorgegebene Eingangsgröße  $u(t)$  entsteht dann am Ausgang das Signal  $y(t)$ , das sich auf Grund der Fehlerwirkung vom Ausgang  $\hat{y}(t)$  des fehlerfreien Systems unterscheidet.

|| Fehler verändern das E/A-Verhalten des betroffenen Systems.

Im Modell führen sie auf Parameteränderungen oder werden durch zusätzliche Signale beschrieben.

Modellbasierte Diagnoseverfahren überprüfen, ob der Zusammenhang zwischen den gemessenen Signalen  $u(t)$  und  $y(t)$  derselbe ist wie im Modell des fehlerfreien Systems. Wenn sie den Fehler  $f$  identifizieren sollen, muss das Prozessmodell auch Informationen darüber enthalten, wie die Fehler das Systemverhalten verändern. Das Diagnoseverfahren vergleicht dann die gemessenen Signale mit den Modellen, die das System bei den Fehlern  $f \in \mathcal{F}$  beschreiben.



**Abb. 10.4:** Dekomposition eines fehlerhaften Systems

Da der Fehler  $f$  das E/A-Verhalten des Systems verändert, ist der Unterschied zwischen den Ausgängen des fehlerfreien und des fehlerhaften Systems von der Eingangsgröße  $u(t)$  abhängig. Der in Abb. 10.3 rechts gezeigte graue Block kann deshalb in eine Parallelschaltung des Modells des fehlerfreien Systems und eines Blocks zerlegt werden, der die vom Fehler  $f$  hervorgerufenen Verhaltensänderungen beschreibt. Die am fehlerhaften System auftretende Ausgangsgröße  $y(t)$  unterscheidet sich von der Ausgangsgröße  $\hat{y}(t)$ , die das fehlerfreie System bei der Eingangsgröße  $u(t)$  erzeugt, um den Wert  $r(t)$ , der *Residuum* (lat. „Rest“, „das Zurückgebliebene“) genannt wird (Abb. 10.4).

**Multiplikative Fehler.** Bezüglich der Art, wie Fehler die Modelleigenschaften verändern, unterscheidet man zwischen multiplikativen und additiven Fehlern. Multiplikative Fehler verändern die Modellparameter. Für den in Aufgabe 3.7 auf S. 114 behandelten Wärmeübertrager veranschaulicht das in Abb. 3.20 dargestellte Rohr ein ganzes Rohrbündel. Wenn ein oder mehrere Rohre des Rohrbündels verstopft sind, verändern sich die Fläche, über die der Wärmeaustausch erfolgt, und die Geschwindigkeit, mit der das Kühlmittel durch den Wärmeübertrager fließt, gegenüber dem Nominalfall. Beide Wirkungen des Fehlers führen auf Parameteränderungen, also auf Veränderungen der Matrix  $A$ , des Vektors  $b$  oder des Vektors  $c^T$  im Zustandsraummodell. Da diese Größen mit den Signalen  $x(t)$  und  $u(t)$  multipliziert werden, spricht man

von *multiplikativen Fehlern*. In allgemeiner Form kann man das Modell des mit dem Fehler  $f$  behafteten Systems  $\Sigma_f$  in der folgenden Form schreiben:

$$\Sigma_f : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}(f)\mathbf{x}(t) + \mathbf{b}(f)u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T(f)\mathbf{x}(t). \end{cases}$$

**Additive Fehler.** Ein Beispiel für einen Fehler, der durch ein zusätzliches Signal repräsentiert wird, ist die Drift eines Sensors. Sensoren für den pH-Wert, wie sie im Beispiel 10.3 auf S. 308 betrachtet werden, verändern schon nach kurzer Zeit ihren „Nullpunkt“, d. h., die Messung ist mit einem systematischen Fehler behaftet. Die Differenz zwischen dem vom Sensor angezeigten Wert  $y_m(t)$  und dem wahren Messwert  $y(t)$  wird durch ein zusätzliches (unbekanntes) Eingangssignal  $f(t)$  beschrieben:

$$y_m(t) = y(t) + f(t). \quad (10.1)$$

In anderen Anwendungsfällen wie bei dem im Beispiel 10.4 auf S. 320 behandelten Stellgerät tritt der Fehler als zusätzliche Eingangsgröße in der Zustandsgleichung auf, so dass das fehlerbehaftete System  $\Sigma_f$  durch ein Modell der Form

$$\Sigma_f : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}f(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = \mathbf{c}^T\mathbf{x}(t) \end{cases} \quad (10.2)$$

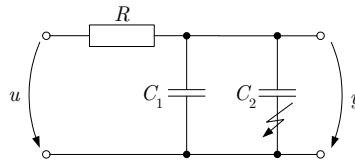
beschrieben ist.  $\mathbf{g}$  ist ein Vektor, der angibt, wie der Fehler  $f(t)$  auf die Komponenten  $\dot{x}_i(t)$  des Vektors  $\dot{\mathbf{x}}(t)$  wirkt. Da der Fehler in diesem Modell als zusätzlicher Summand erscheint, wird er als *additiver Fehler* bezeichnet.

**Analytische Redundanz.** Um eine Prozessdiagnose durchführen zu können, braucht man redundante Informationen über das Systemverhalten. Bei der modellbasierten Diagnose besteht diese Redundanz darin, dass man neben den Messgrößen über ein (analytisches) Modell verfügt, mit Hilfe dessen man einen Teil der Messgrößen berechnen kann. Man spricht deshalb von einer analytischen Redundanz. Aus gemessenen und berechneten Größen kann man das Residuum  $r(t) = y(t) - \hat{y}(t)$  bilden und überprüfen, ob im System ein Fehler eingetreten ist.

In dieser Beziehung unterscheidet sich die Diagnose von der Vorhersage. Wenn mit einem Modell das zukünftige Verhalten eines Systems berechnet werden soll, genügt es, neben dem Modell die Eingangsgröße  $u(t)$  und den Anfangszustand  $\mathbf{x}_0$  zu kennen. Man hat dann keine redundanten Informationen, denn aus den gegebenen Größen wird die unbekannte Größe  $y(t)$  bestimmt. Im Gegensatz dazu muss man bei der Diagnose außer der Eingangsgröße  $u(t)$  und dem Anfangszustand  $\mathbf{x}_0$  auch die Ausgangsgröße  $y(t)$  messen. Die Messgröße  $y(t)$  und die mit dem Modell berechnete Größe  $\hat{y}(t)$  liefern redundante Informationen, die die Diagnose möglich machen. Das Modell wird nicht für die Berechnung einer unbekannten Größe, sondern zur Überprüfung des Systemverhaltens eingesetzt.

### 10.1.5 Diagnose mit statischen und dynamischen Modellen

Modellbasierte Diagnoseverfahren unterscheiden sich bezüglich des Charakters der verwendeten Modelle. Klassische Verfahren verwenden statische Modelle, die von der Vorstellung ausgehen, dass Fehler zu einem neuen statischen Arbeitspunkt führen. Wenn beispielsweise der Anschluss an einen Druckkessel undicht geworden ist, wird der Druck im Kessel einen kleineren Wert als im Normalbetrieb annehmen, woran man den Fehler detektieren kann.



**Abb. 10.5:** RC-Glied mit fehlerhaft angeschlossenen Kondensator

Wichtige Fehler sind jedoch nur aus einer Analyse des dynamischen Verhaltens zu erkennen. Als Beispiel zeigt Abb. 10.5 ein RC-Glied, bei dem zwei Kondensatoren parallel geschaltet sind und der Fehler darin besteht, dass sich ein Anschluss eines der beiden Kondensatoren gelöst hat. In allen statischen Arbeitspunkten hat die Ausgangsspannung  $y$  denselben Wert wie die Eingangsspannung  $u$ , unabhängig davon, ob ein oder beide Kondensatoren angeschlossen sind. Der Fehler kann nur daran erkannt werden, dass sich die Zeitkonstante  $T$  des Auflade- oder Entladevorgangs mit der Kapazität ändert, im fehlerfreien Zustand also  $T = R \cdot (C_1 + C_2)$  gilt, während im fehlerhaften Zustand der Übergangsvorgang mit einer kleineren Zeitkonstante  $T_f = R \cdot C_1$  abklingt.

Dynamische Modelle sind auch dann erforderlich, wenn man Fehler zwar im statischen Zustand erkennen könnte, das System jedoch aufgrund der äußeren Erregung keine statischen Zustände annimmt.

Die in diesem Kapitel und im Kap. 16 behandelten Diagnoseverfahren verwenden dynamische Modelle. Die Beispiele werden zeigen, dass mit diesen Modellen Fehler auch innerhalb von Übergangsvorgängen erkannt werden können und dass es möglich ist, nicht nur das Auftreten eines Fehlers zu erkennen, sondern die Fehler auch zu identifizieren.

### 10.1.6 Prinzip der konsistenzbasierten Diagnose

**Fehlererkennung.** Vereinfacht dargestellt läuft eine modellbasierte Diagnose folgendermaßen ab: Man berechnet mit dem Modell des fehlerfreien Systems unter Verwendung der gemessenen Eingangsgröße  $u(t)$  die Ausgangsgröße  $\hat{y}(t)$  und vergleicht diese mit dem gemessenen Systemausgang  $y(t)$ . Das System verhält sich genauso wie das Modell, wenn die Beziehung  $y(t) = \hat{y}(t)$  (wenigstens näherungsweise) gilt und man sagt dann, dass das Systemverhalten mit dem Modell *konsistent* ist.

Wird eine Abweichung zwischen dem Modellausgang und dem gemessenen Prozessausgang erkannt, so kann gefolgert werden, dass sich das System nicht so wie im fehlerfreien Betriebszustand verhält und folglich ein Fehler aufgetreten sein muss. Die Differenz

$r(t) = y(t) - \hat{y}(t)$  zwischen dem gemessenen und dem berechneten Ausgang bildet das Residuum das als Indikator für einen Fehler verwendet wird. Ein nicht verschwindendes Residuum zeigt das Auftreten eines Fehlers an (Abb 10.4).

Das Prinzip der *konsistenzbasierten Diagnose* kann folgendermaßen zusammengefasst werden:

|| **Fehlererkennung:** Wenn das gemessene Systemverhalten nicht mit dem Modell des fehlerfreien Systems konsistent ist, ist ein Fehler aufgetreten.

Die Aufgabe der Fehlererkennung kann also mit dem Modell des fehlerfreien Systems gelöst werden.

Mit der Bezeichnung „gemessene Eingänge und Ausgänge“ soll darauf hingewiesen werden, dass  $u(t)$  und  $y(t)$  die aktuell an der Anlage vorhandenen Größen sind. Ob sie in jedem Fall gemessen oder aus einer anderen Komponente der Automatisierungseinrichtung (wie z. B. einem Regler) direkt übermittelt werden, ist dabei gleichgültig.

**Fehlerlokalisierung und Fehleridentifikation.** Das Prinzip der konsistenzbasierten Diagnose kann man so erweitern, dass man es auch für die Aufgaben der Fehlerlokalisierung und Fehleridentifikation einsetzen kann. Dafür wendet man es auf Modelle an, die die Anlage unter der Einwirkung der Fehler  $f \in \mathcal{F}$  beschreiben. Wenn das gemessene Verhalten nicht mit dem für den Fehler  $f$  aufgestellten Modell konsistent ist, so kann der Fehler  $f$  ausgeschlossen werden. Alle anderen Fehler können möglicherweise im System vorhanden sein und werden deshalb als *Fehlerkandidaten* bezeichnet.

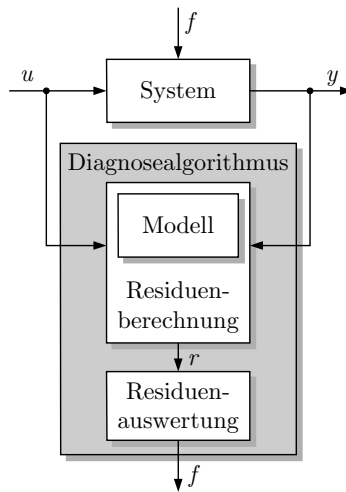
|| **Fehlerlokalisierung und Fehleridentifikation:** Alle Fehler  $f \in \mathcal{F}$ , für die das gemessene Systemverhalten mit dem für  $f$  gültigen Modell konsistent ist, sind Fehlerkandidaten.

Für die Fehlerlokalisierung benötigt man ein Modell, das zeigt, wie Fehler das Verhalten der einzelnen Komponenten verändern, und für die Fehleridentifikation ein Modell, das die Wirkung von Fehlern unterschiedlicher Größe auf das E/A-Verhalten darstellt. Da das Aufstellen derartiger Modelle in der Praxis häufig schwierig ist, sei hier betont, dass ohne Informationen darüber, wie Fehler das Systemverhalten verändern können, nur diagnostiziert werden kann, *ob* ein Fehler eingetreten ist, jedoch keine weitergehende Eingrenzung des Fehlers möglich ist. Will man diese Eingrenzung des Fehlers vornehmen, braucht man die erwähnten Modelle.

Auf einen zweiten für die Anwendung wichtigen Aspekt sei abschließend hingewiesen: Das Ergebnis jeder Diagnose sind *Fehlerkandidaten*. Selbst wenn die Diagnosemethode nur einen einzigen Fehler  $f$  als Lösung der Diagnoseaufgabe ermittelt, ist nicht gesichert, dass dieser Fehler tatsächlich aufgetreten ist. Bewiesen ist lediglich, dass sich bei diesem Fehler das System so verhält, wie es die gemessenen Signale zeigen. Nur wenn man sicher ist, alle möglichen Fehler im Modell berücksichtigt zu haben, und wenn die Lösung eindeutig ist, kann man den Fehler mit Sicherheit angeben.

**Struktur der Diagnosealgorithmen.** Das Prinzip der konsistenzbasierten Diagnose führt für kontinuierliche Systeme auf die in Abb. 10.6 gezeigte Struktur von Diagnosealgorithmen. Die Diagnose erfolgt in zwei Schritten:

1. Residuenberechnung
2. Residuenauswertung.



**Abb. 10.6:** Prozessdiagnose kontinuierlicher Systeme

Von einem Residuum erwartet man, dass es verschwindet, solange kein Fehler auftritt, und dass es sich wesentlich von null unterscheidet, wenn ein Fehler aufgetreten ist:

$$\begin{aligned}
 f = f_0 &\implies r(t) = 0 \\
 f \neq f_0 &\implies r(t) \neq 0.
 \end{aligned}$$

Inwieweit diese Forderung trotz des Auftretens von Störungen und Modellunbestimmtheiten erfüllt werden kann, wird im Folgenden untersucht. Im Allgemeinen kann die Forderung wenigstens näherungsweise erfüllt werden, so dass bei der Residuenauswertung überprüft wird, ob das Residuum betragsmäßig einen vorgegebenen Grenzwert  $\bar{r}$  überschreitet:

$$|r(t)| \stackrel{?}{>} \bar{r}. \quad (10.3)$$

In vielen Anwendungsfällen werden mehrere Residuen berechnet, wobei im Idealfall das  $i$ -te Residuum die Ungleichung (10.3) erfüllt, wenn der  $i$ -te Fehler eingetreten ist.

In den nachfolgenden Abschnitten wird zunächst das Problem der Fehlererkennung behandelt und ein Verfahren erläutert, das mit einem Modell des fehlerfreien Systems auskommt. Im Abschn. 10.4 wird diese Vorgehensweise für die Fehleridentifikation erweitert.

## 10.2 Fehlererkennung mit einem Zustandsbeobachter

Die bei der Diagnose zu beantwortende Frage, ob die Messgrößen  $u_{[0,t_e]}$  und  $y_{[0,t_e]}$  mit einem Modell konsistent sind, hängt vom Anfangszustand  $x_0$  ab. Wenn der Anfangszustand – wie es häufig der Fall ist – unbekannt ist, beinhaltet die Diagnoseaufgabe die Beobachtungsaufgabe, den Anfangszustand  $x_0$  bzw. den aktuellen Zustand  $x(t_e)$  zu finden, mit dem die Konsistenz festgestellt werden kann. Aus diesem Grund beruhen viele Diagnosemethoden auf Methoden zur Zustandsbeobachtung.

Beim Zustandsbeobachter beschreibt das Residuum  $r(t) = y(t) - \hat{y}(t)$  die Differenz zwischen der gemessenen Ausgangsgröße  $y(t)$  und der durch den Beobachter berechneten Ausgangsgröße  $\hat{y}(t)$ . Es stimmt also mit dem Signal überein, das innerhalb von Beobachtern für die Realisierung der Rückführung (9.5) gebildet wird.

Im Abschn. 9.2.4 wurde gezeigt, dass der Fehler  $e(t)$ , mit dem ein Beobachter den Zustand eines Systems berechnet, nicht verschwindet, wenn das im Beobachter verwendete Prozessmodell nicht mit dem beobachteten System übereinstimmt. Diese Tatsache macht man sich bei der beobachtergestützten Diagnose zu Nutze. Man überprüft die Konsistenz der gemessenen Größen  $u_{[0,t_e]}$  und  $y_{[0,t_e]}$  mit dem Modell des fehlerfreien Systems dadurch, dass man diese Größen mit einem Beobachter (9.7)

$$\Sigma_B : \begin{cases} \dot{\hat{x}}(t) = (A - lc^T) \hat{x}(t) + bu(t) + ly(t), & \hat{x}(0) = \hat{x}_0 \\ \hat{y}(t) = c^T \hat{x}(t) \end{cases} \quad (10.4)$$

verarbeitet und das Residuum

$$r(t) = y(t) - \hat{y}(t) \quad (10.5)$$

erzeugt. Wenn nach dem Einschwingen des Beobachters das Residuum nicht verschwindet, kann auf das Auftreten eines Fehlers geschlossen werden (Abb. 10.7). Da man im Beobachter das Modell des fehlerfreien Systems verwendet, bedeutet ein bleibendes Residuum, dass sich das System anders als das Modell verhält, und man kann daraus schließen, dass ein Fehler aufgetreten ist. Auf diese Weise ist eine Fehlererkennung möglich.

---

### Algorithmus 10.1 Beobachterentwurf zur Residuenberechnung für die Fehlererkennung

---

**Gegeben:** Modell (10.2) des zu diagnostizierenden Prozesses.

1. Wählen Sie  $n$  Beobachtereigenwerte  $\lambda_{Bi}$ .
2. Berechnen Sie die Beobachterrückführung  $l$  mit Gl. (9.12).

**Ergebnis:** Beobachter (10.4), (10.5) zur Residuenberechnung.

---

Der hier für die Fehlererkennung eingesetzte Beobachter unterscheidet sich von den im Kap. 9 betrachteten Beobachtern nur dadurch, dass jetzt nicht der Beobachterzustand  $\hat{x}$ , sondern das Residuum entsprechend Gl. (10.3) weiter verarbeitet wird. Die Residuenauswertung erfolgt i. Allg. durch die Überprüfung, ob das Residuum einen vorgegebenen Grenzwert überschreitet.



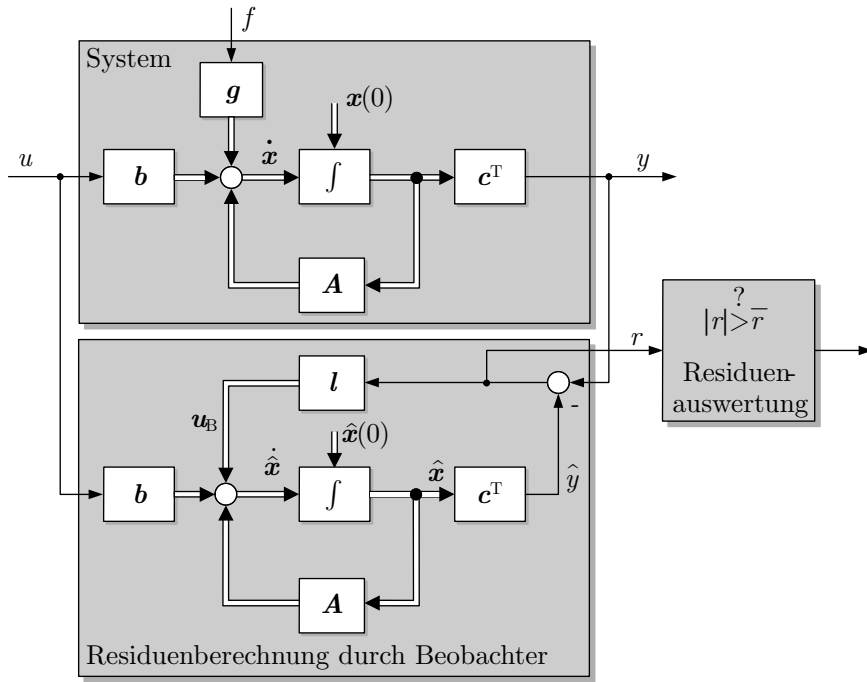


Abb. 10.7: Fehlererkennung mit Hilfe eines Beobachters

**Beispiel 10.2** Einklemmschutz für einen Fensterheber

Elektrische Fensterheber in Kraftfahrzeugen, die das Fenster auf Knopfdruck vollständig schließen, können zu Unfällen führen, wenn die Scheibe Körperteile gegen den oberen Türrahmen quetscht. Deshalb schreibt die Straßenverkehrszulassungsordnung vor, dass diese Fensterheber mit einem Einklemmschutz zu versehen sind, der die Einklemmkraft auf  $F_K = 100 \text{ N}$  beschränkt, bevor der Rückwärtslauf des Motors eine Umkehrbewegung der Scheibe bewirkt. Im Folgenden wird die Aufgabe betrachtet, den Einklemmfall zu diagnostizieren.

**Modellbildung.** Der vom Fensterhebermotor aufgebrauchten Kraft  $F_A(t)$  wirken die Gewichtskraft  $F_m(t)$ , die Reibungskraft  $F_R(t)$  sowie eine Störkraft  $F_d(t)$  entgegen, die durch Straßenunebenheiten hervorgerufen wird (Abb. 10.8). Ein zwischen der Oberkante der Fensterscheibe und dem Türrahmen eingeklemmter Gegenstand erzeugt eine zusätzliche Kraft  $F_K(t)$ , die nur im Fehlerfall auftritt und mit der die Beziehung

$$F_m(t) = F_A(t) - F_R(t) - F_d(t) - F_K(t)$$

gilt. Das Einklemmen eines Gegenstands, der die zusätzlich wirkende Kraft  $F_K(t)$  erzeugt, führt dazu, dass ein Beobachter für den Fensterhebermotor ein bleibendes Residuum erzeugt.

Zur Aufstellung des Modells werden die Hebeeinrichtung und die Fensterscheibe als eine vom Motor zu bewegend Masse zusammengefasst, die auf den Motor mit dem Trägheitsmoment  $J$  wirkt (Abb. 3.2 auf S. 87). Für den Motor muss das in Gl. (3.8) auf der rechten Seite stehende Beschleunigungsmoment wie bisher aus dem Motormoment  $M(t)$  und dem Reibungsmoment  $M_R(t)$  sowie den hier neu auftretenden Momenten  $M_d(t)$  und  $M_K(t)$  berechnet werden, womit man die Beziehung

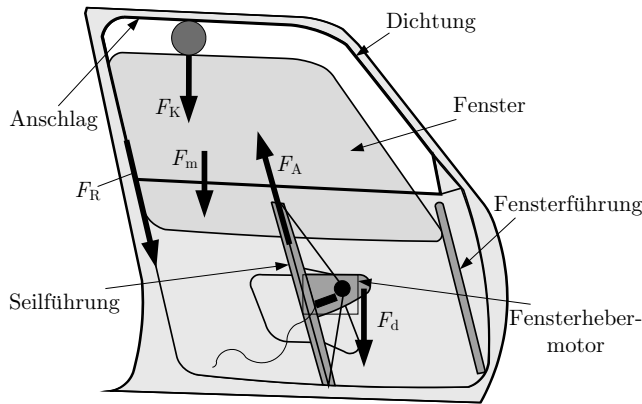


Abb. 10.8: Elektrischer Fensterheber

$$J\dot{\omega}(t) = M(t) - M_R(t) - M_d(t) - M_K(t)$$

erhält. Die zwei ganz rechts stehenden Momente werden durch die Straßenunebenheiten bzw. den eingeklemmten Gegenstand hervorgerufen. An Stelle der zuvor eingeführten Kräfte erscheinen in dieser Gleichung die über die Motorübersetzung wirkenden Momente. Damit erweitert sich das Zustandsraummodell (3.16) des Gleichstrommotors zum Modell

$$\text{Fensterheber : } \left\{ \begin{array}{l} \frac{d}{dt} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} -\frac{R}{L} & -\frac{k_M}{L} \\ \frac{k_T}{J} & -\frac{k_B}{J} \end{pmatrix} \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix} u(t) \\ \quad + \begin{pmatrix} 0 \\ -\frac{1}{J} \end{pmatrix} M_K(t) + \begin{pmatrix} 0 \\ -\frac{1}{J} \end{pmatrix} M_d(t), \quad \begin{pmatrix} i(0) \\ \omega(0) \end{pmatrix} = \begin{pmatrix} i_0 \\ \omega_0 \end{pmatrix} \\ y(t) = (0 \quad 1) \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix} \end{array} \right. \quad (10.6)$$

mit dem Zustandsvektor

$$\mathbf{x}(t) = \begin{pmatrix} i(t) \\ \omega(t) \end{pmatrix},$$

das durch

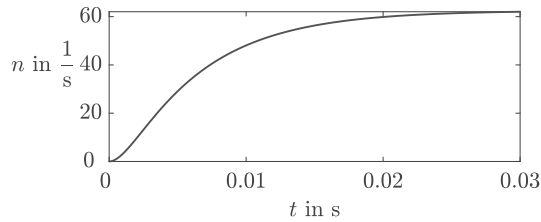
$$\text{Fensterheber : } \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) + \mathbf{g}M_K(t) + \mathbf{g}M_d(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \\ y_L(t) = \mathbf{c}^T \mathbf{x}(t) \end{array} \right.$$

abgekürzt wird. Die Winkelgeschwindigkeit  $y(t) = \omega(t)$  wird am Motor gemessen.

**Beobachterentwurf.** Der für die Diagnose verwendete Beobachter heißt

$$\Sigma_B : \left\{ \begin{array}{l} \dot{\hat{\mathbf{x}}}(t) = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) \hat{\mathbf{x}}(t) + \mathbf{b}u(t) + \mathbf{l}y(t), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0 \\ r(t) = -\mathbf{c}^T \hat{\mathbf{x}}(t) + y(t), \end{array} \right.$$

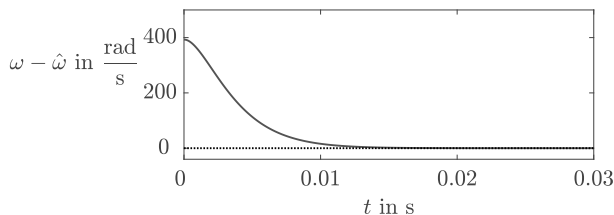
wobei die Ausgangsgleichung für das Residuum aufgeschrieben ist. Für die Parameter



**Abb. 10.9:** Verlauf der Drehzahl  $n$  beim Einschalten des Fensterhebers

$R = 0,5 \, \Omega$	Widerstand der Wicklung
$L = 5 \cdot 10^{-4} \, \text{H}$	Induktivität der Wicklung
$k_M = 0,03 \, \text{Vs}$	Motorparameter
$k_T = 0,03 \, \frac{\text{Nm}}{\text{A}}$	Motorparameter
$k_R = 32,7 \cdot 10^{-6} \, \text{Nm s}$	Motorparameter
$J = 13 \cdot 10^{-6} \, \frac{\text{Nm s}^2}{\text{rad}}$	Trägheitsmoment der Fensterscheibe

läuft der Motor sehr schnell an, wie der in Abb. 10.9 gezeigte Einschaltvorgang beim Anlegen der Betriebsspannung von 12 Volt zeigt. Die Eigenwerte des Systems liegen bei  $\lambda_1 = -776 \, \frac{1}{\text{s}}$  und  $\lambda_2 = -226 \, \frac{1}{\text{s}}$ .

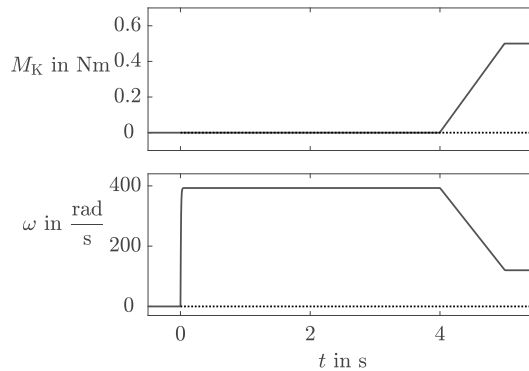


**Abb. 10.10:** Verlauf des Beobachtungsfehlers

Die Beobachterrückführung  $\mathbf{l}$  wird so ausgewählt, dass die beiden Beobachtereigenwerte bei  $-500 \, \frac{1}{\text{s}}$  liegen. Der Beobachter schwingt deshalb in einer mit dem Einschaltvorgang des Motors vergleichbaren Zeit ein. Abbildung 10.10 zeigt das Einschwingen, wenn der Beobachter mit verschwindendem Anfangszustand dem bereits eingeschalteten Motor zur Zeit  $t = 0$  zugeschaltet wird. Der Beobachtungsfehler verschwindet innerhalb von etwa 10 Millisekunden und zeigt den ordnungsgemäßen Betrieb des Fensterhebers an.

**Fehlererkennung.** Wenn ein elastischer Gegenstand zwischen der Fensterscheibe und den Türrahmen eingeklemmt wird, baut sich ein Moment  $M_K(t)$  auf, das die Motorgeschwindigkeit herabsetzt. Dieser Vorgang ist in Abb. 10.11 für  $t > 4 \, \text{s}$  gezeigt. Man beachte, dass dieses Moment nicht messbar ist. Gemessen werden kann lediglich die Geschwindigkeitsabnahme des Motors, die im unteren Teil der Abbildung zu sehen ist.

Der Diagnosealgorithmus soll das Einklemmen so schnell wie möglich anzeigen. Diese Aufgabe kann er nicht allein anhand des Verlaufs der Winkelgeschwindigkeit  $\omega(t)$  lösen, denn die Winkelgeschwindigkeit könnte auch auf null absinken, weil der Motor abgeschaltet wurde. Der Einklemmfall

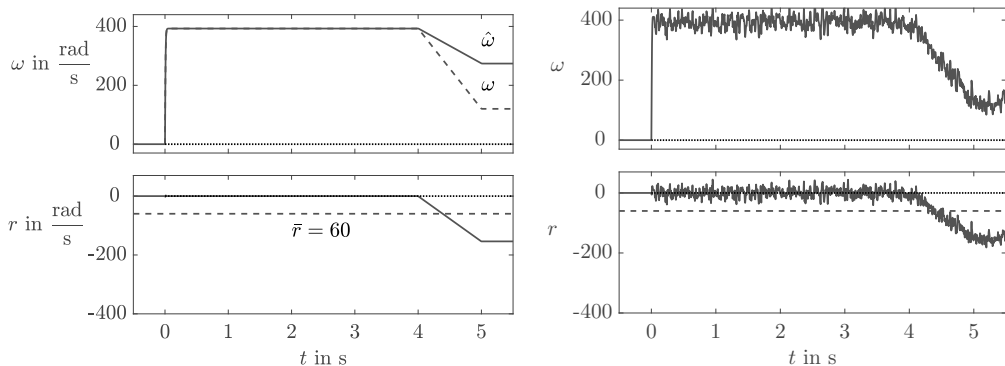


**Abb. 10.11:** Verhalten des Fensterhebers beim Einklemmen eines Gegenstands

kann nur unter gleichzeitiger Betrachtung der Eingangsgröße und der Ausgangsgröße diagnostiziert werden, wie es der Beobachter macht.

Abbildung 10.12 (links) zeigt im unteren Teil das vom Beobachter berechnete Residuum  $r(t)$ . Zur Zeit  $t = 4$  s tritt der Einklemmfall auf. Das Residuum hat sofort einen von null verschiedenen Wert. Grund ist die Differenz zwischen der gemessenen Winkelgeschwindigkeit  $\omega(t)$  und der vom Beobachter berechneten Winkelgeschwindigkeit  $\hat{\omega}(t)$ . Dieses Ergebnis gilt für  $M_d(t) = 0$ . Für die Residuenauswertung entsprechend Gl. (10.3) wird ein Grenzwert  $\bar{r}$  festgelegt, der die später betrachteten Störungen berücksichtigt. In diesem Beispiel erkennt das Diagnosesystem zur Zeit  $t = 4,3$  s den Einklemmfall und kann die Rückwärtsbewegung des Fensterhebermotors einleiten.

Dass die Diagnose auch unter praktischen Randbedingungen möglich ist, zeigt Abb. 10.12 (rechts). Die Abbildung gibt das Diagnoseergebnis wieder, das unter der Wirkung eines stochastischen Moments  $M_d(t)$  erhalten wurde, das den Einfluss von Straßenunebenheiten (z. B. Kopfsteinpflaster) beschreibt. Die Winkelgeschwindigkeit ist gestört, wie im oberen Teil der Abbildung zu sehen ist. Diese Störung überträgt sich auf das Residuum. Für die Abbildung wurde in beiden Teilen derselbe Maßstab verwendet, so dass man sieht, dass der Beobachter die Störung mit einer gewissen Dämpfung auf das Residuum überträgt. Trotz der Störung zeigt das Diagnosesystem den Einklemmfall zweifelsfrei zur Zeit  $t = 4,3$  s an.



**Abb. 10.12:** Erkennen des Einklemmfalls bei  $|r(t)| \neq 0$  (links: ohne Störung; rechts: bei Störung durch Straßenunebenheiten)

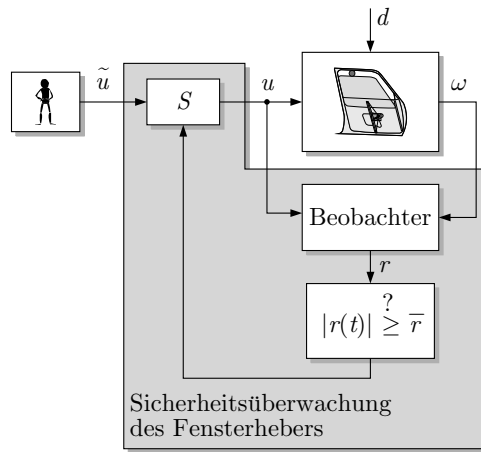


Abb. 10.13: Sicherheitsüberwachung des Fensterhebers

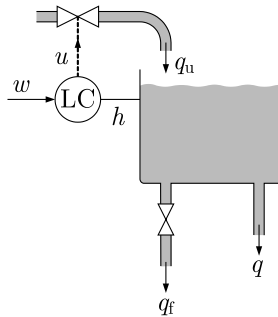
**Struktur der Sicherheitsüberwachung des Fensterhebers.** Mit dem beschriebenen beobachterbasierten Diagnosesystem hat die Sicherheitsüberwachungseinrichtung des Fensterhebers die in Abb. 10.13 gezeigte Struktur. Der Block  $S$  gibt den durch das Signal  $\tilde{u}(t)$  dargestellten Wunsch des Fahrers, das Fenster zu öffnen oder zu schließen, nur dann als Signal  $u(t)$  an den Motor weiter, wenn kein Einklemmfall detektiert wurde. Im Einklemmfall schaltet der Block  $S$  den Motor auf „Öffnen des Fensters“ um. Das Verhalten des Fensters wird mit Hilfe des Beobachters überwacht, der das Residuum  $r(t)$  erzeugt, dessen Wert mit dem Grenzwert  $\bar{r}$  verglichen wird.  $\square$

#### Aufgabe 10.1 Fehlerdiagnose eines Pendels

Das durch das lineare Modell (6.16) auf S. 193 beschriebene Pendel ändert seine Schwingungsdauer, wenn seine Masse  $m$  beispielsweise dadurch verändert wird, dass ein Teil des Pendels abfällt. Dieser Fehler soll mit Hilfe eines Zustandsbeobachters diagnostiziert werden. Da das Pendel keine Eingangsgröße  $u$  besitzt, entfällt in der Beobachtergleichung der entsprechende Term. Wie hängt das Residuum  $r(t)$  von der Masseänderung ab? Beantworten Sie diese Frage für die beiden Fälle, dass der Winkel bzw. die Winkelgeschwindigkeit des Pendels gemessen werden kann.  $\square$

#### Aufgabe 10.2\* Fehlerdiagnose eines Reaktors mit Füllstandsregelung

Der in Abb. 10.14 gezeigte Reaktor hat eine proportionale Füllstandsregelung, die den Füllstand  $h(t)$  auf einem konstanten Sollwert  $w$  halten soll. Bei dem hier verwendeten linearen Modellansatz ist der Volumenstrom  $q(t)$  der ausfließenden Flüssigkeit proportional zum Füllstand. Ein Diagnosesystem soll überprüfen, dass sich das linke Auslaufventil, das während eines Batchprozesses häufig geöffnet und geschlossen wird, gegenwärtig im geschlossenen Zustand befindet und somit für den Volumenstrom der dort austretenden Flüssigkeit die Beziehung  $q_f(t) = 0$  gilt. Dafür steht dem Diagnosesystem die Messgröße  $q(t)$  zur Verfügung.



**Abb. 10.14:** Reaktor mit Füllstandsregelung

1. Stellen Sie ein Zustandsraummodell des geregelten Reaktors für den fehlerfreien und den Fehlerfall auf. Wodurch unterscheiden sich die Modelle für beide Fälle? Welches Modell wird für die Fehlerdetektion eingesetzt?
2. Zeichnen Sie ein Blockschaltbild, aus dem die Kopplung des geregelten Reaktors mit dem Diagnosesystem hervorgeht. Beachten Sie dabei, dass bei diesem Anwendungsfall der Reaktor zusammen mit seiner Füllstandsregelung diagnostiziert wird.
3. Stellen Sie die Gleichungen für ein beobachtergestütztes Diagnosesystem auf und berechnen Sie die Beobachterrückführung so, dass der Beobachter etwa doppelt so schnell ist wie der geregelte Reaktor.
4. Beschreiben Sie das Residuum durch ein Modell und zeigen Sie, dass im fehlerfreien Fall die Beziehung  $r(t) = 0$  gilt und dass das Residuum mit der Größe des Fehlers ansteigt.
5. Wie verändern sich Blockschaltbild, Modell und Diagnosesystem, wenn der rechte Auslauf des Reaktors durch ein Ventil verschlossen ist, dessen Stellung durch eine externe Steuergröße  $u_2(t)$  vorgegeben wird und wenn als Messgröße der Füllstand  $h(t)$  dem Diagnosesystem zur Verfügung gestellt wird?
6. Kann man den Fehler auch an einer Änderung der statischen Werte der Messgrößen erkennen? Welchen Vorteil hat die beobachtergestützte Diagnose gegenüber diesem Vorgehen? Verändern sich die Bedingungen für die Diagnose, wenn an Stelle des P-Reglers ein PI-Regler eingesetzt wird? □

**Aufgabe 10.3** Erprobung des Diagnosealgorithmus aus Aufgabe 10.2



Implementieren Sie den in der Aufgabe 10.2 betrachteten Diagnosealgorithmus in MATLAB und untersuchen Sie, wie schnell der zur Zeit  $t = 0$  „eingeschaltete“ Fehler bei unterschiedlichem Anfangsfüllstand  $h_0$  des Behälters erkannt wird. Wie verändert sich das Diagnoseergebnis, wenn Sie den für das lineare Modell entworfenen Diagnosealgorithmus auf den geregelten nichtlinearen Reaktor

$$\begin{aligned}\dot{h}(t) &= k_P(w - h(t)) - k_f\sqrt{h(t)} - k\sqrt{h(t)} \\ q(t) &= k\sqrt{h(t)}\end{aligned}$$

anwenden, wobei  $k_f = 0$  im fehlerfreien Zustand und  $k_f = k$  im Fehlerfall gilt. Verwenden Sie für Ihre Untersuchungen die folgenden Parameter:

$k = 0,05 \frac{\text{m}^2}{\text{s}}$	Ausflusskonstante
$k_f = 0,05 \frac{\text{m}^2}{\text{s}}$	Ventilkonstante
$k_P = 0,8 \frac{\text{m}^2}{\text{s}}$	Reglerparameter
$A = 0,3 \text{ m}^2$	Grundfläche des Tanks
$h_0 = 1 \text{ m}; 2 \text{ m}; 0 \text{ m}$	Anfangsfüllstand
$w = 1,5 \text{ m}$	Füllstandssollwert $\square$

## 10.3 Sensorüberwachung

### 10.3.1 Aufgabenstellung

In diesem Abschnitt wird das Prinzip, Fehler mit Hilfe von Beobachtern zu diagnostizieren, für Sensorfehler angewendet. Typische Sensorfehler führen auf eine Drift des Messwertes, Skalierungsfehler oder Veränderungen der Ankopplung des Sensors an das zu messende Objekt. Die Messgröße kann ausfallen, wenn die Stromversorgung des Sensors unterbrochen oder die Informationskopplung zwischen dem Sensor und der Automatisierungseinrichtung ausgefallen ist.

Wenn man, wie bei vielen technischen Systemen, nicht nur mit einer einzelnen Messgröße, sondern mit vielen Messgrößen arbeitet, so kann man die in einem Teil der Messgrößen steckenden Informationen über das Systemverhalten nutzen, um die Funktionsfähigkeit der anderen Sensoren zu überprüfen. Man nutzt dabei die Redundanz der Informationen, über die man durch die Messungen und durch ein analytisches Modell des betrachteten Systems verfügt, für die Prozessüberwachung aus (*analytische Redundanz*). Im Unterschied dazu beruht der in vielen Systemen heute eingesetzte Weg der Sensordiagnose auf einer Mehrfachmessung kritischer Signale, also auf einer physikalischen Redundanz.

Das Zustandsraummodell des Systems  $\Sigma$  hat jetzt  $p$  Ausgabegleichungen für die  $p$  Ausgangsgrößen  $y_1(t), y_2(t), \dots, y_p(t)$ :

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y_i(t) = \mathbf{c}_i^T \mathbf{x}(t), & i = 1, \dots, p. \end{cases} \quad (10.7)$$

Wenn die Ausgangsgrößen zu einem  $p$ -dimensionalen Vektor zusammengefasst werden, lautet die Ausgabegleichung

$$\mathbf{y} = \mathbf{C}\mathbf{x}(t) = \begin{pmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \vdots \\ \mathbf{c}_p^T \end{pmatrix} \mathbf{x}(t), \quad (10.8)$$

wobei die Vektoren  $\mathbf{c}_1^T$  bis  $\mathbf{c}_p^T$  die  $p$  Zeilen der Matrix  $\mathbf{C}$  bilden. Ein Sensorfehler kann als additiver Fehler wie in Gl. (10.1) dargestellt werden

$$y_i(t) = \mathbf{c}_i^T \mathbf{x}(t) + f_i,$$

wobei es für die  $p$  Messgrößen  $p$  verschiedene Fehler geben kann. Die Sensorfehlerdiagnose wird hier für Fehler untersucht, die in dem für die Diagnose betrachteten Zeitintervall als konstant angesehen werden können.

### 10.3.2 Fehlerlokalisierung mit dedizierten Beobachtern

Der zur Sensorüberwachung eingesetzte Beobachter muss nicht nur für eine einzelne Ausgangsgröße  $y(t)$ , sondern für einen  $p$ -dimensionalen Ausgangsvektor  $\mathbf{y}(t)$  entworfen werden. Dies bereitet für den Entwurf des Beobachters keine Schwierigkeiten, denn das im Kap. 9 beschriebene Entwurfsverfahren kann ohne Weiteres auf einen  $p$ -dimensionalen Ausgangsvektor  $\mathbf{y}(t)$  erweitert werden. Problematisch ist die Entscheidung, welcher der eingesetzten Sensoren fehlerbehaftet ist, wenn ein Residuenvektor  $\mathbf{r}(t) = \mathbf{y}(t) - \hat{\mathbf{y}}(t)$  auftritt, bei dem i. Allg. alle Komponenten  $r_i(t)$  von null verschieden sind.

Um zu Residuen zu kommen, die eindeutig einen Fehler im  $i$ -ten Sensor anzeigen, wird die Idee, Prozessdiagnoseaufgaben mit Hilfe von Beobachtern zu lösen, für die Sensorüberwachung einen Schritt weiterentwickelt. Dieses Vorgehen ist in Abb. 10.15 für einen Prozess mit drei Messgrößen veranschaulicht. Man verwendet drei beobachtergestützte Diagnosesysteme, die mit unterschiedlichen Messinformationen versorgt werden. Der Beobachter 1 erhält nur das Messsignal  $y_1(t)$ , mit Hilfe dessen er den Prozesszustand  $\mathbf{x}(t)$  beobachtet. Er arbeitet deshalb mit dem Modell (10.7), das nur die erste Ausgangsgröße enthält

$$\Sigma_1 : \begin{cases} \dot{\mathbf{x}}_1(t) = \mathbf{A}\mathbf{x}_1(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y_1(t) = \mathbf{c}_1^T \mathbf{x}_1(t) + f_1 \end{cases}$$

und dessen Zustand zur Unterscheidung von anderen Modellen mit  $\mathbf{x}_1(t)$  bezeichnet wird. Aus dem Beobachtungsergebnis  $\hat{\mathbf{x}}_1(t)$  rekonstruiert er den zweidimensionalen Ausgangsvektor

$$\hat{\mathbf{y}}_1(t) = \begin{pmatrix} \hat{y}_{12}(t) \\ \hat{y}_{13}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{c}_2^T \\ \mathbf{c}_3^T \end{pmatrix} \hat{\mathbf{x}}_1(t),$$

der Schätzwerte für die Ausgangsgrößen  $y_2(t)$  und  $y_3(t)$  (also die nicht für die Beobachtung verwendeten Messgrößen) enthält. Die Elemente dieses Vektors werden mit den entsprechenden Messgrößen verglichen, wobei das Residuum

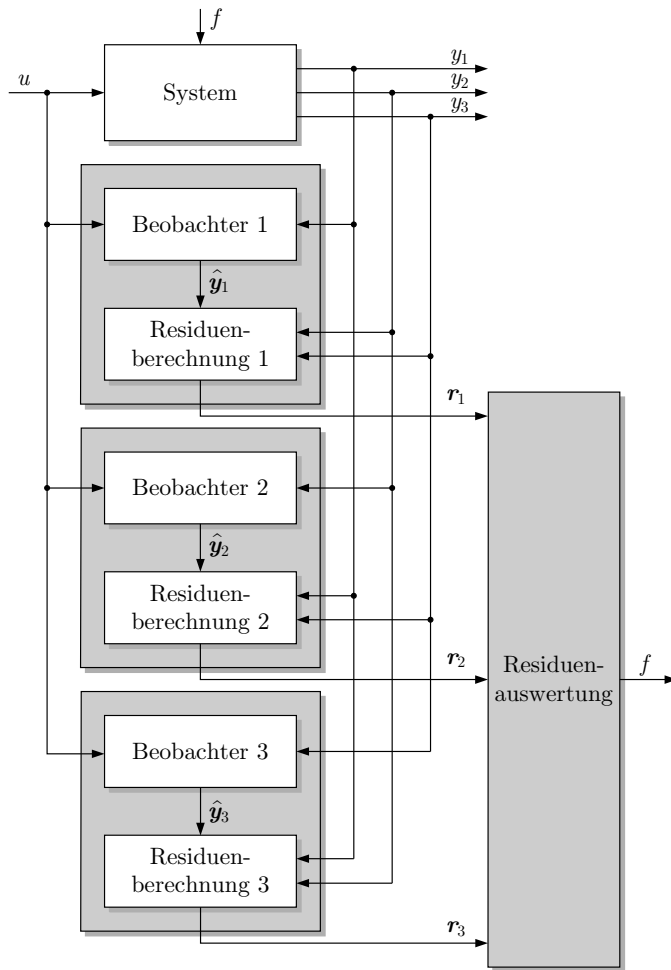
$$\mathbf{r}_1(t) = \begin{pmatrix} r_{12}(t) \\ r_{13}(t) \end{pmatrix} = \begin{pmatrix} y_2(t) - \hat{y}_{12}(t) \\ y_3(t) - \hat{y}_{13}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{c}_2^T (\mathbf{x}(t) - \hat{\mathbf{x}}_1(t)) \\ \mathbf{c}_3^T (\mathbf{x}(t) - \hat{\mathbf{x}}_1(t)) \end{pmatrix} + \begin{pmatrix} f_2 \\ f_3 \end{pmatrix}$$

entsteht. Es ist ein 2-dimensionaler Vektor. Wenn  $y_1(t)$  fehlerfrei ist ( $f_1 = 0$ ), gilt

$$\hat{\mathbf{x}}_1(t) \xrightarrow{t \rightarrow \infty} \mathbf{x}(t)$$

und folglich





**Abb. 10.15:** Sensorüberwachung mit dedizierten Beobachtern

$$\mathbf{r}_1(t) \xrightarrow{t \rightarrow \infty} \begin{pmatrix} f_2 \\ f_3 \end{pmatrix}.$$

Ist der Sensor für  $y_1(t)$  fehlerhaft ( $f_1 \neq 0$ ), so gilt  $\hat{x}_1(t) \neq x(t)$  und folglich  $\mathbf{r}_1(t) \neq \mathbf{0}$  unabhängig von  $f_2$  und  $f_3$ .

Für ein System mit  $p$  Ausgangsgrößen werden auf diese Weise  $p$  Beobachter verwendet, wobei der  $i$ -te Beobachter die  $i$ -te Messgröße  $y_i(t)$  verarbeitet und alle außer der  $i$ -ten Ausgangsgröße rekonstruiert. Die dabei erhaltenen Größen werden mit den  $p - 1$  Messgrößen verglichen, so dass jeder Residuenvektor  $p - 1$  Elemente enthält.

Für das in Abb. 10.15 gezeigte Überwachungssystem für drei Sensoren erhält man die durch Haken gekennzeichneten Abhängigkeiten der Residuen von den Fehlern

	$f_1$	$f_2$	$f_3$
$\mathbf{r}_1 = \begin{pmatrix} r_{12} \\ r_{13} \end{pmatrix}$	✓	✓	
	✓		✓
$\mathbf{r}_2 = \begin{pmatrix} r_{21} \\ r_{23} \end{pmatrix}$	✓	✓	
		✓	✓
$\mathbf{r}_3 = \begin{pmatrix} r_{31} \\ r_{32} \end{pmatrix}$	✓		✓
		✓	✓

Der Fehler  $f_i \neq 0$  führt dazu, dass der Residuenvektor  $\mathbf{r}_i(t)$  nicht verschwindet und außerdem die Komponente  $r_{ji}(t)$  des  $j$ -ten Residuenvektors den Wert  $f_i$  annimmt. Somit sind alle in der Tabelle durch einen Haken markierten Signale betroffen. Da die Tabelle drei unterschiedliche Spalten hat, können die Fehler eindeutig identifiziert werden.

Bei der Residuenauswertung fordert man, dass der Betrag der Elemente von  $\mathbf{r}_i(t)$  für endliche Zeiten vorgegebene Grenzwerte unterschreitet, um damit einerseits die Wirkung von Modellunbestimmtheiten und von Störungen zu berücksichtigen und andererseits nicht bis  $t \rightarrow \infty$  warten zu müssen. Das Diagnoseergebnis erhält man also durch gemeinsame Auswertung aller Residuen:

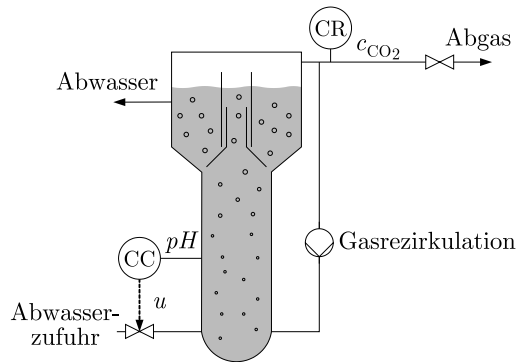
- Verschwinden alle Residuen, so arbeitet der Prozess (einschließlich aller Sensoren) fehlerfrei.
- Überschreiten alle Elemente des  $i$ -ten Residuenvektors  $\mathbf{r}_i(t)$  sowie die zur  $i$ -ten Messgröße gehörenden Elemente aller anderen Residuenvektoren ihre Grenzwerte, so ist der  $i$ -te Sensor fehlerhaft.
- Überschreiten alle Residuen die vorgegebenen Grenzwerte, so ist ein Fehler aufgetreten, der nicht in einem der Sensoren liegt.

Dieses Verfahren verwendet separate Diagnosesysteme für jeden einzelnen Sensor. Da das gesamte Diagnosesystem aus  $p$  Beobachtern besteht, spricht man auch von einer Beobachterbank. Die dem jeweiligen Sensor zugeordneten Beobachter heißen *dedizierte Beobachter* (abgeleitet vom englischen Begriff *dedicated observer*; lateinischer Wortstamm *dedicare* - widmen).

### Beispiel 10.3 Überwachung eines pH-Sensors

Bei dem in Abb. 10.16 gezeigten Biogasreaktor soll eine Überwachung des pH-Wert-Sensors dazu verwendet werden, den Zeitpunkt der nächsten Kalibrierung zu bestimmen.

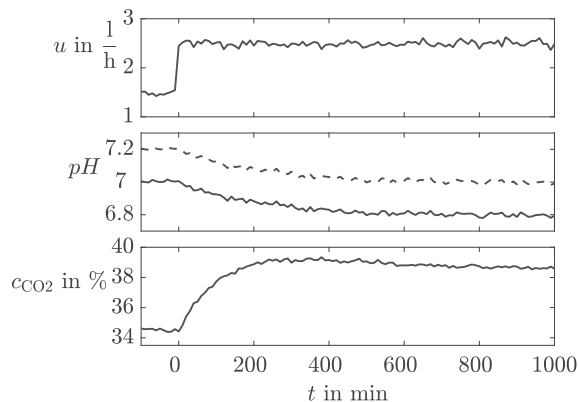
Der Reaktor dient der biologischen Reinigung von Abwasser, das am Reaktorfuß eingeleitet wird. Die organischen Verbindungen werden durch Mikroorganismen zu Gas umgesetzt, das am Reaktorkopf abgezogen wird. Das geklärte Abwasser fließt am Kopf aus dem Reaktor heraus, wobei ein Sedimenter



**Abb. 10.16:** Aufbau eines Bioreaktors

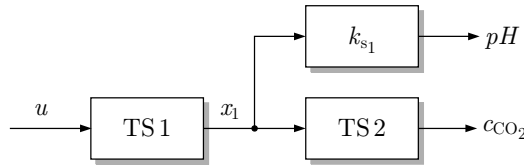
das Ausschwemmen der Biomasse aus dem Reaktor verhindert. Ein Teil des Abgases wird über eine Pumpe am Fuß in den Reaktor eingepresst, um durch die dabei entstehende Gaszirkulation die Biomasse gleichmäßig im Reaktor zu verteilen.

In den hier betrachteten Experimenten ist die Abwasserzufuhrate  $u(t)$  die Stellgröße. Gemessen werden in den mit CR und CC bezeichneten Komponenten die Konzentration  $c_{\text{CO}_2}(t)$  von Kohlendioxid im Abgas und der pH-Wert. Die Stellgröße  $u(t)$  wird für eine pH-Wert-Regelung eingesetzt, die den pH-Wert sehr genau auf dem Wert 7,0 halten muss, damit die Gasproduktion und die Abwasserklärung möglichst hoch sind.



**Abb. 10.17:** Verhalten des Biogasreaktors bei Erhöhung des Zulaufes um  $1 \frac{1}{h}$

Abbildung 10.17 zeigt ein Experiment mit dem Biogasreaktor bei abgeschalteter pH-Wert-Regelung, wobei der Abwasserzulauf sprunghaft um  $1 \frac{1}{h}$  erhöht wurde. Durch diese Stellgrößenänderung sinkt der pH-Wert ab und die  $\text{CO}_2$ -Konzentration steigt an. Im Folgenden soll erkannt werden, dass der pH-Sensor den pH-Wert um den systematischen Fehler von 0,2 falsch anzeigt, so dass er an Stelle der in Abb. 10.17 durchgezogenen Kurve den gestrichelt dargestellten Messwertverlauf liefert.



**Abb. 10.18:** Struktur des Modells für die Beschreibung des Bioreaktors

**Modellbildung.** Das für die Sensorüberwachung verwendete Modell beruht auf dem in Abb. 10.18 gezeigten Blockschaltbild. Teilsystem 1 beschreibt die Abhängigkeit der Konzentration  $x_1(t)$  des organischen und anorganischen Kohlenstoffs von der Abwasserzufuhrate  $u(t)$ . Daraus kann durch Multiplikation mit dem Faktor  $k_{s1}$  der pH-Wert berechnet werden. Das Teilsystem 2 beschreibt, wie sich die Kohlendioxidkonzentration  $c_{CO_2}(t)$  im Abgas in Abhängigkeit von der Kohlenstoffkonzentration bzw. dem dazu proportionalen pH-Wert verhält. Bei einer Veränderung des pH-Wertes verschiebt sich das Dissoziationsgleichgewicht der Kohlensäure in der Flüssigphase, was zu einem sofortigen Austrag von Kohlendioxid in die Gasphase führt, bis sich ein neues Dissoziationsgleichgewicht einstellt.

Diese Wirkungsketten führen zum Zustandsraummodell

$$\text{Biogasreaktor : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} -\frac{1}{T_1} & 0 \\ 1 & -\frac{1}{T_2} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{T_1} \\ 0 \end{pmatrix} u(t) \\ \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} x_{10} \\ x_{20} \end{pmatrix} \\ pH(t) = (k_{s1} \ 0) \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} \\ c_{CO_2}(t) = \left( \frac{k_{s2}T_0}{T_2} \ \frac{k_{s2}}{R_2} \left( 1 - \frac{T_0}{T_2} \right) \right) \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \end{array} \right. \quad (10.9)$$

dessen Ausgangsgrößen für die Parameter

$$\begin{array}{lll} k_{s1} = -0,21, & k_{s2} = 3,95 & \\ T_1 = 219, & T_2 = 165, & T_0 = 434 \end{array}$$

eine sehr gute Übereinstimmung mit den in Abb. 10.17 dargestellten Experimentergebnissen zeigen. Bei diesem Modell wird die Zeit in Minuten, die Zulauftrate in Litern pro Stunde und die  $CO_2$ -Konzentration in Prozent gemessen.  $pH(t)$  hat keine Maßeinheit. Die im Modell auftretenden Signale sind die Abweichungen vom Arbeitspunkt, der bei  $\overline{pH} = 7,0$  liegt.

**Beobachterentwurf für die Sensorüberwachung.** Für den Biogasreaktor wird jetzt ein dedizierter Beobachter entworfen, der unter Verwendung der Kohlendioxidkonzentration den pH-Wert rekonstruiert und mit dessen Messwert vergleicht. Die Struktur dieses Sensorüberwachungssystems ist in Abb. 10.19 zu sehen. Wenn das Residuum  $r(t)$  einen vorgegebenen Grenzwert  $\bar{r}$  überschreitet, so zeigt dies an, dass der pH-Sensor einen fehlerhaften Messwert liefert und kalibriert werden muss.

Ein zweiter dedizierter Beobachter für die Überwachung des  $CO_2$ -Sensors kann nicht entworfen werden, weil das System (10.9) nicht vollständig beobachtbar ist, was man mit Hilfe des Beobachtbarkeitskriteriums (Satz 5.2 auf S. 163) erkennt. Der Grund liegt darin, dass wegen der parallel geschalteten Blöcke in Abb. 10.18 der pH-Wert nicht von der Zustandsgröße  $x_2(t)$  des Teilsystems 2 abhängt.

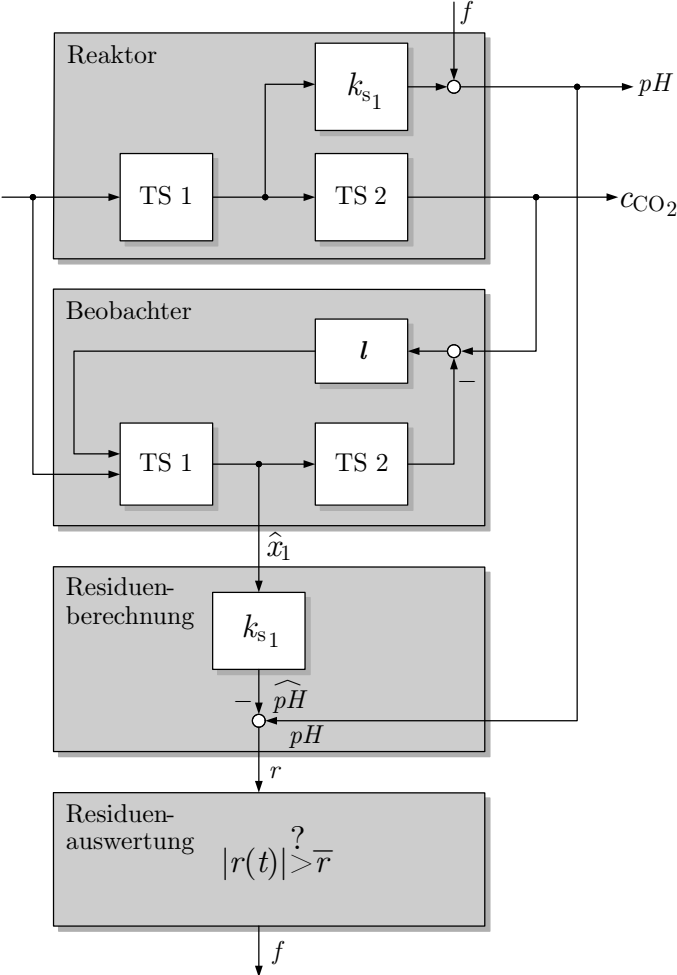


Abb. 10.19: Überwachung des pH-Sensors

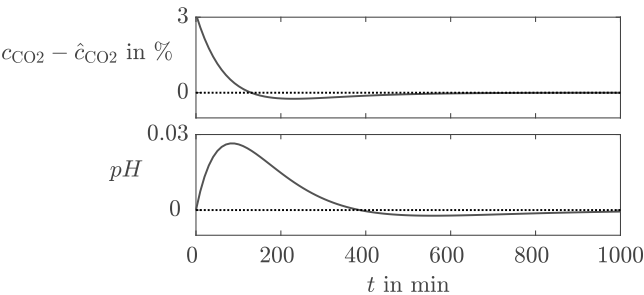


Abb. 10.20: Einschwingverhalten des dedizierten Beobachters bei fehlerfreien Sensoren

Beim Entwurf des dedizierten Beobachters für den pH-Wert wird das Modell (10.9) verwendet und die Rückführung  $l$  so festgelegt, dass die Beobachtoreigenwerte bei

$$\lambda_{B1} = \lambda_{B2} = -0,01 \frac{1}{\text{min}}$$

liegen, so dass der Beobachter doppelt so schnell einschwingt wie der Prozess. Abbildung 10.20 zeigt im oberen Teil den Verlauf des Beobachtungsfehlers  $c_{CO_2}(t) - c_{CO_{2B}}(t)$  für den Fall, dass sich der Prozess im Arbeitspunkt befindet und beim Beginn der Beobachtung eine wesentliche Abweichung zwischen den Zuständen des Prozesses und des Beobachters auftritt. Der Beobachtungsfehler verschwindet nach etwa 100 Minuten.

Aus dem Zustand  $\hat{x}_1(t)$  des Beobachters wird der pH-Wert rekonstruiert:

$$\widehat{pH}(t) = k_{s1} \hat{x}_1(t).$$

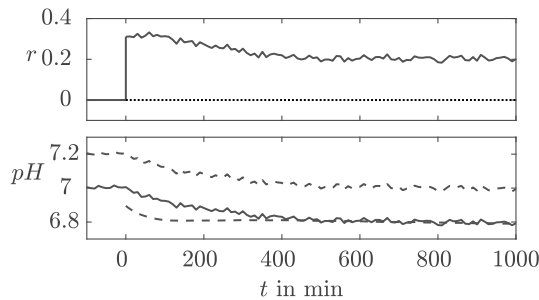
Bei fehlerfreien Sensoren nähert sich dieser Wert dem wahren Wert null innerhalb von etwa 400 Minuten an, wie der untere Teil der Abbildung zeigt. Damit verschwindet auch das Residuum

$$r(t) = pH(t) - \widehat{pH}(t).$$

**Sensorüberwachung.** Zur Erprobung des Sensorüberwachungssystems werden die Messdaten aus Abb. 10.17 um die Arbeitspunktwerte

$$\bar{u} = 1,5, \quad \overline{pH} = 7, \quad \bar{c}_{CO_2} = 34,5$$

verschoben und durch das Diagnosesystem verarbeitet, wobei der Beobachter zur Zeit  $t = 0$  zu arbeiten beginnt und den Anfangszustand  $\widehat{pH}(0) = 0,3, \hat{c}_{CO_2}(0) = 0$  hat. Die Ergebnisse sind in Abb. 10.21 dargestellt. Das im oberen Teil der Abbildung angegebene Residuum ist von Beginn an sehr groß und verschwindet nicht. Daran erkennt man den Sensorfehler.



**Abb. 10.21:** Sensordiagnose des Biogasreaktors

Zur Erklärung der Art und Weise, wie dieses Residuum zustande kommt, sind im unteren Abbildungsteil der fehlerfreie und der fehlerbehaftete Messwertverlauf aus Abbildung 10.17 dargestellt. Die untere gestrichelte Linie zeigt den vom Beobachter rekonstruierten pH-Wert  $\widehat{pH}(t)$ , der nach dem Einschwingen des Beobachters ab  $t = 400$  min den wahren pH-Wert-Verlauf sehr genau rekonstruiert. Da sich dieser Verlauf um den Wert 0,2 vom gemessenen pH-Wert unterscheidet, bleibt ein Residuum mit der Größe von 0,2, das den Sensorfehler anzeigt. Der Sensor muss gegebenenfalls neu kalibriert werden.

**Diskussion.** Das Beispiel verdeutlicht sehr anschaulich die bei der Fehlerdiagnose verwendete Vorgehensweise. Um Fehler diagnostizieren zu können, braucht man redundante Informationen. In dem Beispiel hat man zwei Messgrößen, die Informationen über das Verhalten desselben Systems enthalten. Da außerdem ein Systemmodell verfügbar ist, kann man mit der Messgröße  $c_{\text{CO}_2}(t)$  und dem Modell die Messgröße  $pH(t)$  berechnen, wofür ein Beobachter eingesetzt wird. Die Redundanz entsteht, weil man außer den Messgrößen über ein analytisches Modell des Prozesses verfügt (analytische Redundanz).

Als Alternative dazu hätte man ohne Beobachter den pH-Wert-Sensor nur dadurch überwachen können, dass man einen zweiten Sensor installiert und die beiden erhaltenen Messwerte miteinander vergleicht. Dann hätte man lediglich eine Fehlererkennung durchführen können, jedoch keine Fehleridentifikation, denn wenn beide Messwerte signifikant voneinander abweichen, weiß man zwar, dass einer der beiden Sensoren fehlerbehaftet ist, aber man weiß nicht, wo der Fehler liegt.

Entsprechend der im Abschn. 10.2 gezeigten Vorgehensweise könnte man die Diagnoseaufgabe auch dadurch lösen, dass man einen Beobachter mit dem pH-Wert als Messgröße entwirft. Dafür hätte man das Modell auf das Teilsystem 1 reduzieren müssen, um die Beobachtbarkeit sicherzustellen. Wenn der pH-Sensor fehlerbehaftet arbeitet, kann der Beobachter nicht einspringen und zeigt einen Fehler an. Da man bei diesem Vorgehen offensichtlich auf die Messgröße  $c_{\text{CO}_2}(t)$  verzichten kann, erhebt sich die Frage, welchen Vorteil der hier verwendete dedizierte Beobachter liefert, der beide Messgrößen braucht.

Die Antwort ergibt sich aus einem Vergleich der Ergebnisse, die man mit beiden Methoden erhält. Mit der im Abschn. 10.2 beschriebenen Vorgehensweise ist nur eine Fehlererkennung möglich. Wenn der Beobachtungsfehler nicht asymptotisch verschwindet, verhält sich der Biogasreaktor nicht so wie das Modell, was auf einen Fehler irgendwo im Reaktor schließen lässt. Dieser Fehler kann im pH-Sensor liegen, muss er aber nicht, denn viele andere Fehler haben eine ähnliche Wirkung.

Beim dedizierten Beobachter kann man den Fehler lokalisieren, weil man einen Beobachter mit der Messgröße  $c_{\text{CO}_2}(t)$  verwendet. Wenn dieser Beobachter einspringt, weiß man, dass der Biogasreaktor einschließlich des  $\text{CO}_2$ -Sensors fehlerfrei arbeitet. Das Residuum weist deshalb eindeutig auf einen Fehler im pH-Sensor.  $\square$

Das hier beschriebene Prinzip der Sensorüberwachung ist nur anwendbar, wenn der Prozess über jede einzelne Messgröße beobachtbar ist, denn ansonsten können die in dem Diagnosesystem verwendeten Beobachter nicht auf verschwindende Beobachtungsfehler führen. Die Sensordiagnose ist also in der beschriebenen Weise nur durchführbar, wenn die Matrix  $A$  zusammen mit jeder einzelnen Zeile  $c_i^T$ , ( $i = 1, 2, \dots, r$ ) der Ausgangsmatrix  $C$  das Beobachtbarkeitskriterium (Satz 5.2 auf S. 163) erfüllt. Die im folgenden Abschnitt angegebene Erweiterung mindert diese Voraussetzung.

#### Aufgabe 10.4\* *Intelligenter pH-Sensor*

Wenn man den Herstellern „intelligenter Sensoren“ glaubt, haben Sensoren heute eine eingebaute Intelligenz, mit der sie erkennen können, unter welchen Betriebsbedingungen der von ihnen gelieferte Messwert fehlerfrei ist. Bei diesen Sensoren müsste sich also das in diesem Abschnitt beschriebene Sensorüberwachungssystem erübrigen.

Ein intelligenter pH-Sensor liefert nicht nur das Messsignal „pH-Wert“, sondern auch ein Statussignal, das die Güte des Messwertes beschreibt. Dieses zusätzliche Signal ist in Abb. 10.22 als „pH-Sensor-Fehler“ bezeichnet. Im einfachsten Fall nimmt es zwei Werte an, von denen der Wert 1 den fehlerfreien pH-Wert und der Wert 0 einen Fehlerzustand des Sensors anzeigt.

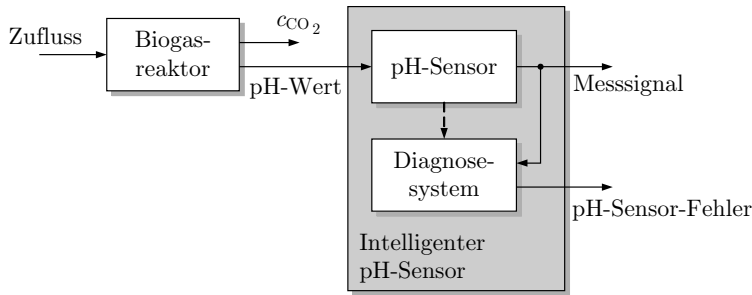


Abb. 10.22: Intelligenter pH-Sensor

Wodurch unterscheidet sich das in Abb. 10.22 gezeigte Diagnosesystem prinzipiell von dem im Beispiel 10.3 behandelten Diagnosesystem? Welche Art von Fehlern kann ein intelligenter Sensor anzeigen, welche nicht?  $\square$

### 10.3.3 Erweiterung

Das Prinzip der dedizierten Beobachter kann so verändert werden, dass man nicht mehr die Beobachtbarkeit des betrachteten Prozesses von jeder einzelnen Messgrößen fordern muss. In der in Abb. 10.23 gezeigten verallgemeinerten Beobachterstruktur für die Sensorüberwachung erhält jeder Beobachter alle bis auf eine Ausgangsgröße. Der Beobachter 1 wird deshalb für das System

$$\Sigma_1 : \begin{cases} \dot{\mathbf{x}}_1(t) = \mathbf{A}\mathbf{x}_1(t) + \mathbf{b}u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ \begin{pmatrix} y_2(t) \\ y_3(t) \\ \vdots \\ y_p(t) \end{pmatrix} = \begin{pmatrix} c_2^T \\ c_3^T \\ \vdots \\ c_p^T \end{pmatrix} \mathbf{x}_1(t) \end{cases}$$

entworfen. Wenn die Messgrößen  $y_2(t)$  bis  $y_p(t)$  fehlerfrei sind, wird der Beobachter einschwingen und die Ausgangsgröße  $y_1(t)$  entsprechend  $\hat{y}_1(t) = \mathbf{c}_1^T \hat{\mathbf{x}}_1(t)$  richtig rekonstruieren, so dass das Residuum

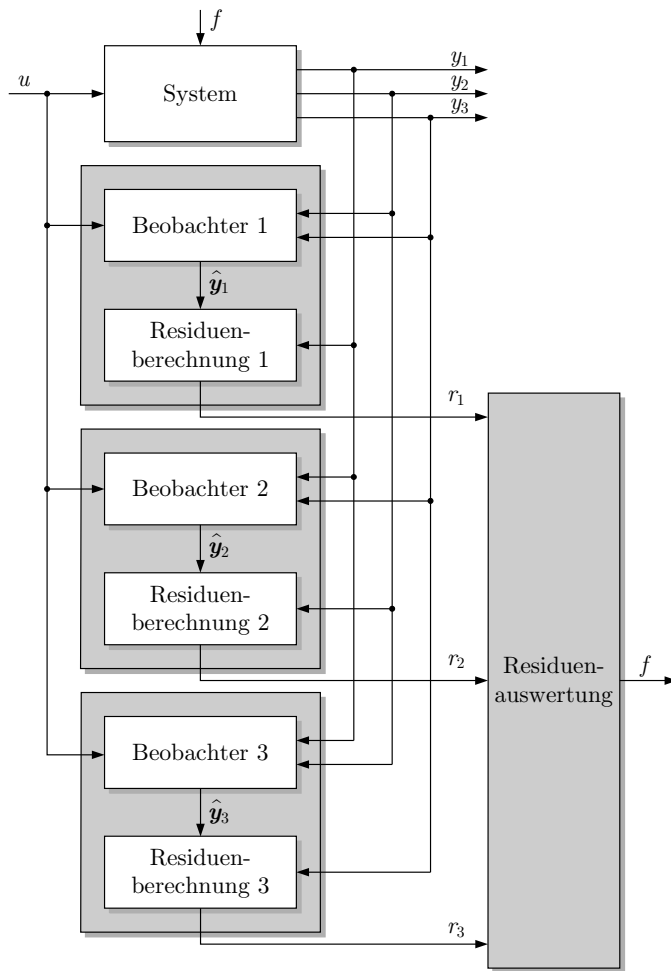
$$r_1(t) = y_1(t) - \hat{y}_1(t)$$

für  $t \rightarrow \infty$  verschwindet.

Wenn der Sensor für  $y_1(t)$  fehlerbehaftet ist, werden alle Beobachter bis auf den ersten, der entsprechend Abb. 10.23 als einziger nicht mit der Messgröße  $y_1(t)$  versorgt wird, einen verschwindenden Beobachtungsfehler erzeugen. Mehrfachfehler der Sensoren können auf diese Weise nicht erkannt werden.

Eine genauere Analyse dieser Vorgehensweise zeigt zwei Vorteile. Erstens muss bei diesem Vorgehen nur gesichert sein, dass der Prozess über alle  $(p-1)$ -dimensionalen Ausgangsvektoren beobachtbar ist, die man durch Herausstreichen einer Ausgangsgröße aus dem Ausgangsvektor  $\mathbf{y}(t)$  erhält. Diese Forderung ist natürlich wesentlich schwächer als die beim dedizierten





**Abb. 10.23:** Sensorüberwachung mit einer verallgemeinerten Beobachterbank

Beobachter aufgestellte Bedingung, dass der Prozess über jede einzelne Messgröße beobachtbar sein muss. Zweitens kann man diese Beobachterbank so entwerfen, dass die Fehlerlokalisierung auch unter der Wirkung von  $p - 1$  Störgrößen möglich ist.

## 10.4 Fehleridentifikation

### 10.4.1 Fehleridentifikation mit einer Beobachterbank

Im Abschnitt 10.2 wurde mit Beobachtern nur eine Fehlererkennung, aber keine genaue Bestimmung der Größe des Fehlers vorgenommen. Auch konnte nicht zwischen unterschiedlichen Fehlern unterschieden werden, denn es wurde nur mit dem Modell des fehlerfreien Systems gearbeitet. In diesem Abschnitt wird das Diagnoseprinzip für die Fehleridentifikation erweitert.

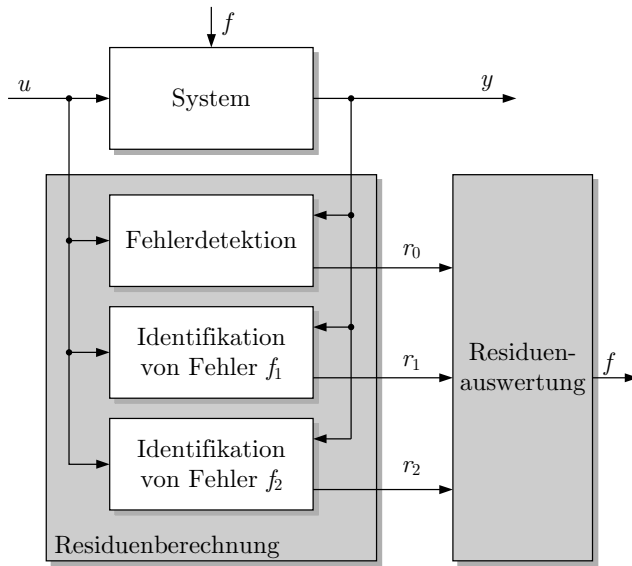


Abb. 10.24: Fehleridentifikation mit einer Beobachterbank

Dafür wird die im Abschnitt 10.3 beschriebene Beobachterbank von Sensorfehlern auf beliebige Fehler übertragen, wobei das in Abb. 10.24 gezeigte Diagnosesystem entsteht. Es werden Beobachter für das fehlerfreie System sowie für alle betrachteten Fehler entworfen und als Beobachterbank zusammengeschaltet. Dabei erhält man mehrere Residuen, wobei das Residuum  $r_0(t)$  durch den Beobachter erzeugt wird, der mit dem fehlerfreien Modell arbeitet, und die Residuen  $r_i(t)$  die Beobachter, die die Fehler  $f_i$ , ( $i = 1, 2, \dots, q$ ) berücksichtigen.

Bei der Residuenauswertung sind für ein System mit zwei Fehlern folgende Fälle zu unterscheiden:

- $r_0(t) = 0, r_1(t) \neq 0, r_2(t) \neq 0$ :

Das Systemverhalten ist konsistent mit dem Modell des fehlerfreien Systems, aber inkonsistent mit den Modellen für die Fehler  $f_1$  und  $f_2$ . Daraus wird geschlossen, dass das System fehlerfrei ist.

- $r_0(t) \neq 0, r_1(t) = 0, r_2(t) \neq 0$ :  
Das Systemverhalten ist fehlerbehaftet und konsistent mit dem Modell, das den Fehler  $f_1$  berücksichtigt. Daraus wird geschlossen, dass der Fehler  $f_1$  aufgetreten ist. In ähnlicher Weise schließt man auf den Fehler  $f_2$ , wenn  $r_0(t) \neq 0$  und  $r_1(t) \neq 0, r_2(t) = 0$  gilt.
- $r_0(t) \neq 0, r_1(t) \neq 0, r_2(t) \neq 0$ :  
Das Systemverhalten ist mit allen verwendeten Modellen inkonsistent. Daraus wird geschlossen, dass das System fehlerhaft ist, aber weder der Fehler  $f_1$  noch der Fehler  $f_2$  eingetreten ist.

Diese wie auch die an anderer Stelle aufgeführten Schlussfolgerungen gelten natürlich immer nur unter der Annahme, dass keine Fehler auftreten, unter deren Wirkung sich der Prozess entweder wie im fehlerfreien Fall oder wie beim Auftreten eines der hier betrachteten Fehler verhält. Die im ersten Fall aufgestellte Schlussfolgerung, dass das System fehlerfrei ist, gilt eben nur, wenn es keinen Fehler  $f_3$  gibt, bei dem sich das System wie im fehlerfreien Fall verhält und der Fehler  $f_3$  folglich nicht diagnostizierbar ist.

Mit der hier beschriebenen Vorgehensweise kann man mehrere Fehler unterscheiden. Das Diagnosesystem ist jedoch sehr aufwändig, weil für jeden Fehler ein eigener Beobachter entworfen und realisiert werden muss.

#### 10.4.2 Fehleridentifikation unter Verwendung von Fehlermodellen

Wenn sich der Fehler zeitlich ändert, wird er durch ein Signal  $f(t)$  dargestellt. Dann können bei der Fehleridentifikation Informationen über den typischen Verlauf von  $f(t)$  ausgenutzt werden.

Bei der folgenden Erläuterung werden Systeme betrachtet, deren Fehler als eine unbekannte Eingangsgröße  $f(t)$  interpretiert werden können, so dass sie sich durch ein Zustandsraummodell (10.2) beschreiben lassen. Da aus technischen Gründen nicht jeder beliebige Verlauf der Eingangsgröße  $f(t)$  möglich bzw. plausibel ist, beschreibt man die Menge der möglicherweise auftretenden Fehlersignale  $f(t)$  durch ein Zustandsraummodell

$$\dot{\mathbf{x}}_F(t) = \mathbf{A}_F \mathbf{x}_F(t), \quad \mathbf{x}_F(0) = \mathbf{x}_{F0} \quad (10.10)$$

$$f(t) = \mathbf{c}_F^T \mathbf{x}_F(t) \quad (10.11)$$

mit der dynamischen Ordnung  $n_F$ , dessen Matrix  $\mathbf{A}_F$  und Vektor  $\mathbf{c}_F^T$  die Klasse der möglichen Fehlersignale festlegen. Dieses Modell wird als *Fehlermodell* bezeichnet. Entsprechend Gl. (4.9) auf S. 127 schränkt das Fehlermodell die möglichen Fehlersignale auf Funktionen der Form

$$f(t) = \mathbf{c}_F^T e^{\mathbf{A}_F t} \mathbf{x}_{F0} \quad (10.12)$$

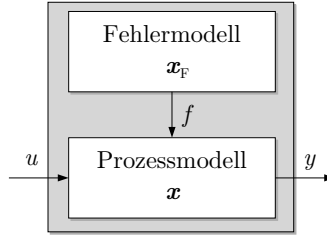
mit beliebigem (unbekanntem) Anfangszustand  $\mathbf{x}_{F0}$  ein. Wenn man beispielsweise annimmt, dass der Fehler nur „ein- oder ausgeschaltet“ sein kann, so ist  $f(t)$  ein sprungförmiges Signal, das durch das Modell erster Ordnung

$$\dot{\mathbf{x}}_F(t) = 0, \quad \mathbf{x}_F(0) = \mathbf{x}_{F0} \quad (10.13)$$

$$f(t) = c_F x_F(t) \quad (10.14)$$

beschrieben ist. Dann gilt

$$f(t) = c_F x_{F0}.$$



**Abb. 10.25:** Erweitertes Prozessmodell

Bei der Fehlerdiagnose ist der Anfangszustand  $x_{F0}$  des Fehlermodells nicht bekannt, sondern muss aus dem Verlauf der Eingangs- und Ausgangsgröße ermittelt werden. Wie man leicht sieht, ist diese Aufgabe eine Beobachtungsaufgabe. Für ihre Lösung kombiniert man das Fehlermodell (10.10), (10.11) mit dem Zustandsraummodell (10.2) des Prozesses zum erweiterten Modell

$$\frac{d}{dt} \begin{pmatrix} x(t) \\ x_F(t) \end{pmatrix} = \underbrace{\begin{pmatrix} A & g c_F^T \\ O & A_F \end{pmatrix}}_{\tilde{A}} \underbrace{\begin{pmatrix} x(t) \\ x_F(t) \end{pmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{pmatrix} b \\ 0 \end{pmatrix}}_{\tilde{b}} u(t), \quad \begin{pmatrix} x(0) \\ x_F(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ x_{F0} \end{pmatrix} \quad (10.15)$$

$$y(t) = \underbrace{(c^T \quad 0^T)}_{\tilde{c}^T} \underbrace{\begin{pmatrix} x(t) \\ x_F(t) \end{pmatrix}}_{\tilde{x}(t)}. \quad (10.16)$$

Das System zu diagnostizieren heißt, für dieses erweiterte Modell den Anfangszustand und dabei insbesondere die Komponente  $x_{F0}$  dieses Anfangszustands zu bestimmen (Abb. 10.25). Dabei entsteht ein Beobachter entsprechend Gl. (9.7)

$$\frac{d\hat{x}(t)}{dt} = (\tilde{A} - \tilde{l}\tilde{c}^T)\hat{x}(t) + \tilde{b}u(t) + \tilde{l}y(t), \quad \hat{x}(0) = \hat{x}_0 \quad (10.17)$$

dessen Zustand  $\hat{x}(t)$  Schätzwerte für den aus dem Prozesszustand  $x(t)$  und dem Zustand  $x_F(t)$  des Fehlermodells bestehenden Zustand  $\tilde{x}(t)$  liefert. Da für diesen Beobachter bei geeigneter gewählter Rückführung  $\tilde{l}$  der Beobachtungsfehler asymptotisch verschwindet, kann man durch

$$\hat{f} = (0^T \quad c_F^T)\hat{x}(t) \quad (10.18)$$

den Fehler aus dem Beobachterzustand ablesen.

Die Fehleridentifikation beruht bei diesem Vorgehen auf der Anpassung des Modellzustands an das gemessene Systemverhalten. Es wird also nicht die Inkonsistenz zwischen dem Modell und den Messgrößen geprüft, sondern die Inkonsistenz durch Anpassung der Zustände des

Fehlermodells an das in den Messungen von  $u(t)$  und  $y(t)$  sichtbar werdende Systemverhalten beseitigt. Dann wird der Fehler aus dem Anfangszustand des Fehlermodells abgelesen.

**Diagnostizierbarkeit.** Für den praktischen Einsatz der Prozessdiagnose ist es wichtig zu wissen, ob ein betrachteter Fehler  $f$  überhaupt durch eine geeignete Auswertung der im Eingang  $u(t)$  und im Ausgang  $y(t)$  enthaltenen Informationen erkannt werden kann. Für die hier betrachteten beobachtergestützten Diagnoseverfahren kann über die Diagnostizierbarkeit mit Hilfe einer Beobachtbarkeitsanalyse entschieden werden. Das Paar  $(\tilde{A}, \tilde{c}^T)$  muss vollständig beobachtbar sein, damit der Zustand  $\tilde{x}(t)$  des erweiterten Prozessmodells aus dem Eingang  $u(t)$  und dem Ausgang  $y(t)$  berechnet und daraus der Fehler abgelesen werden kann.

Die Analyse zeigt, dass die Diagnostizierbarkeit (wie die Beobachtbarkeit) eine Eigenschaft des fehlerbehafteten Systems ist und nicht davon abhängt, wie der Fehler diagnostiziert wird. Diese Eigenschaft hängt insbesondere nicht davon ab, wie die Beobachterrückführung  $l$  gewählt wird. Maßgebend sind die durch die Matrix  $A$  und die Vektoren  $g$  und  $c^T$  beschriebenen Systemeigenschaften, wobei  $g$  angibt, in welcher Weise der Fehler  $f$  das System beeinflusst. Die Diagnostizierbarkeit wird außerdem durch das Fehlermodell beschrieben, dessen Eigenschaften in der Matrix  $A_F$  und im Vektor  $c_F^T$  dargestellt sind. Andererseits ist sie unabhängig von der Fehleramplitude, die durch den Anfangszustand  $x_{F0}$  des Fehlermodells wiedergegeben wird.

### 10.4.3 Entwurf beobachtergestützter Diagnosesysteme

Die Vorgehensweise, Fehler mit Hilfe von Beobachtern zu identifizieren, kann in folgendem Algorithmus zusammengefasst werden:

---

#### Algorithmus 10.2 Entwurf eines Beobachters zur Fehleridentifikation

---

**Gegeben:** Zu diagnostizierender Prozess (10.2)  
Fehlermodell (10.10), (10.11).

1. Bilden Sie aus dem Prozessmodell und dem Fehlermodell das erweiterte Prozessmodell (10.15), (10.16).
2. Prüfen Sie die Diagnostizierbarkeit der Fehler durch eine Beobachtbarkeitsanalyse des Paares  $(\tilde{A}, \tilde{c}^T)$ .
3. Wählen Sie  $n + n_F$  Beobachtereigenwerte  $\lambda_{Bi}$ .
4. Berechnen Sie die Beobachterrückführung  $l$  aus Gl. (9.12).

**Ergebnis:** Diagnosesystem (10.17), (10.18), das das Fehlersignal  $\hat{f}(t)$  nach Gl. (10.5) erzeugt.

---

Durch Verarbeitung der Signale  $u_{[0,t_e]}$  und  $y_{[0,t_e]}$  mit dem Diagnosesystem (10.17), (10.18) erhält man das Fehlersignal  $\hat{f}_{[0,t_e]}$ , das das Auftreten eines oder mehrerer Fehler anzeigt. Das

Residuum, als das man das Signal  $\hat{f}(t)$  interpretieren kann, zeigt bei diesem Vorgehen den Fehler direkt an.

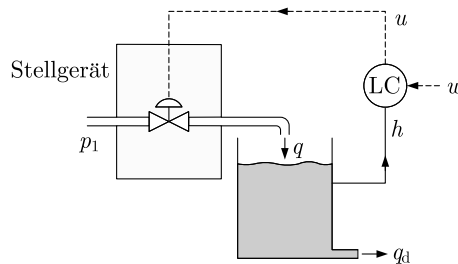


Abb. 10.26: Füllstandsregelkreis (LC – level controller)

#### Beispiel 10.4 Fehlerdiagnose eines Servoventils

Um eine Regelung realisieren zu können, braucht man zuverlässige Stellgeräte, mit denen der Regler das Verhalten der Regelstrecke beeinflussen kann. Bei dem in Abb. 10.26 gezeigten Füllstandsregelkreis verändert der Regler mit Hilfe des Servoventils den Zufluss  $q(t)$  zum Reaktor, so dass der Füllstand  $h(t)$  auch bei einer Flüssigkeitsentnahme  $q_d(t)$  auf der vorgegebenen Höhe  $w$  verbleibt. Das Blockschaltbild des Regelkreises zeigt Abb. 10.27.

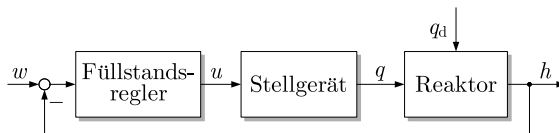
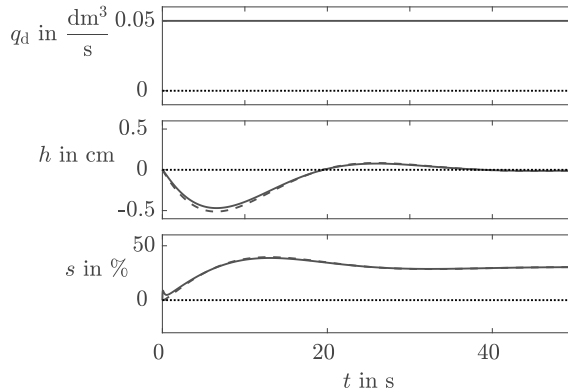


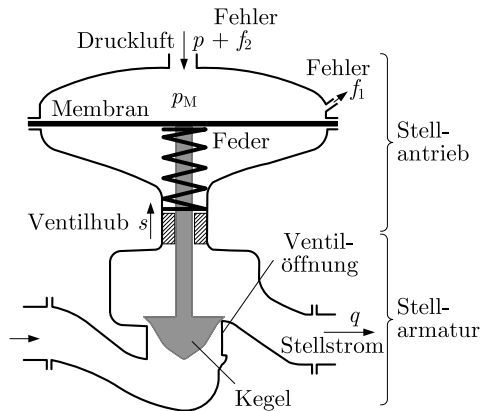
Abb. 10.27: Blockschaltbild des Füllstandsregelkreises

In diesem Beispiel wird die Fehlerdiagnose des Stellgerätes behandelt. Da Fehler im Stellglied durch eine entsprechende Reaktion des Reglers ausgeglichen werden können, sind derartige Fehler, zumindest solange sie noch hinreichend klein sind, schwer aus dem Verhalten des Regelkreises ablesbar. Abbildung 10.28 zeigt das Verhalten des Füllstandsregelkreises bei dem im oberen Abbildungsteil angegebenen Verlauf der Störung  $q_d(t)$ . Im zweiten Teil ist die Abweichung  $h(t)$  des Füllstands vom konstanten Sollwert  $w(t) = 0$  und im dritten Teil der Ventilhub  $s(t)$  um den Arbeitspunkt gezeigt. Zum Zeitpunkt  $t = 0$  tritt ein über den gesamten Zeitraum wirkender Fehler auf. Wie die Abbildung zeigt, macht sich dieser Fehler nur über eine kurze Zeitspanne im Füllstand  $h(t)$  und im Ventilhub  $s(t)$  bemerkbar und ist dann auf Grund der Wirkung der Regelung nicht mehr sichtbar. Die einzige bleibende Änderung betrifft die Stellgröße  $u(t)$ , die der Füllstandsregler dem veränderten Verhalten des Ventils anpasst. Da das Stellsignal jetzt etwas größer als im fehlerfreien Fall ist, verhält sich der Ventilhub  $s(t)$  des fehlerbehafteten Ventils wie der des fehlerfreien Ventils und der Füllstand  $h(t)$  folgt der alten Trajektorie. Wie die folgenden Betrachtungen zeigen, ist die Fehleridentifikation trotz der geringen Verhaltensänderungen im Regelkreis möglich, ohne dass zusätzliche Messungen durchgeführt werden müssen.



**Abb. 10.28:** Verhalten des Füllstandsregelkreises mit fehlerfreiem Stellgerät (---) und nach Auftreten eines Lecks im Ventilgehäuse (Fehler  $f_1$  —)

**Funktionsweise des Servoventils.** Es muss zunächst der Aufbau des Stellgerätes erläutert werden. Abbildung 10.29 zeigt den Stellantrieb und die Stellarmatur. Der Volumenstrom  $q(t)$  des im unteren Teil der Abbildung von links nach rechts fließenden Stoffes wird durch die Ventilöffnung beeinflusst, die vom Hub  $s(t)$  der Antriebsstange abhängt. Die Antriebsstange wird durch die von oben mit dem Druck  $p_M(t)$  auf eine Membran drückende Luft sowie eine Feder bewegt.



**Abb. 10.29:** Aufbau eines Servoventils

Der Druck  $p(t)$  der Luft, die das Ventil bewegt, wird durch einen Stellungsregler beeinflusst, der mit dem Stellantrieb den in Abb. 10.30 gezeigten Regelkreis bildet. Auf Grund dieser im Stellgerät enthaltenen Regelung ist das Ventil ein Servoventil. Die Stellgröße  $u(t)$  des Füllstandsreglers stellt den Sollwert des Stellungsreglers und der Luftdruck  $p(t)$  oberhalb der Membran die Stellgröße des Stellungsreglers dar (Abb. 10.26 und 10.30). Als Stellungsregler wird häufig ein P-Regler

$$p(t) = -k_{PV}(s(t) - u(t)) \quad (10.19)$$

eingesetzt. Da der Hub nach oben positiv gezählt wird, muss der Regler den Druck  $p(t)$  bei positiver Regelabweichung  $s(t) - u(t)$  erhöhen, um  $s$  zu verkleinern. Aus dieser Überlegung resultiert das positive Vorzeichen auf der rechten Seite des Reglergesetzes (10.19). Dass ein P-Regler den Hub  $s(t)$  nicht ohne eine bleibende Regelabweichung an den Sollwert  $u(t)$  angleichen kann, spielt bei Servoventilen keine Rolle, weil der Füllstandsregler diese bleibende Regelabweichung durch eine Anpassung der Stellgröße  $u(t)$  ausgleicht.

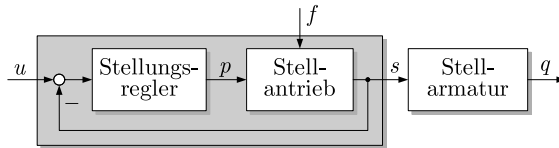


Abb. 10.30: Regelkreis des Servoventils

**Modell des Servoventils.** Bei der Beschreibung des Stellantriebs muss man zwei dynamische Elemente berücksichtigen. Dass sich der Druck  $p_M(t)$  oberhalb der Membran nur mit einiger Verzögerung dem durch den Regler vorgegebenen Druck  $p(t)$  angleichen kann, wird durch ein System erster Ordnung mit der Zeitkonstanten  $T_D$  beschrieben:

$$\dot{p}_M(t) = -\frac{1}{T_D}p_M(t) + \frac{1}{T_D}p(t) \quad \text{mit } T_D = 1 \text{ s.} \quad (10.20)$$

Für die Bewegung der Antriebsstange mit dem Kegel erhält man die Differentialgleichung zweiter Ordnung

$$m\ddot{s}(t) + k_R\dot{s}(t) + k_F s(t) = -k_N A_M p_M(t) \quad (10.21)$$

mit den Parametern

$$\begin{aligned} m &= 2,5 \text{ kg} && \text{Masse der Antriebsstange und des Kegels} \\ k_R &= 1500 \frac{\text{Ns}}{\text{m}} && \text{Koeffizient für die Reibung in der Packung} \\ k_F &= 60000 \frac{\text{N}}{\text{m}} && \text{Federkonstante} \\ A_M &= \frac{\pi}{4} 0,2^2 \text{ m}^2 && \text{Membranfläche.} \end{aligned}$$

Der Faktor  $k_N = 1,91 \cdot 10^6$  wurde eingeführt, um eine normierte Darstellung der im Stellantrieb wirkenden Luftdrücke  $p(t)$  und  $p_M(t)$  sowie des Hubs  $s(t)$  zu erreichen. Alle drei Größen liegen damit im Intervall  $-0,5 \dots +0,5$  um ihre Arbeitspunktwerte. Das negative Vorzeichen auf der rechten Seite der Differentialgleichung resultiert aus der Tatsache, dass eine Druckerhöhung eine Verkleinerung des Hubs bewirkt.

**Fehler.** Es werden zwei Fehler betrachtet. Der erste Fehler besteht in einer undichten Stelle im Stellantrieb oberhalb der Membran bzw. in der Membranabdichtung, wodurch sich der Druck  $p_M(t)$  um bis zu 10% verringert. Da sich das Ventil in der Nähe des Arbeitspunkts bewegt, ändert sich der Druck  $p_M(t)$  nur geringfügig im Vergleich zu seinem Absolutwert und der durch das Leck hervorgerufene Druckabfall wird als konstante Druckminderung um den Wert  $f_1(t)$  angesetzt. Durch den Fehler mindert sich die auf die Membran wirkende Kraft, so dass an Stelle von Gl. (10.21) jetzt die Gleichung

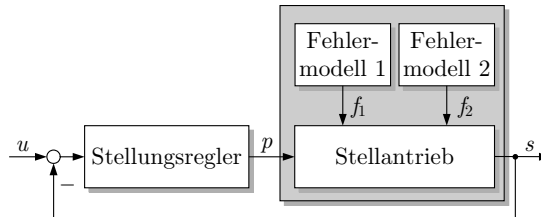
$$m\ddot{s}(t) + k_R\dot{s}(t) + k_F s(t) = -k_N A_M (p_M(t) - f_1(t)) \quad (10.22)$$

gilt. Der Fehler  $f_1(t)$  ist konstant und wird durch das Fehlermodell



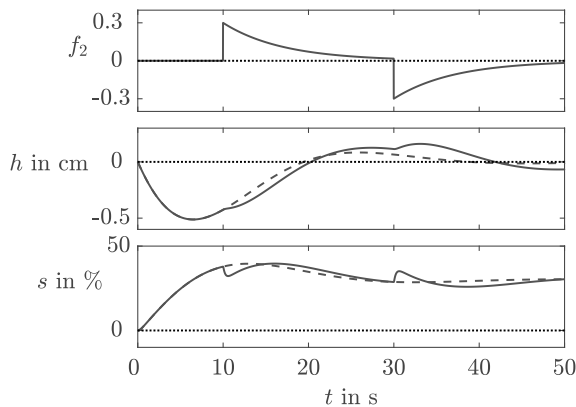
$$\dot{f}_1(t) = 0, \quad f_1(0) = f_{10} \quad (10.23)$$

beschrieben, wobei bei der Fehlerdiagnose die Größe  $f_{10}$  nicht bekannt ist und bei den Simulationsuntersuchungen  $f_{10} = 0,1$  gesetzt wird (Abb. 10.31).



**Abb. 10.31:** Stellgerät mit Fehlermodell

Wie Abb. 10.28 zeigt, ist die Wirkung des Fehlers nur über eine kurze Zeitspanne im Regelkreisverhalten sichtbar, weil der Füllstandsregler die Wirkung des Fehlers dadurch ausgleicht, dass er eine größere Stellgröße  $u(t)$  vorgibt, mit der das Ventil denselben Durchfluss wie im fehlerfreien Zustand erzeugt.



**Abb. 10.32:** Verhalten des Füllstandsregelkreises mit (—) und ohne Druckschwankung (---) in der Versorgungsleitung des Ventils

Der zweite Fehler stellt Druckschwankungen in der Versorgungsleitung des Ventils dar. Der Ausgang  $p(t)$  des Stellungsreglers gibt an, wie stark der Versorgungsdruck gemindert werden soll. Schwankt der Versorgungsdruck, so wirkt sich dies auf den Membrandruck des Ventils aus. An Stelle von Gl. (10.19) gilt jetzt

$$p(t) = -k_{PV}(s(t) - u(t)) + f_2(t). \quad (10.24)$$

Der Fehler  $f_2(t)$  wird als eine abklingende e-Funktion

$$f_2(t) = f_{20} e^{-\frac{t}{T_{F2}}}$$

beschrieben, die die Eigenbewegung des Fehlermodells

$$\dot{f}_2(t) = -\frac{1}{T_{F2}}f_2(t), \quad f_2(0) = f_{20} \quad (10.25)$$

ist.

Abbildung 10.32 zeigt das Regelkreisverhalten bei dem im obersten Teil der Abbildung gezeigten Verlauf des Fehlers  $f_2(t)$ . Es wurde derselbe Verlauf der Störung  $q_d(t)$  wie in Abb. 10.28 verwendet, d. h., dass die Störung über den betrachteten Zeithorizont konstant ist und die Regelung die Wirkung dieser Störung auf die Regelgröße  $h(t)$  abbaut, wie es im zweiten Abbildungsteil mit der gestrichelten Linie gezeigt wird. Durch den Druckanstieg zur Zeit  $t = 10$  s vermindert sich der Hub gegenüber dem fehlerfreien Verhalten, so dass der Füllstand  $h(t)$  nicht so schnell ansteigt wie im fehlerfreien Fall und der Füllstandsregler eine größere Stellgröße  $u(t)$  erzeugt. Beim Druckabfall zur Zeit  $t = 30$  s treten die Änderungen mit entgegengesetztem Vorzeichen auf. Auch hier sind die Änderungen im Regelkreisverhalten relativ klein.

**Beobachterentwurf für die Fehleridentifikation.** Für die Diagnose wird das Modell des fehlerbehafteten Stellgerätes benötigt, das man aus den Gln. (10.20), (10.22) und (10.24) erhält:

$$\begin{aligned} \begin{pmatrix} \dot{p}_M(t) \\ \dot{s}(t) \\ \dot{s}(t) \end{pmatrix} &= \begin{pmatrix} -\frac{1}{T_D} & -\frac{k_{PV}}{T_D} & 0 \\ 0 & 0 & 1 \\ -\frac{k_{NA}A_M}{m} & -\frac{k_F}{m} & -\frac{k_R}{m} \end{pmatrix} \begin{pmatrix} p_M(t) \\ s(t) \\ \dot{s}(t) \end{pmatrix} + \begin{pmatrix} -\frac{k_{PV}}{T_D} \\ 0 \\ 0 \end{pmatrix} u(t) \\ &+ \begin{pmatrix} 0 \\ 0 \\ \frac{k_{NA}A_M}{m} \end{pmatrix} f_1(t) + \begin{pmatrix} \frac{1}{T_D} \\ 0 \\ 0 \end{pmatrix} f_2(t) \\ s(t) &= (0 \ 1 \ 0) \begin{pmatrix} p_M(t) \\ s(t) \\ \dot{s}(t) \end{pmatrix} \\ q(t) &= (0 \ k_{Qs} \ 0) \begin{pmatrix} p_M(t) \\ s(t) \\ \dot{s}(t) \end{pmatrix}. \end{aligned}$$

Es wird als

$$\dot{\mathbf{x}}_V(t) = \mathbf{A}_V \mathbf{x}_V(t) + \mathbf{b}_V u(t) + \mathbf{g}_{V1} f_1(t) + \mathbf{g}_{V2} f_2(t) \quad (10.26)$$

$$s(t) = \mathbf{c}_V^T \mathbf{x}_V(t) \quad (10.27)$$

$$q(t) = k_{Qs} s(t) \quad (10.28)$$

abgekürzt. In der zweiten Ausgangsgleichung bezeichnet  $k_{Qs} = 0,166$  den Übertragungsfaktor der Stellarmatur.

Für den Diagnosealgorithmus ist die Verknüpfung des Ventilmodells (10.26), (10.27) mit den Fehlermodellen (10.23) und (10.25) maßgebend, wobei man

$$\begin{pmatrix} \dot{\mathbf{x}}_V(t) \\ \dot{f}_1(t) \\ \dot{f}_2(t) \end{pmatrix} = \begin{pmatrix} \mathbf{A}_V & \mathbf{g}_{V1} & \mathbf{g}_{V2} \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & -\frac{1}{T_{F2}} \end{pmatrix} \begin{pmatrix} \mathbf{x}_V(t) \\ f_1(t) \\ f_2(t) \end{pmatrix} + \begin{pmatrix} \mathbf{b}_V \\ 0 \\ 0 \end{pmatrix} u(t) \quad (10.29)$$

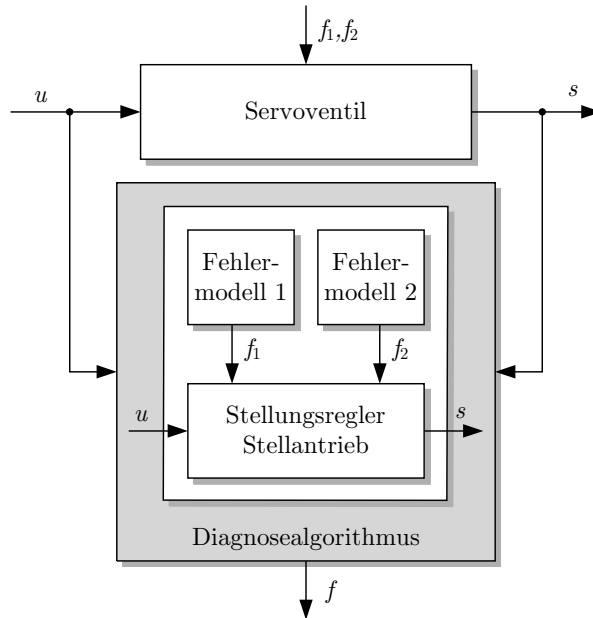
$$\begin{pmatrix} \mathbf{x}_V(0) \\ f_1(0) \\ f_2(0) \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{V0} \\ f_{10} \\ f_{20} \end{pmatrix}$$

$$s(t) = (\mathbf{c}_V^T \ 0 \ 0) \begin{pmatrix} \mathbf{x}_V(t) \\ f_1(t) \\ f_2(t) \end{pmatrix} \quad (10.30)$$

erhält. Dieses Modell wird in

$$\begin{aligned}\frac{d}{dt}\tilde{\mathbf{x}}(t) &= \tilde{\mathbf{A}}\tilde{\mathbf{x}}(t) + \tilde{\mathbf{b}}u(t), \quad \tilde{\mathbf{x}}(0) = \tilde{\mathbf{x}}_0 \\ s(t) &= \tilde{\mathbf{c}}^T\tilde{\mathbf{x}}(t)\end{aligned}$$

umgeschrieben, womit es die Form des Modells (10.15), (10.16) bekommt und wie dieses behandelt werden kann. Es beschreibt das in Abbildung 10.31 gezeigte Stellgerät einschließlich der Fehlermodelle.

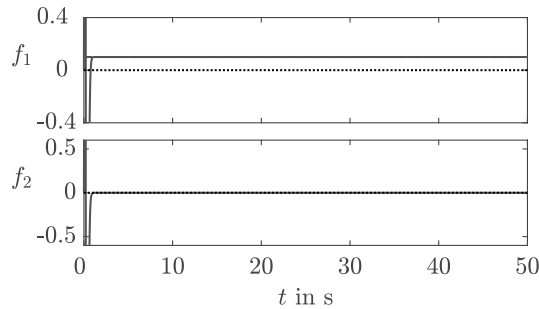


**Abb. 10.33:** Blockschaltbild für die Diagnose des Servoventils mit Kennzeichnung des für die Diagnose verwendeten Modells

Der Beobachter wurde für dieses Beispiel so entworfen, dass er einen fünffachen Eigenwert bei  $-20 \frac{1}{s}$  besitzt. Das Diagnoseergebnis erhält man durch Betrachtung der letzten beiden Komponenten des Zustandsvektors  $(\mathbf{x}_V(t)^T \ f_1(t) \ f_2(t))^T$ . Damit kann das Ventil um ein Diagnosesystem ergänzt werden, das nur mit der Stellgröße  $u(t)$  und dem bereits für den Stellungsregler gemessenen Hub  $s(t)$  arbeitet (Abb. 10.33), also keine zusätzlichen Messungen erfordert.

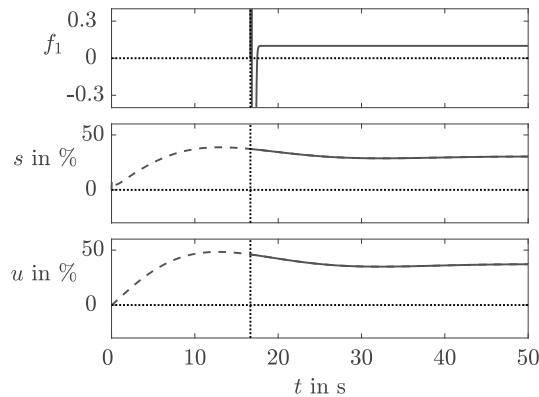
**Diagnose des Servoventils.** Abbildung 10.34 zeigt das Diagnoseergebnis für den Fall, dass der Fehler  $f_1(t)$  bei  $t = 0$  wirksam wird, also gerade zu dem Zeitpunkt, an dem der Diagnosealgorithmus startet. Der Füllstandsregelkreis verhält sich in dieser Situation entsprechend Abb. 10.28. Der Beobachter schwingt schnell ein und erkennt den Fehler mit der richtigen Amplitude  $f_1(t) = 0,1$  eindeutig.

Man kann vermuten, dass das Diagnoseergebnis so gut ist, weil der plötzliche Druckabfall zur Zeit  $t = 0$  eine sofortige Vergrößerung des Hubs mit sich bringt (vgl. Abb. 10.28) und der Diagnosealgorithmus den Fehler an dieser Änderung erkennt. Dies würde bedeuten, dass der Diagnosealgorithmus den Zeitpunkt des Fehlereintritts kennen müsste.



**Abb. 10.34:** Fehleridentifikation für das Stellgerät bei Eintritt des Fehlers  $f_1(t)$  zum Zeitpunkt  $t = 0$

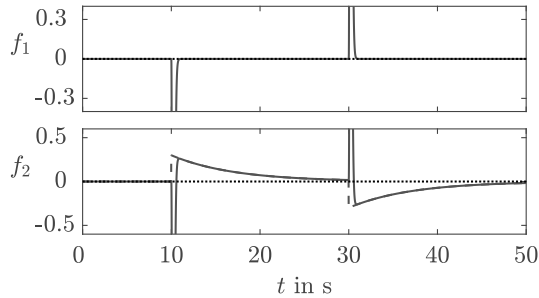
Um diese Vermutung zu widerlegen, wird der Diagnosealgorithmus jetzt zur Zeit  $t_0 = 20$  s gestartet, also zu einem Zeitpunkt, bei dem der Fehler  $f_1(t)$  schon einige Zeit lang wirkt und der Füllstandsregler die Wirkung des Fehlers auf den Füllstand  $h(t)$  bereits vollständig ausgeglichen hat (Abb. 10.28). Das Ergebnis ist in Abb. 10.35 zu sehen. Der Beobachter verarbeitet jetzt nur die in den unteren Teilen der Abbildung als durchgezogene Linien dargestellten Signalabschnitte von  $u(t)$  und  $s(t)$ . Auch hier schwingt der Beobachter schnell ein und erkennt den Fehler  $f_1(t) = 0,1$  eindeutig. Der durch den Beobachter ermittelte Wert für den Fehler  $f_2(t)$  verschwindet wie im ersten Fall nach kurzer Einschwingzeit und ist in der Abbildung nicht dargestellt. Das Beispiel zeigt, dass der Fehler  $f_1(t)$ , der eine Druckminderung oberhalb der Membran hervorruft, ohne die Messung des Drucks erkannt wird, indem die Messwerte von  $u(t)$  und  $s(t)$  mit Hilfe eines dynamischen Modells ausgewertet werden.



**Abb. 10.35:** Fehlererkennung im Stellgerät bei Fehlereintritt zum Zeitpunkt  $t = 0$  und Diagnosebeginn zur Zeit  $t_0 = 20$

Abbildung 10.36 zeigt, dass der Diagnosealgorithmus auch die Druckschwankungen im Versorgungsnetz eindeutig erkennen und von der Druckminderung durch ein Leck oberhalb der Membran unterscheiden kann. Der Beobachter verarbeitet die in Abb. 10.32 mit durchgezogenen Linien dargestellten Verläufe von  $s(t)$  und  $u(t)$ . Zu den Zeitpunkten 10 s und 30 s, an denen Druckschwankungen auftreten, schwingt der Beobachter, wobei über eine kurze Zeitspanne beide Zustandsvariablen  $\hat{f}_1(t)$  und  $\hat{f}_2(t)$  von null verschieden sind. Schon nach kurzer Zeit nähert sich der Zustand  $\hat{f}_2(t)$  dem mit der

gestrichelten Linie im unteren Teil von Abb. 10.36 dargestellten Verlauf des Fehlers  $f_2(t)$  an, während  $\hat{f}_1(t)$  verschwindet.



**Abb. 10.36:** Identifikation des Fehlers  $f_2$

**Diskussion.** Statisch betrachtet haben beide Fehler eine ähnliche Wirkung, denn für  $u(t) = 0$  und konstante Fehler  $f_1(t) = \bar{f}_1$  und  $f_2(t) = \bar{f}_2$  erhält man aus dem Modell (10.26), (10.27) des fehlerbehafteten Ventils die Beziehung

$$s(t) = k_{f1} \bar{f}_1 + k_{f2} \bar{f}_2$$

mit

$$k_{f1} = -\mathbf{c}_V^T \mathbf{A}_V^{-1} \mathbf{g}_{V1} \quad \text{und} \quad k_{f2} = -\mathbf{c}_V^T \mathbf{A}_V^{-1} \mathbf{g}_{V2},$$

die die durch die Fehler hervorgerufenen statischen Veränderungen des Hubs beschreibt. Aus einer gemessenen statischen Abweichung des Hubs kann man also nicht erkennen, ob die Abweichung durch den Fehler  $f_1(t)$  oder den Fehler  $f_2(t)$  hervorgerufen wird.

Das Beispiel zeigt, dass der Beobachter die beiden Fehler während der dynamischen Bewegung des Systems erkennen und auf Grund der unterschiedlichen Dynamik, die in den Fehlermodellen zum Ausdruck kommt, auseinander halten kann. Durch die im Beobachter enthaltenen Fehlermodelle ist nicht nur eine Fehlererkennung, sondern auch eine Fehleridentifikation möglich.  $\square$

#### **Aufgabe 10.5\*** Diagnostizierbarkeit eines Servoventils

Im Beispiel 10.4 wurde gezeigt, dass zwei unterschiedliche Fehler  $f_1(t)$  und  $f_2(t)$  mit einem beobachtergestützten Diagnosesystem auseinander gehalten werden können. Damit die Fehler identifiziert werden können, muss das Ventil eine Beobachtbarkeitsbedingung erfüllen, in die auch die Fehlermodelle für die beiden Fehler eingehen.

1. Untersuchen Sie die Diagnostizierbarkeit des Servoventils mit den im Beispiel 10.4 angegebenen Modellen für das Stellgerät und für die Fehler durch eine strukturelle Beobachtbarkeitsanalyse.
2. Bleibt die Diagnostizierbarkeit aufrechterhalten, wenn beide Fehler konstante Druckänderungen beschreiben? Diskutieren Sie Ihr Ergebnis bezüglich der dynamischen Eigenschaften eines Systems, die für die Fehleridentifikation genutzt werden.  $\square$

## 10.5 Ausblick: Diagnose und fehlertolerante Steuerung

Obwohl mit den in diesem Kapitel verwendeten linearen Modellen das Prinzip der Diagnose gut erklärt werden kann, reichen in vielen Anwendungen lineare Modelle nicht aus. Fehler führen häufig zu Parameteränderungen. Wenn man die Parameter als Zustände des Fehlermodells interpretiert, kommt man auch bei Systemen, deren Nominalverhalten sehr gut durch lineare Modelle dargestellt werden kann, im Fehlerfall zu einer nichtlinearen Beschreibung, denn die als Parameter im Prozessmodell auftretenden Zustandsgrößen des Fehlermodells werden mit den Zustandsgrößen des Prozessmodells multipliziert. Deshalb ist das hier an linearen Modellen erläuterte Grundprinzip in der Vergangenheit für unterschiedliche nichtlineare Modelle erweitert worden.

Ein anderer Weg, Fehler in einem dynamischen System zu diagnostizieren, nutzt die in der Vergangenheit in großer Breite entwickelten Verfahren der Parameteridentifikation, mit Hilfe derer aus den Verläufen der Eingangsgrößen und Ausgangsgrößen die Modellparameter bestimmt werden können. Dieser Weg ist z. B. in [41] dargestellt. Fehler werden dort als unbekannte Parameter interpretiert oder es werden die aktuellen Modellparameter bestimmt und mit vorgegebenen Nominalwerten verglichen. Die Residuen, die das Auftreten von Fehlern anzeigen, entstehen dann als Differenz der identifizierten Parameter und der vorgegebenen Nominalwerte.

Für die Anwendung der Diagnoseverfahren ist deren Unempfindlichkeit gegenüber Modellunbestimmtheiten und Störungen maßgebend. Ferner will man erreichen, dass jedes Element eines  $p$ -dimensionalen Residuums  $r(t)$  jeweils genau einen Fehler anzeigt. Man spricht dann von gerichteten Residuen.

Die Residuenauswertung, die hier stets durch einen Grenzwertest durchgeführt wurde, kann wesentlich erweitert werden, indem stochastische Testverfahren eingesetzt werden oder mit adaptiven Grenzwerten gearbeitet wird.

Schließlich spielt in der Praxis die Komplexität des Diagnoseverfahrens eine wichtige Rolle. Da die Algorithmen auf Rechnern realisiert werden, müssen sie mit zeitdiskret abgetasteten Signalen arbeiten und die zum  $k$ -ten Abtastzeitpunkt erhaltenen Messdaten  $u(k)$  und  $y(k)$  verarbeiten, bevor die  $(k + 1)$ -ten Messwerte eintreffen. Die Echtzeitfähigkeit muss auch dann gewahrt werden, wenn an Stelle einer Eingangsgröße und einer Ausgangsgröße vektorielle Signale auftreten.

**Fehlertolerante Steuerung.** Die Fehlerdiagnose ist in vielen Anwendungsgebieten erst der erste Schritt, um die Verfügbarkeit technischer Systeme zu verbessern. Im zweiten Schritt will man die Regelung dem veränderten Streckenverhalten anpassen, um auf diese Weise zu einer fehlertoleranten Steuerung<sup>4</sup> zu kommen (Abb. 10.37).

Im fehlerfreien Betrieb arbeitet die Regelung ohne Eingriffe aus der Überwachungsebene. Das Diagnosemodul überprüft dabei die Fehlerfreiheit des Prozesses. Wenn ein Fehler aufgetreten ist, wird der Fehler vom Diagnosemodul identifiziert und die Regelung durch einen Rekonfigurationsalgorithmus dem veränderten Prozessverhalten angepasst.

Wenn der erkannte Fehler nur die Dynamik der Regelstrecke modifiziert, aber die fehlerbehaftete Regelstrecke durch die bisher verwendeten Aktoren und Sensoren steuerbar bzw. be-

<sup>4</sup> Hier spricht man i. Allg. nicht von einer fehlertoleranten Regelung, sondern entsprechend dem englischen Begriff *fault-tolerant control* von einer fehlertoleranten Steuerung

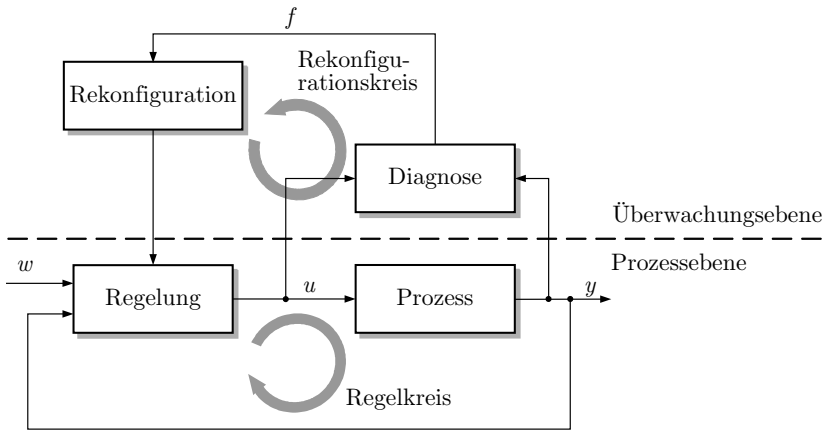


Abb. 10.37: Fehlertolerante Steuerung

obachtbar bleibt, muss der Rekonfigurationsblock lediglich die Reglerparameter anpassen, so dass der Regelkreis mit der fehlerbehafteten Regelstrecke und dem veränderten Regler die Güteforderungen erfüllt. Wenn jedoch Sensoren oder Aktoren ausfallen, ist der für die fehlerfreie Regelstrecke aufgebaute Regelkreis geöffnet und kann nur dann geschlossen werden, wenn die Regelung andere Sensoren bzw. Aktoren als bisher verwendet. Man spricht dann von einer Rekonfiguration des Regelkreises.

## Literaturhinweise

Die Arbeiten auf dem Gebiet der Prozessdiagnose begannen bereits um 1980. Eine der ersten Arbeiten ist [20]. Das Prinzip der beobachtergestützten Prozessdiagnose ist in [30] ausführlich beschrieben.

Viele der im letzten Abschnitt angesprochenen Erweiterungen der Diagnosemethodik nutzen die in diesem Kapitel behandelten Grundideen. Für Übersichten, die auch andere Methoden berücksichtigen, wird auf [19, 31, 42] verwiesen.

Die erst in den letzten Jahren untersuchten Probleme fehlertoleranter Steuerungen sowie die dafür erarbeiteten ersten Lösungsansätze werden in [13] ausführlich behandelt. Die Rekonfiguration von Regelungen wird in [100] und [86] ausführlich behandelt.

Die Parameter des im Beispiel 10.4 betrachteten Servoventils wurden in Anlehnung an die in [33] erschienenen Übersichten über Stellgeräte ausgewählt.





## Teil 3: Automatisierung ereignisdiskreter Systeme



# 11

## Beschreibung diskreter Systeme

*Automaten und Petrinetze sind die grundlegenden Modellformen ereignisdiskreter Systeme. Sie werden in diesem Kapitel eingeführt und für die Beschreibung von Beispielprozessen eingesetzt. Bei der kompositionalen Modellbildung werden die für die Teilsysteme aufgestellten Modelle zum Gesamtsystem verknüpft, was hier für deterministische Automaten erläutert wird.*

### 11.1 Modellbildungsaufgabe

#### 11.1.1 Diskrete Signale und Ereignisse

Dieses und die folgenden Kapitel beschäftigen sich mit diskreten Systemen. Das Attribut „diskret“ betrifft sowohl den Wertebereich der Signale als auch die Darstellung der Zeit. Deshalb wird zur Unterscheidung von den kontinuierlichen Systemen mit anderen Bezeichnungen gearbeitet (Abb. 11.1):

$v$  – Eingangsgröße

$z$  – Zustandsgröße

$w$  – Ausgangsgröße.

Es ist im Weiteren gleichgültig, ob die diskreten Signalwerte symbolisch dargestellt werden und beispielsweise „hoch“, „niedrig“, „blau“ oder „fallend“ heißen oder ob die Elemente der Wertebereiche durchnummeriert sind:

$$v \in \mathcal{V} = \{1, 2, \dots, M\} \quad (11.1)$$

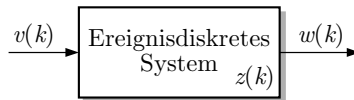
$$z \in \mathcal{Z} = \{1, 2, \dots, N\} \quad (11.2)$$

$$w \in \mathcal{W} = \{1, 2, \dots, R\}. \quad (11.3)$$

Die in den Klammern stehenden Signalwerte müssen als Symbole behandelt werden, d. h., man darf sie nicht addieren oder multiplizieren usw. In Blockschaltbildern für die Beispiele wird der Wertevorrat in geschweiften Klammern an die Signalpfeile angetragen, so dass beispielsweise an der diskreten Ausgabe eines Ventils die Menge

{geöffnet, geschlossen}

steht. Eine Durchnummierung der Signalwerte wird nur dann vorgenommen, wenn es nicht auf die Interpretation der Werte ankommt.



**Abb. 11.1:** Ereignisdiskretes System

Die Menge  $\mathcal{Z}$  der betrachteten Systemzustände heißt *Zustandsmenge* oder Zustandsraum. Das System hat  $N$  verschiedene Zustände. Die Menge der Eingabesymbole heißt  $\mathcal{V}$ , die der Ausgabesymbole  $\mathcal{W}$ .

Beispiele diskreter Systeme sind

- **Personenaufzüge:** der Zustand gibt die Etage an, in der sich der Personenaufzug befindet,
- **Werkzeugmaschinen:** der Zustand beschreibt den Bearbeitungsstand des Werkstücks,
- **Eisenbahnnetze:** der Zustand beschreibt den Bahnhof bzw. den Streckenabschnitt zwischen zwei Bahnhöfen, in dem sich der Zug befindet,
- **Kommunikationsnetze:** der Zustand beschreibt, welche Leitungen belegt bzw. frei sind.

Bei diesen Beispielen wurde nur der diskrete Zustand aufgeführt, aber auch die Eingangs- und Ausgangsgrößen dieser Systeme sind wertdiskret.

Da die im Folgenden betrachteten Signale nur zwischen diskreten Werten hin- und herspringen können, sind sie stückweise konstant. Wie bei dem in Abb. 11.2 für den Zustand  $z$  gezeigten Signalverlauf ändern sie zu den Zeitpunkten  $t_k$ , ( $k = 0, 1, \dots$ ) ihren Wert. Der über die kontinuierliche Zeitachse  $t$  definierte Signalverlauf kann also dadurch erfasst werden, dass man die Folge der Zeitpunkte  $t_k$  und die zu diesen Zeitpunkten jeweils neu angenommenen Signalwerte  $z(t_k)$  notiert. Es reicht also aus, auch auf der Zeitachse nur diskrete Punkte zu betrachten, wobei diese Punkte  $t_k$  vom Signalverlauf (und nicht etwa durch einen Takt) bestimmt werden. Die in Abb. 11.2 gezeigte Zustandstrajektorie kann in der Form

$$z(t) = (t_0, z(t_0); t_1, z(t_1); t_2, z(t_2) \dots)$$

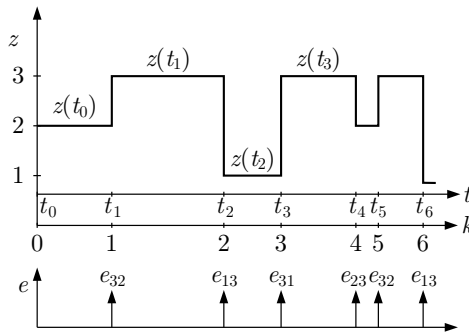


Abb. 11.2: Symbolische Signalwerte und Ereignisfolgen

dargestellt werden, wobei man für den ersten Zustandswert  $z(t_0)$  aus dem Diagramm den Wert 2, für den darauffolgenden Zustand  $z(t_1)$  den Wert 3 usw. ablesen kann. Die rechte Seite ist als abgekürzte Schreibweise für die folgende Zeitfunktion zu interpretieren:

$$z(t) = \begin{cases} z(t_0) & \text{für } t_0 \leq t < t_1 \\ z(t_1) & \text{für } t_1 \leq t < t_2 \\ z(t_2) & \text{für } t_2 \leq t < t_3 \\ \vdots & \end{cases}$$

Viele Beschreibungsformen für ereignisdiskrete Systeme ignorieren jedoch die Werte  $t_k$  und stellen lediglich dar, in welcher Reihenfolge die diskreten Signalwerte auftreten. Bei dem in Abb. 11.2 gezeigten Verlauf werden die Zustandswechsel mit einem Zähler  $k$  durchnummeriert, wie es auf der unteren der beiden Zeitachsen gezeigt ist. Der im Folgenden als Zeit  $k$  interpretierte Zähler beschreibt nicht den exakten Zeitpunkt eines Signalwertwechsels, sondern lediglich die Reihenfolge der dabei angenommenen Werte. Der Verlauf des Zustands ist dann durch die Wertefolge

$$Z = (z(0), z(1), z(2), \dots)$$

repräsentiert. Da diese Beschreibung nur die Reihenfolge der Zustände erfasst, spricht man von der *logischen Zustandsfolge* oder der nicht zeitbewerteten Zustandsfolge. Im Gegensatz dazu bezeichnet man  $z(t)$  als eine *zeitbewertete Zustandsfolge*.

Das Weglassen der Zeitangabe führt zwangsläufig auf die Frage, ob die diesbezüglichen Modelle genügend Informationen über das System enthalten. Die Antwort darauf hängt – wie bei allen Modellbildungsfragen – vom Modellbildungsziel ab. Wenn man sich nur für die Frage interessiert, welche Zustände ein System annehmen kann und in welcher Reihenfolge diese Zustände durchlaufen werden, so spielt der zeitliche Abstand der Zustandsübergänge keine Rolle und die logische Zustandsfolge ist als Verhaltensbeschreibung des betrachteten Systems ausreichend. Wenn man allerdings auch den zeitlichen Abstand der Signalwertänderungen in die Betrachtungen einbeziehen möchte oder muss, so muss man die zeitbewertete Zustandsfolge im Modell erfassen.

Im Folgenden werden diskrete Signale stets durch ihre logischen Folgen beschrieben, wobei zur Vereinfachung die Begriffe Eingangsfolge, Zustandsfolge und Ausgangsfolge ohne den

Zusatz „logisch“ gebraucht werden. An Stelle der Bezeichnung Zustandsfolge ist wie bei kontinuierlichen Systemen auch der Begriff der Zustandstrajektorie üblich.

**Ereignisse.** Den Wechsel zwischen zwei diskreten Signalwerten bezeichnet man als ein *Ereignis*, wobei der Übergang vom Signalwert  $j$  zum Signalwert  $i$  das Ereignis  $e_{ij}$  darstellt ( $e$  für Ereignis,  $i$  und  $j$  in der aus der Matrixschreibweise gebräuchlichen Reihenfolge). Wie die Signalwerte werden auch die Ereignisse durchnummeriert und der  $k$ -te Signalwechsel durch  $e(k)$  symbolisiert. Der Verlauf eines diskreten Signals kann dann durch eine Ereignisfolge

$$E = (e(0), e(1), e(2), \dots)$$

ausgedrückt werden. Diese Tatsache erklärt auch die häufig gebrauchte Bezeichnung *ereignisdiskrete Systeme* für die hier betrachtete Systemklasse.

Zur Unterscheidung der am Eingang, im Inneren bzw. am Ausgang eines Systems auftretenden Ereignisse spricht man auch von Eingangsereignissen, Zustandsereignissen bzw. Ausgangsereignissen.

Für die Darstellung der Signale ist es im Prinzip gleichgültig, ob man die Wertefolge oder die Ereignisfolge verwendet, denn jeder Wertefolge ist eindeutig eine Ereignisfolge zugeordnet und umgekehrt. So kann man den in Abb. 11.2 gezeigten Signalverlauf durch die Wertefolge

$$Z = (2, 3, 1, 3, 2, 3, 1)$$

oder durch die Ereignisfolge

$$E = (e_{32}, e_{13}, e_{31}, e_{23}, e_{32}, e_{13})$$

erfassen. Im Folgenden wird ausschließlich mit Wertefolgen gearbeitet, wobei für die Eingangs-, Zustands- und Ausgangsfolgen die Bezeichnungen

$$\begin{aligned} V(0 \dots k_e) &= (v(0), v(1), v(2), \dots, v(k_e)) \\ Z(0 \dots k_e) &= (z(0), z(1), z(2), \dots, z(k_e)) \\ W(0 \dots k_e) &= (w(0), w(1), w(2), \dots, w(k_e)) \end{aligned}$$

eingeführt werden, bei denen  $k_e$  den betrachteten Zeithorizont angibt. Da die Eingabe  $v(k_e)$  den Zustandsübergang  $z(k_e) \rightarrow z(k_e + 1)$  auslöst, hat die betrachtete Zustandsfolge häufig ein Element mehr:  $Z(0 \dots k_e + 1)$ .

Die Tatsache, dass sich das Verhalten von ereignisdiskreten Systemen in Folgen diskreter Schaltvorgänge äußert, hat schwerwiegende Konsequenzen für die mathematische Behandlung dieser Systeme. Bei den kontinuierlichen Systemen wurde die Bewegung des Systems durch den aktuellen Zustand  $\mathbf{x}(t)$  sowie durch den Differentialquotienten

$$\dot{\mathbf{x}}(t) = \frac{d\mathbf{x}(t)}{dt}$$

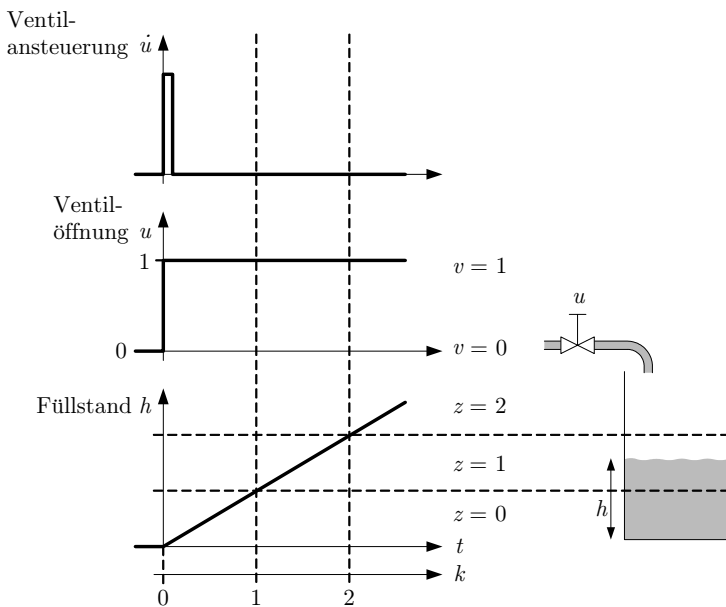
beschrieben, der die Bewegungsrichtung angibt. Bei diskreten Systemen existiert aber der Differentialquotient nicht, weshalb vollkommen andere Modellformen eingeführt werden müssen. Die auf diesen Modellen aufbauenden Lösungsmethoden für Automatisierungsaufgaben unterscheiden sich grundlegend von den für kontinuierliche Systeme entwickelten.

**Beispiel 11.1** Wertefolgen und Ereignisse beim Füllen eines Reaktors

Während in der mathematischen Darstellung Wertefolgen und Ereignisfolgen dasselbe aussagen, ist dies bei den in der Realisierung einer Steuerung verwendeten Signalen häufig nicht der Fall. Bei dem in Abb. 11.3 gezeigten Reaktor soll das Ventil durch ein Eingangsereignis geöffnet werden. Wenn dies wie in der oberen Kurve durch ein impulsförmiges Signal geschieht, das den Stellmotor des Ventils in die geöffnete Position bewegt, so entspricht dem Ereignis ein Impuls, den man sich merken muss, um die Ventilstellung zu bestimmen. Für die Zeit nach der Ventilöffnung hat das Stellsignal nämlich den Wert null und lässt die aktuelle Ventilposition nicht mehr erkennen. Demgegenüber führt das Signal „Ventilöffnung“ auf die Wertefolge

$$V(0..1) = (1, 1)$$

und zeigt für den gesamten betrachteten Zeithorizont die Ventilposition an.



**Abb. 11.3:** Wertefolge und Ereignisse bei einer Reaktorsteuerung

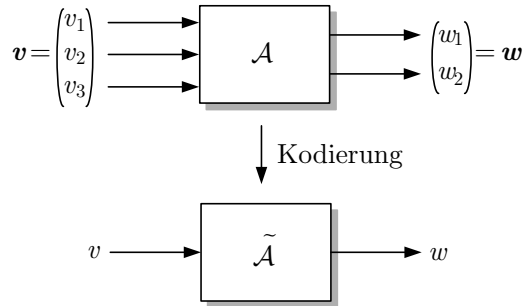
In ähnlicher Weise verhält es sich mit den durch den Reaktor erzeugten Ereignis- und Wertefolgen. Es soll nur zwischen den drei Füllhöhen unterschieden werden, die in der Abbildung durch die gestrichelten Linien gekennzeichnet sind, so dass der Reaktor in einem der diskreten Zustände  $z \in \mathcal{Z} = \{0, 1, 2\}$  sein kann. Der wahre Füllstand ist kontinuierlich und erzeugt beim Überschreiten der angegebenen Grenzwerte ein Ereignis, das die Änderung des diskreten Reaktorzustands signalisiert. Auch hier muss man sich bei impulsförmig dargestellten Ereignissen den Zustand des Reaktors merken, während die Wertefolge

$$Z(0..2) = (0, 1, 2)$$

für alle Zeitpunkte den aktuellen Zustand anzeigt.

□

**Systeme mit mehreren Eingangs- und Ausgangssignalen.** Die meisten diskreten Systeme werden über mehrere Eingangssignale gesteuert. So besitzen verfahrenstechnische Prozesse mehrere (meist sogar sehr viele) Ventile, mit denen Stoffströme beeinflusst werden, und Fertigungssysteme zahlreiche Motoren, die Werkstücke oder Werkzeuge bewegen (Abb. 11.4). Wenn die Stellglieder diskret gesteuert werden, haben sie häufig nur zwei Stellungen „geöffnet“ und „geschlossen“ bzw. „eingeschaltet“ und „ausgeschaltet“, die man einheitlich durch die Werte 0 und 1 symbolisieren kann.



**Abb. 11.4:** Ereignisdiskretes System mit mehreren Eingangs- und Ausgangsgrößen

Für die Darstellung des aktuellen Systemeingangs hat man dann zwei Möglichkeiten. Einerseits kann man die Eingangsgrößen zu einem Vektor  $v$  zusammenfassen, der unter den angegebenen Bedingungen ein binärer Vektor ist. Bei drei Stellgliedern kann dann der Systemeingang einen von acht möglichen Werten annehmen:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}. \quad (11.4)$$

In gleicher Weise bilden die Ausgangsgrößen den Vektor  $w$ . Diese Darstellungsform ist zwar aufwändig; sie stellt jedoch einen sofortigen Bezug zu den Werten der physikalischen Eingangs- und Ausgangsgrößen her.

Andererseits kann man alle möglichen Kombinationen der Eingangssymbole so kodieren, dass sie durch eine skalare Größe dargestellt werden. So können die angegebenen acht Vektoren durchnummeriert werden

$$0, 1, 2, 3, 4, 5, 6, 7,$$

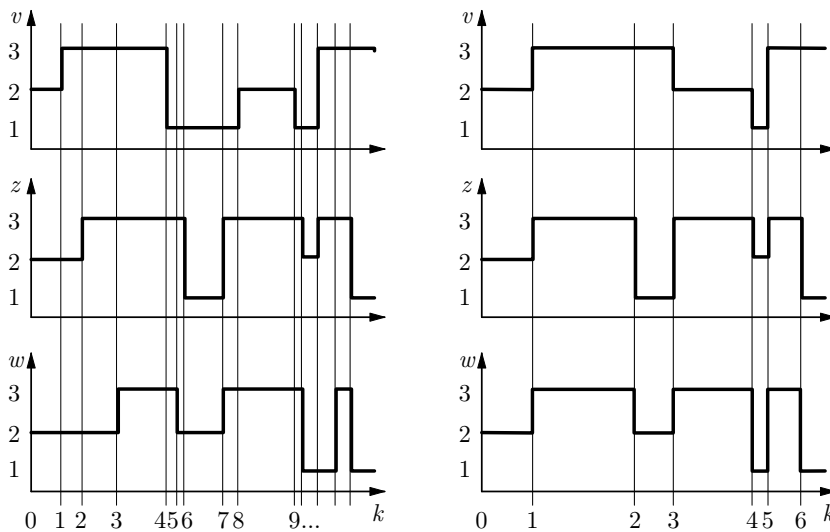
wobei man hier zweckmäßigerweise die Nummerierung bei null beginnt. Diese Darstellungsform wurde in den Gln. (11.1) – (11.3) verwendet. Sie ist einfacher als die erste; für die Interpretation der Signalwerte muss man jedoch die Kodierung rückgängig machen.

Im Folgenden werden beide Möglichkeiten verwendet. Für allgemeine Betrachtungen diskreter Systeme zeigt die zweite Darstellungsform, dass es ausreicht, Systeme mit *einer* Eingangsgröße  $v$ , *einer* Zustandsgröße  $z$  und *einer* Ausgangsgröße  $w$  zu betrachten. Die Variablen  $v$ ,  $z$  und  $w$  stehen dann jedoch i. Allg. für Kombinationen mehrerer diskreter physika-



lischer Signalwerte. Darauf sollen auch die im Folgenden verwendeten Bezeichnungen hinweisen. Wenn man mit den skalaren Größen  $v$ ,  $z$  und  $w$  arbeitet, spricht man nicht mehr von Eingangsgröße, Zustandsgröße und Ausgangsgröße, sondern abstrakter vom Eingang, Zustand bzw. Ausgang. Für einen einzelnen Wert des Eingangs wird die Bezeichnung „Eingabe“ und für den des Ausgangs „Ausgabe“ gebraucht. Die Wertemengen  $\mathcal{V}$ ,  $\mathcal{Z}$  und  $\mathcal{W}$  heißen Eingabealphabet, Zustandsmenge oder Zustandsalphabet bzw. Ausgabealphabet.

**Synchrone und asynchrone Ereignisse.** Ereignisse am Eingang, Zustand und Ausgang können zu beliebigen Zeiten auftreten (Abb. 11.5 links). Bei den im Folgenden behandelten Modellbildungsaufgaben wird vorausgesetzt, dass sich die drei Größen zum selben Zeitpunkt ändern. Dies ist einerseits eine Idealisierung, auf die man bei vielen Systemen durch geringfügige Verschiebung der Ereigniszeitpunkte kommt. So kann man an Stelle der in Abb. 11.5 links dargestellten Signalverläufe die rechts gezeigten Verläufe verwenden, bei denen die Ereignisse synchron auftreten. Andererseits werden bei gesteuerten Systemen die Eingangsereignisse durch die Steuereinrichtung als Reaktion auf Ausgangsereignisse vorgegeben. Wenn die Rechenzeit der Steuereinrichtung wesentlich kleiner ist als die zeitlichen Abstände der Zustands- und Ausgangsereignisse, kann man sie vernachlässigen und Eingangs- und Ausgangsereignisse als synchron auffassen. Die Abbildung zeigt, dass sich dann nicht zu jedem Zeitpunkt  $k$  alle Signalwerte ändern müssen.



**Abb. 11.5:** Asynchrone (links) und synchrone (rechts) Eingangs-, Zustands- und Ausgangsfolgen

**Leeres Symbol, beliebiger Signalwert.** Bei der Betrachtung mehrerer Signale wird der nächste interessante Zeitpunkt durch die Veränderung eines Signals festgelegt. Zu diesem Zeitpunkt muss sich nicht zwangsläufig auch der Wert der anderen Signale ändern. Man drückt diese Tatsache dadurch aus, dass man den vorherigen Signalwert in der Wertefolge wiederholt. Eine andere Möglichkeit besteht in der Einführung des leeren Symbol  $\varepsilon$ , das im Folgenden in mehreren Bedeutungen gebraucht wird und hier verwendet wird um anzuzeigen, dass im System ein Ereignis aufgetreten ist, ohne dass sich der Wert des betreffenden Signals geändert hat.

Die im rechten Teil von Abb. 11.5 gezeigte Eingangsfolge  $V$  heißt dann entweder

$$V(0...6) = (2, 3, 3, 2, 1, 3, 3)$$

oder

$$V(0...6) = (2, 3, \varepsilon, 2, 1, 3, \varepsilon).$$

Mit  $\varepsilon$  wird also nur der Zähler  $k$  für die Zeit um eins erhöht. Im Folgenden ist das leere Symbol  $\varepsilon$  stets ein Element des Wertevorrates aller Signale, ohne dass dies in jedem Fall angegeben wird.

Bei der Beschreibung ereignisdiskreter Systeme haben nicht stets alle Eingangsgrößen einen Einfluss auf das Systemverhalten. Beispielsweise ist es für den Füllstand eines leeren Reaktors belanglos, ob das Ausflussventil geöffnet oder geschlossen oder der Rührer angeschaltet oder abgeschaltet ist. Der Füllstand wird sich nur erhöhen, wenn das Zuflussventil geöffnet wird. Um in diesem Fall nicht alle möglichen Werte für die bedeutungslosen Signale aufzählen zu müssen, wird das Symbol  $*$  eingeführt, das mit jedem Wert des Wertevorrates des betreffenden Signals belegt werden kann. Man bezeichnet diesen Wert zutreffend auch als *don't care*. Der Vektor

$$v = \begin{pmatrix} \text{offen} \\ * \end{pmatrix}$$

bedeutet deshalb, dass das Eingangssignal  $v_1$  den Wert „offen“ hat, während das Eingangssignal  $v_2$  einen beliebigen Wert seines Wertebereichs besitzt.

### 11.1.2 Modellbildungsziel und Modellbildungsschritte

Das Modell eines diskreten Systems soll beschreiben, wie sich der Zustand und der Ausgang des Systems in Abhängigkeit vom Anfangszustand und von der Eingangsfolge verändern. Man geht bei der Modellbildung davon aus, dass die Vorgabe einer neuen Eingabe die Zustandsänderung und die Erzeugung der Ausgabe auslöst. Auf den Zeitpunkt  $t$ , zu dem dies geschieht, kommt es bei den logischen Signalfolgen ohnehin nicht an. Wenn der aktuelle Zustand angenommen ist, kann „irgendwann“ die nächste Eingabe folgen. Das Auftreten der Eingabe ist also das Zeichen dafür, dass das Modell den Zustandswechsel vollführen soll, wobei der Zustandswechsel selbst keine Zeit in Anspruch nimmt. Diese Betrachtungsweise wird durch das für diskrete Systeme eingeführte Attribut „ereignisgetrieben“ treffend charakterisiert.

Das Modell verwendet die diskrete Zeitachse, auf der durch den Zähler  $k$  die Zustandsübergänge durchnummeriert sind. Es soll beschreiben, in welchen Nachfolgezustand  $z(k+1)$  das System übergeht, wenn es zur Zeit  $k$  im Zustand  $z(k)$  ist und die Eingabe  $v(k)$  erhält. Außerdem

soll das Modell beschreiben, welche Ausgabe  $w(k)$  in dieser Situation durch das System erzeugt wird. Das Modell muss also, im Prinzip, die Form

$$z(k+1) = G(z(k), v(k)), \quad z(0) = z_0 \quad (11.5)$$

$$w(k) = H(z(k), v(k)) \quad (11.6)$$

haben. Die folgenden Abschnitte werden zeigen, welche Schwierigkeiten bei der Darstellung der Funktionen  $G$  und  $H$  auftreten und wie man diese Gleichungen erweitern muss. Bis auf diese Einzelheiten eingegangen wird, soll unter dem Modell eines diskreten Systems dieses Gleichungssystem verstanden werden, so dass die Modellbildungsaufgabe folgendermaßen formuliert werden kann:

**Modellbildungsaufgabe für diskrete Systeme**

Gegeben:   Dynamisches System  
                   mit Eingang  $v$  und Ausgang  $w$   
 Gesucht:   Modell (11.5), (11.6)

Diese Aufgabenstellung ist – bis auf Nuancen in den Begriffen – dieselbe wie für kontinuierliche Systeme. Für die Lösung dieser Aufgabe gibt es jedoch zwei gravierende Unterschiede im Vergleich zu der im Kap. 3 beschriebenen Lösung. Der erste Unterschied bezieht sich auf die Art und die Einheitlichkeit der verwendeten Modellform. Modelle für kontinuierliche Systeme haben, bis auf Ausnahmen, stets die Form einer Differentialgleichung bzw. eines Zustandsraummodells (oder äquivalenter Beschreibungsformen). Demgegenüber gibt es für diskrete Systeme keine einheitliche Modellform, sondern eine Vielzahl unterschiedlicher Beschreibungsmittel, von denen hier zwei eingeführt werden: Automaten und Petrinetze.

Um diesen Unterschied zu verdeutlichen, sei daran erinnert, dass bei kontinuierlichen Systemen die physikalischen Gesetze, als algebraische Gleichungen oder Differentialgleichungen geschrieben, in sehr kompakter Form die Zustandstrajektorie darstellen. Beispielsweise erhält man für einen Gleichstrommotor die Differentialgleichung (3.14)

$$\ddot{\omega}(t) + \left( \frac{R}{L} + \frac{k_R}{J} \right) \dot{\omega}(t) + \left( \frac{Rk_R}{JL} + \frac{k_M k_T}{JL} \right) \omega(t) = \frac{k_T}{JL} u(t),$$

die den Zusammenhang zwischen der Eingangsspannung  $u(t)$  und der Motordrehzahl  $\omega(t)$  repräsentiert, und zwar für beliebige Werte der in dieser Gleichung vorkommenden Signale  $\ddot{\omega}(t)$ ,  $\dot{\omega}(t)$ ,  $\omega(t)$  und  $u(t)$ .

Für einige diskrete Systeme ist eine derart kompakte Beschreibung der Zustandsübergänge auch möglich. Betrachtet man einen Zähler mit dem diskreten Zustand  $z \in \{0, 1, 2, 3, \dots\}$  und dem Eingang  $v \in \{0, 1\}$ , bei dem sich der Wert des Zustands bei der Eingabe 1 um eins erhöht, so erhält man die Beziehung

$$z(k+1) = z(k) + v(k), \quad z(0) = 0, \quad (11.7)$$

in der die Funktion  $G$  die einfache Form

$$G(z(k), v(k)) = z(k) + v(k)$$

hat. Für die meisten diskreten Systeme findet man jedoch keine derartige analytische Beschreibung, sondern man muss alle Zustandswechsel einzeln betrachten und aufschreiben. Die Ursache dafür liegt in der Tatsache, dass die Werte diskreter Signale i. Allg. symbolisch sind und es deshalb keine algebraischen Operationen gibt, mit denen man diese Signalwerte verknüpfen kann. Der Zähler (11.7) ist eine Ausnahme, denn dessen Zustand hat einen ganzzahligen Wert, zu dem der Wert des ganzzahligen Eingangs addiert werden kann.

Der zweite Unterschied zwischen der Modellierung kontinuierlicher und ereignisdiskreter Systeme bezieht sich auf den Lösungsweg der gestellten Modellbildungsaufgabe. Bei kontinuierlichen Systemen entstehen viele Modelle aus den physikalischen Gesetzen, die das Verhalten der Systemkomponenten beschreiben (vgl. Schritt 2 im Algorithmus 3.1 auf S. 86). Dieser Modellierungsweg versagt bei diskreten Systemen, weil die Mehrzahl der mit einem diskreten Modell zu beschreibenden Prozesse in Wirklichkeit kontinuierliche Zustandsübergänge haben und die diskrete Betrachtungsweise eine Abstraktion von diesen kontinuierlichen Vorgängen beinhaltet. So sind alle Bewegungen der im Beispiel 11.2 betrachteten Stanze kontinuierlich in Ort und Zeit, aber das diskrete Modell soll nur den Übergang zwischen den diskreten Zuständen wie „Stanze ist geöffnet“ und „Stanze ist geschlossen“ wiedergeben. Würde man sich für den genauen Bewegungsablauf interessieren, so könnte man die Bewegungsgleichung der Stanze in Form einer Differentialgleichung aufschreiben. Beim diskreten Modell will man aber gerade diese Bewegung ignorieren und nur den Ausgangspunkt (geöffnete Stanze) und das Ergebnis der Bewegung (geschlossene Stanze) in Beziehung zueinander setzen. Dieser Zusammenhang resultiert nicht nur aus den physikalischen Bewegungsgesetzen, sondern auch aus der Interpretation des Anfangs- und Endpunktes der Bewegung als Zustand der Stanze. Deshalb führen die physikalischen Gesetze nicht direkt zu einer analytischen Darstellung der diskreten Zustandsübergänge.

Der Zähler (11.7) ist auch in dieser Hinsicht eine Ausnahme, für die sich der angegebene analytische Ausdruck für die Funktion  $G$  aus der Funktionsweise des betrachteten Systems ergibt. Hier ist die mathematische Definition des Zählers der Ausgangspunkt und die Funktion  $G$  gibt diese Definition wieder.

Die Schwierigkeit, die Funktionen  $G$  und  $H$  im Modell (11.5), (11.6) aus einer Analyse des zu beschreibenden Prozesses als analytische Ausdrücke aufzuschreiben, ist die Ursache für die Vielzahl von Beschreibungsmitteln, die in der Automatisierungstechnik für diskrete Systeme entwickelt wurden und eingesetzt werden. Diese Beschreibungsmittel sollen es dem Fachmann erleichtern, sein Wissen über das System so aufzuschreiben, dass es direkt für die Lösung von Simulations-, Analyse- oder Steuerungsaufgabe eingesetzt werden kann.

**Automaten und Petrinetze.** Alle Beschreibungsmittel für diskrete Systeme beruhen auf derselben Grundidee: es müssen diskrete Zustandsübergänge in Abhängigkeit von diskreten Eingaben dargestellt werden. Die folgenden Abschnitte dieses Kapitels befassen sich deshalb nicht mit allen diesen Beschreibungsmitteln, sondern mit den ihnen zu Grunde liegenden Ideen der diskreten Modellierung. Mit dem Automaten wird die Grundform der Modelle diskreter Systeme eingeführt. Automaten haben als Beschreibungselemente nur die in den Gleichungen (11.5) und (11.6) vorkommenden Größen: die Signale für den Eingang, den Zustand und den Ausgang sowie die Funktionen, die die Zustandsübergänge und die Erzeugung der Ausgabe beschreiben.

Sie sind also die sparsamste Modellform, die gerade die minimal notwendigen Modellelemente besitzt. Deshalb zeigen Automaten in besonders klarer Form, welche Phänomene bei diskreten Systemen auftreten können. Die meisten automatisierungstechnischen Fragestellungen werden in den nachfolgenden Kapiteln deshalb für Automaten gelöst.

Als zweite Modellform werden Petrinetze behandelt, die mehr Modellelemente als Automaten haben. Das Mehr an Modellelementen ermöglicht eine kompaktere Systemdarstellung, was in der Modellbildungsphase sehr wichtig ist. Die Überführung von Petrinetzen in Automaten und umgekehrt zeigt einerseits, dass Petrinetze häufig besser überschaubar sind als Automaten und dass sie andererseits (in der hier verwendeten Grundform) keine zusätzlichen Phänomene diskreter Systeme erfassen können. Die Petrinetze werden hier also als ein Schritt in Richtung auf komfortablere Beschreibungsmittel in die Betrachtungen einbezogen.

**Modellbildungsweg.** Diese Erläuterungen zeigen, dass man bei der Modellbildung diskreter Systeme einen anderen Weg als bei kontinuierlichen Systemen gehen muss. Da man für die einzelnen Systemkomponenten keine Gleichungen aufschreiben kann, muss man für das betrachtete System die Gesamtheit der Eingaben, Zustände und Ausgaben im Blick behalten:

---

**Algorithmus 11.1** *Modellbildung diskreter Systeme*

---

**Gegeben:** Diskretes System mit Eingang  $v$  und Ausgang  $w$ .

1. Definieren Sie die diskreten Wertemengen des Eingangs, des Zustands und des Ausgangs.
2. Ermitteln Sie alle diskreten Zustandsübergänge in Abhängigkeit von der aktuellen Eingabe.
3. Ordnen Sie die Ausgaben den Zustandsübergängen zu.

**Ergebnis:** Modell der Form (11.5), (11.6).

---

In den Schritten 2 und 3 werden Automaten und Petrinetze für die Beschreibung der Funktionen  $G$  und  $H$  eingesetzt. Das hier durch die Beziehungen (11.5), (11.6) symbolisierte Modell hat deshalb die Form einer Automatentabelle oder einer Netzmatrix.

Modelle diskreter Systeme können sich auf die Steuerstrecke, auf die Steuereinrichtung oder auf den Regelkreis als Ganzes beziehen. Die in diesem Kapitel behandelten Beispiele zeigen alle diese Möglichkeiten.

## 11.2 Deterministische Automaten

### 11.2.1 Autonome deterministische Automaten

Automaten sind die grundlegende Beschreibungsform dynamischer Systeme mit wertdiskreten Signalen. Es wird zunächst der autonome deterministische Automat eingeführt, der keinen Eingang hat und bei dem angenommen wird, dass die Ausgabe gleich dem Zustand ist.

Autonome deterministische Automaten werden durch das Tripel

$$\text{Autonomer deterministischer Automat: } \mathcal{A} = (\mathcal{Z}, G, z_0)$$

beschrieben, bei dem  $\mathcal{Z}$  die Zustandsmenge,  $G$  die *Zustandsübergangsfunktion* (oder Zustandsüberföhrungsfunktion)

$$G : \mathcal{Z} \rightarrow \mathcal{Z}$$

und  $z_0 \in \mathcal{Z}$  den Anfangszustand beschreibt. Bei allen hier betrachteten Automaten hat die Zustandsmenge endlich viele Elemente. Man spricht deshalb auch von *endlichen Automaten* oder *endlichen Zustandsmaschinen*, wobei allerdings das Attribut „endlich“ im Folgenden weggelassen wird.

Die Zustandsübergangsfunktion  $G$  gibt an, in welchen Zustand  $z(k+1)$  der Automat übergeht, wenn er im Zustand  $z(k)$  ist:

Zustandsraumdarstellung autonomer deterministischer Automaten:

$$\mathcal{A} : z(k+1) = G(z(k)), \quad z(0) = z_0. \quad (11.8)$$

Das Attribut „deterministisch“ besagt, dass für jeden Zustand  $z(k) \in \mathcal{Z}$  der Nachfolgezustand  $z(k+1)$  *eindeutig* bestimmt ist.

Wenn man Eigenschaften der Zustandsübergangsfunktion  $G$  untersuchen will, spielt es keine Rolle, zu welchem Zeitpunkt  $k$  ein Zustandsübergang auftritt. Man schreibt dann Gl. (11.8) kürzer als

$$z' = G(z). \quad (11.9)$$

Wendet man Gl. (11.8) für  $k = 0, 1, 2, \dots$  an, so erhält man die Zustandsfolge

$$Z = (z(0), z(1), z(2), \dots),$$

die im vorgegebenen Anfangszustand  $z(0) = z_0$  beginnt. Weil die Zustandsübergangsfunktion  $G$  jedem Zustand  $z(k)$  genau einen Nachfolgezustand  $z(k+1)$  zuordnet, ist die Trajektorie  $Z$  eindeutig.

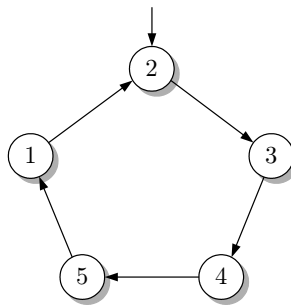
Die Zustandsübergänge von Automaten sind abrupt. Es wird nicht beschrieben, wie diese Übergänge im Einzelnen ablaufen. Der Automat schaltet von einem zum nachfolgenden Zustand, wobei die Schaltzeit als vernachlässigbar klein angesehen wird. Die Zustandsübergangsfunktion wird deshalb gelegentlich auch als *Schaltregel* bezeichnet. Der abrupte Zustandsübergang ist nicht nur für den hier angegebenen Automaten, sondern auch für alle weiteren ereignisdiskreten Beschreibungsformen charakteristisch.

**Automatentabelle.** Da für die meisten diskreten Systeme die Funktion  $G$  nicht analytisch darstellbar ist, muss sie beispielsweise durch die Menge aller Paare  $(z', z)$  von Zuständen  $z$  und zugehörigen Nachfolgezuständen  $z'$  notiert werden. Diese Paare werden üblicherweise in einer Tabelle angeordnet, die man Automatentabelle nennt:

$$G = \begin{array}{|c|c|} \hline z' & z \\ \hline 2 & 1 \\ 3 & 2 \\ 4 & 3 \\ 5 & 4 \\ 1 & 5 \\ \hline \end{array}$$

Man hätte über die erste Tabellenspalte auch  $z(k+1)$  und über die zweite Spalte  $z(k)$  schreiben können.

**Automatengraph.** Eine andere Darstellungsmöglichkeit für die Zustandsübergangsfunktion, die sehr anschaulich ist, bietet der Automatengraph  $\mathcal{G}(\mathcal{Z}, \mathcal{E})$ , dessen Knoten die Zustände  $z \in \mathcal{Z}$  und dessen gerichtete Kanten  $e \in \mathcal{E}$  die durch  $G$  beschriebenen Zustandsübergänge repräsentieren. Die Kanten repräsentieren Ereignisse. Da die Funktion  $G$  jedem Zustand  $z$  eindeutig einen Nachfolgezustand  $z'$  zuordnet, geht von jedem Knoten genau eine Kante  $(z, z')$  aus, die von  $z$  nach  $z'$  gerichtet ist. Für den durch die o. a. Automatentabelle beschriebenen Automaten ist der Graph in Abb. 11.6 dargestellt.



**Abb. 11.6:** Automatengraph eines autonomen deterministischen Automaten

Wenn der Anfangszustand des Automaten bekannt ist, wird er im Automatengraphen durch einen Pfeil gekennzeichnet, wie dies in Abb. 11.6 am Zustand 2 zu sehen ist.

**Matrixdarstellung deterministischer Automaten.** Zur Matrixdarstellung deterministischer Automaten kommt man nach der Einführung eines  $N$ -dimensionalen Vektors  $\mathbf{p}(k)$ , dessen  $i$ -te Komponente angibt, ob sich der Automat zum Zeitpunkt  $k$  im Zustand  $i$  befindet ( $p_i(k) = 1$ ) oder nicht ( $p_i(k) = 0$ ). Dabei ist  $N$  die Anzahl der Automatenzustände (vgl. Gl. (11.2)). Die

Zustandsübergangsfunktion  $G$  wird als  $(N, \times N)$ -Matrix  $\mathbf{G}$  geschrieben, deren Element  $g_{ij}$  genau dann gleich eins ist, wenn  $i = G(j)$  gilt, also der Automat vom Zustand  $j$  in den Zustand  $i$  übergeht. Alle anderen Elemente sind gleich null. Die Matrix  $\mathbf{G}$  besitzt in jeder Spalte genau eine Eins.

Mit diesen Größen kann das Verhalten des Automaten in der Form

$$\mathcal{A}: \mathbf{p}(k+1) = \mathbf{G} \mathbf{p}(k), \quad \mathbf{p}(0) = \mathbf{p}_0 \quad (11.10)$$

geschrieben werden. Der Anfangszustand  $\mathbf{p}_0$  enthält bis auf die zu  $z_0$  gehörende Stelle nur Nullen und bei  $z_0$  eine Eins. Die Zustandstrajektorie  $Z$  wird dann durch die Folge der Vektoren  $\mathbf{p}(k)$  beschrieben, die man durch wiederholte Multiplikation des Vektors  $\mathbf{p}(0)$  mit der Matrix  $\mathbf{G}$  erhält:

$$Z = (\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots).$$

**Automatendarstellung durch boolesche Funktionen.** In der Steuerungstechnik wird auch eine boolesche Darstellung verwendet, die hier als letzte Darstellungsmöglichkeit der Funktion  $G$  erwähnt werden soll. Grundlage dessen ist eine boolesche Beschreibung der diskreten Zustände, die man beispielsweise für ein System mit acht Zuständen aus der binären Kodierung (11.4) dadurch erhalten kann, dass man die drei Elemente des binären Vektors als boolesche Variablen  $z_1$ ,  $z_2$  und  $z_3$  interpretiert. Die Automatentabelle kann man dann mit diesen booleschen Größen aufstellen, was für das oben angegebene Beispiel auf

$z'_3$	$z'_2$	$z'_1$	$z_3$	$z_2$	$z_1$
0	1	0	0	0	1
0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	1	0	0
0	0	1	1	0	1

führt. Jetzt kann man die Größen  $z'_i$  als boolesche Ausdrücke mit den booleschen Variablen  $z_1$ ,  $z_2$  und  $z_3$  aufschreiben

$$\begin{aligned} z'_1 &= (\bar{z}_1 \wedge z_2 \wedge \bar{z}_3) \vee (\bar{z}_1 \wedge \bar{z}_2 \wedge z_3) \vee (z_1 \wedge \bar{z}_2 \wedge z_3) \\ z'_2 &= (z_1 \wedge \bar{z}_2 \wedge \bar{z}_3) \vee (\bar{z}_1 \wedge z_2 \wedge \bar{z}_3) \\ z'_3 &= (z_1 \wedge z_2 \wedge \bar{z}_3) \vee (\bar{z}_1 \wedge \bar{z}_2 \wedge z_3), \end{aligned}$$

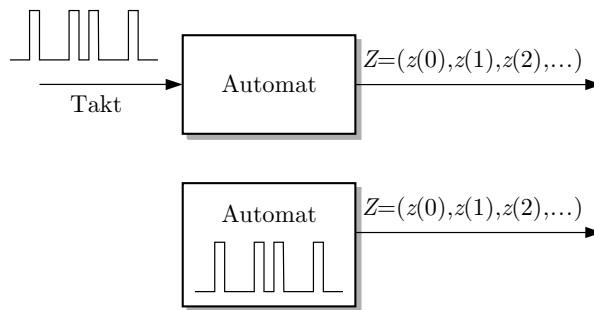
wobei der Querstrich die Komplementbildung bedeutet. Jeder Klammerausdruck auf der rechten Seite beschreibt eine Wertekombination für  $z_1$ ,  $z_2$  und  $z_3$ , für die der Größe  $z'_i$  in der Automatentabelle der Wert eins zugeordnet ist.

Diese Darstellungsform wird nicht weiter verwendet und zeigt hier lediglich, dass diskrete Systeme auch durch boolesche Ausdrücke beschrieben werden können. Die angegebenen booleschen Gleichungen sind eine analytische Darstellungsform der Zustandsübergangsfunktion  $G$ , die mit Hilfe der booleschen Algebra umgeformt, vereinfacht und analysiert werden kann. Da man für jede Automatentabelle eine solche Darstellung finden kann, ist jede Zustandsübergangsfunktion  $G$  durch analytische Ausdrücke darstellbar. Wenn davon gesprochen wird, dass



es für  $G$  keine kompakte Darstellungsform gibt, so ist damit gemeint, dass man diese Darstellung nicht direkt bei einer Analyse des Systemverhaltens hinschreiben kann. Die booleschen Ausdrücke wurden für das Beispiel durch Umformung und Kodierung der Automatentabelle erzeugt.

**Takt.** Da der Automat keine innere Uhr besitzt, die den Zeitpunkt einer Zustandsänderung vorgibt, braucht man eine Vereinbarung darüber, wann der Automat seinen nächsten Zustand annimmt. Bei den später behandelten Automaten mit Eingang wird vereinbart, dass der Zeitpunkt, an dem das nächste Eingangssymbol am Eingang erscheint, der Zeitpunkt der Zustandsänderung ist. Bei den bisher betrachteten autonomen Automaten gibt es aber ein solches Eingangssymbol nicht. Man muss deshalb vereinbaren, wann der Automat das Kommando „schalte“ erhält.



**Abb. 11.7:** Getakteter Automat

In Abhängigkeit von der Anwendung gibt es dafür zwei Möglichkeiten (Abb. 11.7). Eine Möglichkeit ist es, von außen einen Takt vorzugeben, der den Zeitpunkt einer Zustandsänderung anzeigt. Dieser Takt wird häufig durch das leere Eingangssymbol  $\varepsilon$  beschrieben, so dass der Automat auf die Eingangsfolge

$$V(0...k_e) = (\varepsilon, \varepsilon, \varepsilon, \dots, \varepsilon)$$

mit der Zustandsfolge

$$Z(0...k_e) = (z(0), z(1), z(2), \dots, z(k_e))$$

antwortet. Die leeren Eingangssymbole dienen also nur zur Kennzeichnung der Ursache des nächsten Zustandswechsels.

Wenn man nur an den möglichen Zustandsfolgen interessiert ist, verwendet man die zweite Möglichkeit, den Umschaltzeitpunkt des Automaten zu definieren. Man stellt sich vor, dass der Taktgeber Bestandteil des Automaten ist, der immer neue Zustandsübergänge auslöst.

**Beispiel 11.2** Ereignisdiskrete Beschreibung einer Stanze

In einer Stanze werden Blechteile in mehreren aufeinander folgenden Arbeitsgängen gestanzt, wobei beim ersten Arbeitsschritt das Blech von einer Blechrolle abgeschnitten wird und die nachfolgenden Arbeitsschritte der Umformung des Werkstücks dienen. Zum Schluss wird das Werkstück in einer Ablage (Lager, Kiste) gestapelt.

Da es sich bei der Stanze (einschließlich deren Steuerung) um ein autonomes System handelt, das keine Eingaben aus der Umgebung erhält, und weil bei diesem System keine sich vom Zustand unterscheidenden Ausgaben eingeführt werden, müssen nur die Schritte 1 und 2 des Modellbildungsalgorithmus 11.1 durchlaufen werden:

1. **Definition der Zustände:** Für die Beschreibung des Stanzvorgangs wird zwischen folgenden Zuständen unterschieden:

Zustand	Beschreibung
1	Die Stanze steht in der Ausgangsposition.
2	Das Blech befindet sich unter der Stanze.
3	Das Blech ist abgetrennt.
4	Die Stanze ist nach dem ersten Arbeitsgang geöffnet.
5	Das Werkstück ist neu ausgerichtet.
6	Das Werkstück ist umgeformt.
7	Die Stanze ist nach der Umformung des Werkstücks geöffnet.

2. **Beschreibung der Zustandsübergänge:** Die Funktion  $G$  muss folgende Zustandsänderungen erfassen:

$z'$	$z$	Beschreibung
2	1	Einführen des Blechs
3	2	Erster Stanzvorgang (Abtrennen des Blechs)
4	3	Öffnen der Stanze
5	4	Drehen des Werkstücks
6	5	Zweiter Stanzvorgang (Umformen des Blechs)
7	6	Öffnen der Stanze
1	7	Transport des Werkstücks aus der Stanze in das Lager

Der Anfangszustand ist  $z_0 = 1$ . Der Automat kann durch den in Abbildung 11.8 gezeigten Automatengraphen veranschaulicht werden.

3. **Beschreibung der Ausgaben:** Wenn man den aktuellen Zustand der Stanze aus den Messsignalen ermitteln kann, gilt die Beziehung  $w(k) = z(k)$ . Man kann aber auch einzelne Messgrößen für die Bestimmung der Ausgaben heranziehen, beispielsweise die Position der Stanze („auf“, „zu“).

Die Matrixdarstellung dieses Automaten heißt

$$\begin{pmatrix} p_1(k+1) \\ p_2(k+1) \\ p_3(k+1) \\ p_4(k+1) \\ p_5(k+1) \\ p_6(k+1) \\ p_7(k+1) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} p_1(k) \\ p_2(k) \\ p_3(k) \\ p_4(k) \\ p_5(k) \\ p_6(k) \\ p_7(k) \end{pmatrix}, \quad \begin{pmatrix} p_1(0) \\ p_2(0) \\ p_3(0) \\ p_4(0) \\ p_5(0) \\ p_6(0) \\ p_7(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

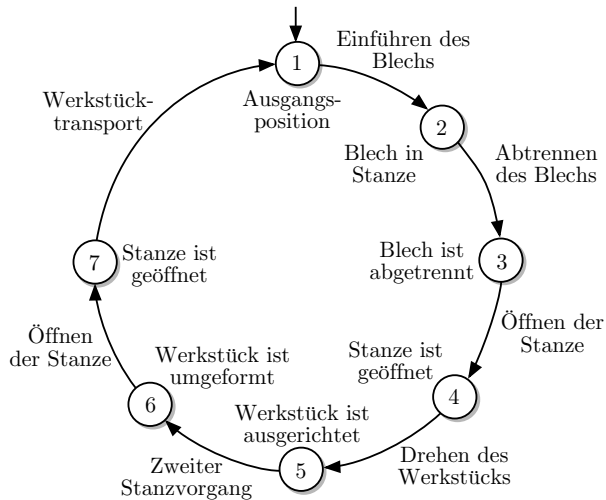


Abb. 11.8: Automatengraph der Stanze

Daraus erhält man die Zustandsfolge

$$Z = \left( \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \dots \right).$$

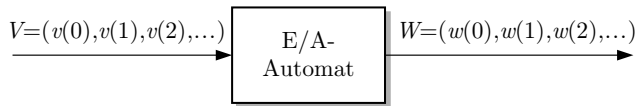
Das Beispiel zeigt, dass der Automat die diskreten Zustände der Stanze sowie die Reihenfolge beschreibt, in der die Stanze diese diskreten Zustände annimmt. Er beschreibt *nicht*, wann die Zustandsübergänge stattfinden und wie sie im Einzelnen ablaufen.

**Diskussion.** Wie in diesem Beispiel wird bei der Modellierung von Werkzeugmaschinen einem Automatenzustand typischerweise ein Zustand der Fertigungseinrichtung und einem Zustandsübergang ein Fertigungsschritt zugeordnet. Das Schalten des Automaten entspricht der Ausführung eines einzelnen Arbeitsschrittes der Stanze.

Die Stanze bildet zusammen mit ihrer Automatisierungseinrichtung ein autonomes System. Da die Steuerung nach jedem Arbeitsschritt eindeutig vorschreibt, was die Stanze als nächstes tun soll, ist das Verhalten deterministisch.  $\square$

### 11.2.2 Deterministische Automaten mit Eingang und Ausgang

In der Automatisierungstechnik spielen Systeme eine große Rolle, deren Verhalten durch einen Eingang beeinflusst und durch Messung eines Ausganges überwacht werden kann, wobei der Ausgang häufig nicht identisch mit dem Zustand ist. Für derartige Systeme wird jetzt der autonome Automat erweitert.



**Abb. 11.9:** Automat mit Eingang und Ausgang

Der deterministische Automat mit Eingang und Ausgang (deterministischer E/A-Automat) ist durch das Tupel

$$\text{Deterministischer E/A-Automat: } \mathcal{A} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, G, H, z_0) \quad (11.11)$$

beschrieben, das angibt, dass der Eingang  $v$  die Wertemenge  $\mathcal{V}$  und der Ausgang  $w$  die Wertemenge  $\mathcal{W}$  hat. Das dynamische Verhalten wird durch die Zustandsübergangsfunktion

$$G : \mathcal{Z} \times \mathcal{V} \rightarrow \mathcal{Z}$$

und die Ausgabefunktion

$$H : \mathcal{Z} \times \mathcal{V} \rightarrow \mathcal{W}$$

dargestellt, wobei die Beziehungen

Zustandsraumdarstellung deterministischer E/A-Automaten:

$$\mathcal{A} : \begin{cases} z(k+1) = G(z(k), v(k)), & z(0) = z_0 \\ w(k) = H(z(k), v(k)) \end{cases} \quad (11.12)$$

gelten. Das Attribut „deterministisch“ besagt, dass jedem Paar  $(z(k), v(k))$  genau ein Nachfolgezustand  $z(k+1)$  und eine Ausgabe  $w(k)$  zugeordnet werden.

Die Gleichungen (11.12) ähneln dem im Kap. 3 eingeführten Zustandsraummodell kontinuierlicher Systeme, nur dass jetzt die Zustandsgleichung keine Differentialgleichung ist, sondern eine Beziehung, die aus dem aktuellen Zustand und der Eingabe den Nachfolgezustand bestimmt. Die für den deterministischen Automaten angegebenen Gleichungen können deshalb gleichfalls als Zustandsraummodell bezeichnet werden, obwohl dieser Begriff bei ereignisdiskreten Systemen nicht so gebräuchlich ist wie bei kontinuierlichen Systemen. Auch ist die Definition des Automaten als Tupel (11.11) analog zur Definition des Zustandsraummodells eines nichtlinearen Systems, die man entsprechend Gl. (3.28) auf S. 103 in der Form

$$\Sigma_{\text{NL}} = (\mathcal{X}, \mathcal{U}, \mathcal{Y}, g, h, x_0)$$

schreiben kann. In beiden Definitionen treten dieselben Größen auf:

$\mathcal{Z}$	$\mathcal{X}$	Zustandsmenge, Zustandsraum
$\mathcal{V}$	$\mathcal{U}$	Eingangsalphabet, Eingangsraum
$\mathcal{W}$	$\mathcal{Y}$	Ausgangsalphabet, Ausgangsraum
$G$	$g$	Zustandsübergangsfunktion
$H$	$h$	Ausgabefunktion
$z_0$	$x_0$	Anfangszustand.

Unterschiedlich sind allerdings die Darstellungsformen, denn wie beim autonomen Automaten kann man auch beim E/A-Automaten typischerweise keine analytischen Ausdrücke für die Funktionen  $G$  und  $H$  angeben, während dies für die entsprechenden Funktionen  $g$  und  $h$  kontinuierlicher Systeme möglich ist.

**Zustand deterministischer Automaten.** Die Analogie zwischen diskreten und kontinuierlichen Systemen lässt sich auch auf den Zustandsbegriff ausdehnen. Ähnlich zur Definition 3.1 auf S. 94 des Zustands kontinuierlicher Systeme wird der Zustand ereignisdiskreter Systeme folgendermaßen definiert:

**Definition 11.1 (Zustand eines ereignisdiskreten Systems)**

*Das Signal  $z(k)$  heißt Zustand eines ereignisdiskreten Systems, wenn für einen beliebigen Zeithorizont  $k_e \geq 0$  sein Wert  $z(0) = z_0$  zum Zeitpunkt  $k = 0$  zusammen mit der Eingabefolge  $V(0 \dots k_e)$  den Wert  $w(k_e)$  der Ausgabe eindeutig bestimmt.*

Damit hat bei der Modellbildung die Festlegung einer geeigneten Zustandsmenge für ereignisdiskrete Systeme dieselbe Bedeutung wie für kontinuierliche Systeme. Hier ist die Zustandsmenge allerdings kein linearer Vektorraum, sondern eine endliche Menge von Symbolen. Die Kunst der Modellbildung besteht in der geeigneten Definition des Signals  $z(k)$ , so dass die Signalwerte  $z \in \mathcal{Z}$  alle Informationen enthalten, die für die *eindeutige* (deterministische) Bewegung des Systems notwendig sind.

**Reaktion auf eine leere Eingabe.** Vereinbarungsgemäß gehört das leere Symbol  $\varepsilon$  zur Menge  $\mathcal{V}$  der Eingabewerte und zur Menge  $\mathcal{W}$  der Ausgabewerte. Dieses Symbol kann verwendet werden um auszudrücken, dass der Automat „nichts tut“. Bei den meisten Systemen wird angenommen, dass sie für die Eingabe  $v = \varepsilon$  keinen Zustandswechsel ausführen und kein Ausgabesymbol erzeugen, was auf die Modellgleichungen

$$z = G(z, \varepsilon) \quad (11.13)$$

$$\varepsilon = H(z, \varepsilon) \quad (11.14)$$

führt.

Alternativ kann man das leere Symbol einsetzen, um spontane Zustandsübergänge darzustellen, die nicht durch ein Eingabesymbol, sondern durch interne Vorgänge des betrachteten Prozesses ausgelöst wurden. Dann führt das System ohne Eingabe den Zustandswechsel von  $z$  zu  $z'$  aus

$$z' = G(z, \varepsilon),$$

was im Automatengraphen durch die Kante  $z \xrightarrow{\varepsilon/w} z'$  dargestellt ist. Derartige spontane Ereignisse treten beispielsweise bei Uhren auf, die nach Ablauf einer eingestellten Zeit – und folglich ohne Eingabe – in einen neuen Zustand übergehen. Außerdem kann man die leere Eingabe verwenden, um die spontane Erzeugung einer Ausgabe darzustellen:

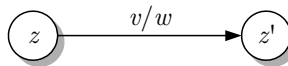
$$w = H(z, \varepsilon).$$

**Automatentabelle und Automatengraph.** Für Automaten mit Eingang und Ausgang muss die Automatentabelle erweitert werden, wobei man die Argumente  $z$  und  $v$ , die den aktuellen Zustand und die aktuelle Eingabe beschreiben, rechts und den Nachfolgezustand  $z'$  und die Ausgabe  $w$  links anordnet:

$z'$	$w$	$z$	$v$
2	1	1	1
2	1	5	2
3	1	2	1
3	2	1	2
4	2	3	1
4	1	2	2
5	2	4	1
5	2	3	2
1	2	5	1
1	1	4	2

Diese Tabelle enthält die Funktionen  $G$  und  $H$ .

Der zu diesem Beispiel gehörige Automatengraph ist in Abb. 11.11 zu sehen. Die Kanten des Graphen sind mit der Eingabe und der Ausgabe gekennzeichnet, wie es in Abb. 11.10 gezeigt ist. Um die Bewegung des Systems für die Eingabe  $v = 1$  von der für die Eingabe  $v = 2$  zu unterscheiden, ist bei diesem Beispiel die zweite Gruppe von Zustandsübergängen durch gestrichelte Pfeile gekennzeichnet.



**Abb. 11.10:** Kante im Automatengraphen zur Kennzeichnung des Zustandsübergangs von  $z$  nach  $z'$  unter der Wirkung der Eingabe  $v$ , wobei die Ausgabe  $w$  erzeugt wird

**Mealy-Automat und Moore-Automat.** In Abhängigkeit davon, ob die Ausgabefunktion  $H$  von der Eingabe abhängt oder nicht, führt man für den Automaten unterschiedliche Bezeichnungen ein:<sup>1</sup>

- **Mealy-Automat:** Automat, bei dem die Ausgabefunktion  $H$  von  $z$  und  $v$  abhängt:

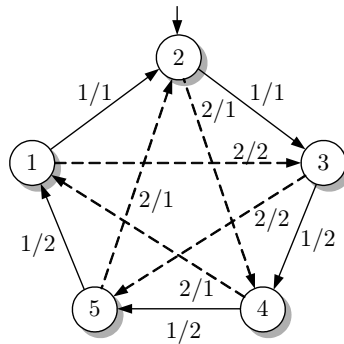
$$w(k) = H(z(k), v(k)).$$

Bei diesen Automaten tritt ein Durchgriff vom Eingang  $v$  auf den Ausgang  $w$  auf, denn  $w(k)$  hängt direkt von der zur gleichen Zeit vorkommenden Eingabe  $v(k)$  ab.

- **Moore-Automat:** Automat, dessen Ausgabefunktion  $H$  nicht von der Eingangsgröße  $v$  abhängig ist, d. h., für den gilt

$$w(k) = H(z(k)). \quad (11.15)$$

<sup>1</sup> benannt nach EDWARD F. MOORE (1925 – 2003) und GEORGE H. MEALY (1927 – 2010).



**Abb. 11.11:** Automatengraph eines deterministischen Automaten mit Eingang und Ausgang

Der Moore-Automat entspricht nicht sprungfähigen kontinuierlichen Systemen, in deren Ausgangsgleichung (3.17) auf S. 91 der Parameter  $d$  verschwindet und  $y(t)$  folglich nur vom Zustand  $x(t)$  abhängt bzw. in deren Funktion  $h$  in Gl. (3.26) auf S. 102 die Größe  $u(t)$  nicht vorkommt. Beim Moore-Automaten ist die Ausgabe dem Zustand zugeordnet. Derartige Automaten repräsentieren beispielsweise Systeme, deren Ausgabe den Sensorwerten im aktuellen Zustand entsprechen.

Die Unterscheidung beider Automatentypen kann man anhand der Automatentabelle oder des Automatengraphen vornehmen. In der Automatentabelle eines Moore-Automaten tritt in allen Zeilen, die zu einem bestimmten Zustand  $z$  gehören, stets derselbe Wert  $w$  für die Ausgabe auf. Dies ist in dem angegebenen Beispiel nicht der Fall, denn in den zu  $z = 1$  gehörenden Zeilen tritt einmal  $w = 1$  und einmal  $w = 2$  auf:

$z'$	$w$	$z$	$v$
2	1	1	1
...	...	...	...
3	2	1	2
...	...	...	...

Das Beispiel ist also ein Mealy-Automat. Im Automatengraphen eines Moore-Automaten muss an allen von einem Zustand ausgehenden Kanten dieselbe Ausgabe stehen. Man kann die Ausgabe  $w$  deshalb dem Zustandsknoten  $z$  zuordnen. Im Beispiel haben die vom Knoten 1 ausgehenden Kanten unterschiedliche Ausgaben, so dass sich auch auf diesem Wege herausstellt, dass es sich um einen Mealy-Automaten handelt.

**Matrixdarstellung deterministischer E/A-Automaten.** Für die Matrixdarstellung muss zur Beschreibung der Ausgabe ein  $R$ -dimensionaler Vektor  $p_w(k)$  eingeführt werden, dessen  $i$ -tes Element genau dann gleich eins ist, wenn der Ausgang zur Zeit  $k$  den Wert  $i$  annimmt. An Stelle von Gl. (11.10) schreibt man jetzt das folgende Gleichungssystem:

$$p(k+1) = G(v(k)) p(k), \quad k = 0, 1, 2, \dots \quad (11.16)$$

$$p_w(k) = H(v(k)) p(k). \quad (11.17)$$

Die Matrizen  $\mathbf{G}$  und  $\mathbf{H}$  sind von der Eingabe  $v(k)$  abhängig. Sie besitzen in jeder Spalte genau eine Eins, sonst Nullen. Diese Eigenschaft gilt für alle Eingaben  $v(k)$ , wobei sich aber die Position der Einsen in Abhängigkeit von  $v(k)$  verschieben kann. Die zur aktuellen Eingabe  $v(k)$  gehörenden Matrizen werden mit dem aktuellen Vektor  $\mathbf{p}(k)$  multipliziert, um den neuen Zustand  $\mathbf{p}(k+1)$  bzw. die aktuelle Ausgabe  $\mathbf{p}_w(k)$  zu erhalten.

**Steuerbare und nicht steuerbare Ereignisse.** Ereignisse sind bei deterministischen Automaten durch den Übergang vom Zustand  $z$  in einen Nachfolgezustand  $z'$  beschrieben, im Automatengraphen also durch die Bewegung entlang einer Kante  $z \rightarrow z'$ . Wenn im Automatengraphen von einem Knoten  $z$  mehrere Kanten ausgehen, kann der Automat vom Zustand  $z$  eines von mehreren möglichen Ereignissen erzeugen. Für die Lösung einer Steuerungsaufgabe ist es wichtig, dass man durch die Eingabe  $v$  Einfluss darauf nehmen kann, welches Ereignis als nächstes erzeugt wird. Man nennt ein Ereignis  $e$  *steuerbar* im Zustand  $z$ , wenn es möglich ist, durch eine geeignet gewählte Eingabe zu erreichen, dass der sich im Zustand  $z$  befindende Automat als nächstes das Ereignis  $e$  erzeugt und folglich den Zustandsübergang  $z \xrightarrow{e} z'$  ausführt.

Die Prüfung, welche Ereignisse steuerbar sind, ist sehr einfach, weil die Bewegung eines deterministischen Automaten eindeutig durch den aktuellen Zustand  $z(k)$  und die Eingabe  $v(k)$  bestimmt ist. Wenn jede Kante des Automatengraphen durch ein Eingabesymbol bewertet ist, sind alle Ereignisse steuerbar. In verschiedenen Anwendungsfällen kann es jedoch vorkommen, dass bestimmte Ereignisse nicht durch die Eingabe bestimmt werden, sondern spontan auftreten. Man kann dann weder durch die Eingabe den Zeitpunkt bestimmen, an dem das Ereignis auftritt, noch kann man durch die Eingabe verhindern, dass das betreffende Ereignis erzeugt wird. Ein solches Ereignis ist nicht steuerbar. Es ist im Automatengraphen durch eine Kante  $z \xrightarrow{\varepsilon/w} z'$  ohne Eingangssymbol repräsentiert.

**Takt.** Der Zeitpunkt, an dem der Automat seinen Zustand ändert und die Ausgabe erzeugt, wird durch die Eingabe bestimmt. Es wird also entsprechend Abb. 11.9 davon ausgegangen, dass das Auftreten des nächsten Eingangssymbols  $v(k)$  den Automaten zum Schalten aktiviert. Heißt das Eingangssymbol  $v(k) = \varepsilon$ , so schaltet der Automat nicht. Dementsprechend wird die leere Eingabe verwendet, wenn man mehrere Automaten gemeinsam betrachtet und nicht alle Automaten gleichzeitig schalten.

#### **Aufgabe 11.1\*** Modell eines Getränkeautomaten

Ein Getränkeautomat kann Kaffee und Kakao herstellen. Nachdem der Kunde eine ausreichende Summe Geld eingeworfen hat, blinken die Wahlkosten für Kaffee und Kakao. Nach der Wahl wird das gewünschte Getränk zubereitet und ausgeteilt. Über eine Taste kann der Kunde den Vorgang abbrechen, bevor das Getränk zubereitet ist. Beschreiben Sie das Verhalten des Getränkeautomaten durch einen deterministischen Automaten, wobei Sie den Automatengraphen und die Automatentabelle angeben.  $\square$

#### **Aufgabe 11.2\*** Modell eines Zählers

Beschreiben Sie das Verhalten eines Zählers durch einen deterministischen Automaten. Geben Sie einen analytischen Ausdruck für die Zustandsübergangsfunktion  $G$  an.  $\square$



**Aufgabe 11.3\*** Ereignisdiskrete Beschreibung des Autofahrens

Beim Autofahren werden Aktionen wie Beschleunigen, Bremsen, Lenken usw. in Abhängigkeit von der qualitativ erfassten Verkehrssituation ausgewählt. Stellen Sie einen Automaten auf, der dieses Verhalten in Abhängigkeit von der Verkehrssituation darstellt.  $\square$

**Aufgabe 11.4\*** Beschreibung eines Bestellvorgangs

Beschreiben Sie die in einem Buchladen ablaufenden Vorgänge von der Bestellung eines Buches bis zum Verkauf des angelieferten Buches durch einen Automaten.  $\square$

**Aufgabe 11.5** Automatentheoretische Beschreibung der Fußball-Weltmeisterschaft

Der Fortgang der Fußball-WM wird in der Endrunde im K.o.-System durch die diskreten Spielergebnisse beschrieben, die für jede noch beteiligte Mannschaft nur die „Eingangsgrößen“  $g$  (gewonnen) und  $v$  (verloren) annehmen können. Der Zustand wird dann durch die sich noch im Spiel befindenden Mannschaften beschrieben. Wie groß ist die Zustandsmenge, wenn zu Beginn der K.o.-Spiele 16 Mannschaften im Turnier sind? Welche Zustandsübergänge gibt es? Zeichnen Sie einen Ausschnitt des Automatengraphen, der den Fortgang der Fußball-WM in der K.o.-Runde repräsentiert.  $\square$

## 11.3 Nichtdeterministische Automaten

### 11.3.1 Deterministische und nichtdeterministische Systeme

Von der Zustandsraumdarstellung wertkontinuierlicher Systeme  $\Sigma$  ist bekannt, dass der Anfangszustand  $x_0$  all diejenigen Informationen enthält, die für eine *eindeutige* Vorhersage des zukünftigen Systemverhaltens notwendig sind. So kann man aus dem Gleichungssystem

$$\Sigma : \begin{cases} \dot{x}(t) = Ax(t), & x(0) = x_0 \\ y(t) = c^T x(t) \end{cases}$$

bei Kenntnis von  $x_0$  die Trajektorie

$$y(t) = c^T e^{At} x_0$$

für das gesamte Zeitintervall  $0 \leq t < \infty$  eindeutig bestimmen. Dieselbe Eigenschaft besitzen deterministische Automaten  $\mathcal{A}$ , denn mit Gl. (11.8)

$$\mathcal{A} : z(k+1) = G(z(k)), \quad z(0) = z_0$$

kann ebenfalls für einen Anfangszustand  $z_0$  die Zustandsfolge  $Z(0 \dots k_e)$  für einen beliebig großen Zeithorizont  $k_e$  eindeutig vorhergesagt werden. Bei beiden Systemklassen bleibt die

eindeutige Vorhersagbarkeit des Verhaltens auch dann erhalten, wenn das System durch eine bekannte Eingangsgröße  $u(t)$  bzw.  $v(k)$  beeinflusst wird.

Im Gegensatz dazu reicht bei nichtdeterministischen Systemen die im aktuellen Zustand enthaltene Information nicht aus, um das zukünftige Verhalten eindeutig vorherzusagen. Streng genommen ist damit der Zustand kein „Zustand“ mehr im Sinne der Zustandsdefinition (Definitionen 3.1 und 11.1)! Man kann für diese Systeme möglicherweise eine Menge  $\mathcal{X}(x_0)$  bestimmen, die alle möglichen Zustandstrajektorien enthält:  $x(t) \in \mathcal{X}(x_0)$ .

|| Nichtdeterministische Systeme sind Systeme, bei denen die vorhandenen Kenntnisse über die dynamischen Eigenschaften und über den Anfangszustand nicht ausreichen, um das Systemverhalten eindeutig vorherzusagen.

Diese Charakterisierung nichtdeterministischer Systeme ist unabhängig davon, ob es mit mehr Informationen über das System möglich wäre, das Verhalten eindeutig vorherzubestimmen.

Beispiele für nichtdeterministische Systeme sind

- ein **Würfel**: der Ausgang des nächsten Wurfes ist nicht vorhersagbar,
- die **Brownsche Bewegung**: die Position der in einer Flüssigkeit oder in einem Gas schwebenden Teilchen während ihrer regellosen, zitternden Bewegung ist nicht vorhersagbar,
- **technische Systeme** bezüglich ihres Ausfallverhaltens: der Zeitpunkt des Wechsels zwischen fehlerfreiem und fehlerbehaftetem Zustand ist nicht exakt vorhersagbar,
- **Fertigungsprozesse** mit mehreren parallel ablaufenden Fertigungsschritten: es ist nicht eindeutig vorhersagbar, welcher Fertigungsschritt zuerst beendet wird.

In der angegebenen Reihenfolge nimmt die Genauigkeit, mit der das Systemverhalten beschrieben werden kann, zu. Bei allen Beispielen bleibt jedoch eine Unsicherheit in Bezug auf die Reihenfolge bzw. die Richtung der nächsten Zustandswechsel.

Der Nichtdeterminismus spielt bei diskreten Systemen eine größere Rolle als bei kontinuierlichen Systemen. Dafür gibt es zwei Gründe. Erstens ignorieren die hier behandelten Modelle die zeitlichen Abstände der Ereignisse, so dass bei parallel ablaufenden Prozessen ein nichtdeterministisches Verhalten aus der unterschiedlichen Reihenfolge resultiert, in der die Prozesse beendet werden und einen neuen Systemzustand erzeugen.

|| Die logische Beschreibung paralleler Prozesse führt i. Allg. auf ein nichtdeterministisches Modell, weil ohne Zeitangaben nicht entschieden werden kann, in welcher Reihenfolge die Prozesse beendet werden.

Zweitens werden Störungen, die bei kontinuierlichen Systemen durch ein zusätzliches Eingangssignal  $d(t)$  repräsentiert werden, bei diskreten Systemen nur dann erfasst, wenn sie sich in einer veränderten Zustandstrajektorie bemerkbar machen. Beispielsweise tritt wegen einer Störung nicht der Zustandswechsel  $z \rightarrow z'$ , sondern ein anderer Zustandswechsel  $z \rightarrow \tilde{z}'$  ein. Wenn das Modell alle diese möglichen Zustandswechsel erfassen soll, muss es nichtdeterministisch sein, weil einem Zustand  $z$  dann für dieselbe Eingabe mehr als ein Nachfolgezustand zugeordnet werden muss.

Beim Übergang vom deterministischen zum nichtdeterministischen System muss die Modellform erweitert werden. Offenbar können  $G$  und  $H$  in den Gln. (11.5) und (11.6) keine *Funktionen* mehr sein, denn jede Funktion weist ihren Argumenten  $z$  und  $v$  genau *einen* Wert zu. Im Modell eines nichtdeterministischen Systems sind  $G$  und  $H$  *Relationen*, die vorgegebene Größen  $z$  und  $v$  mit mehr als einem Zustand  $z'$  bzw. mehr als einer Ausgabe  $w$  verknüpfen.

### 11.3.2 Autonome nichtdeterministische Automaten

Ein nichtdeterministischer Automat

Autonomer nichtdeterministischer Automat:  $\mathcal{N} = (\mathcal{Z}, G, \mathcal{Z}_0)$

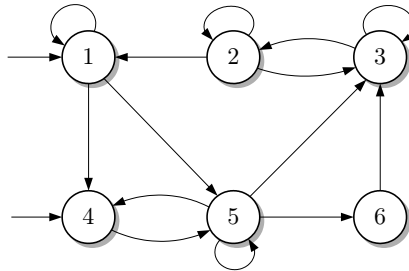
unterscheidet sich vom deterministischen Automaten dadurch, dass seine Zustandsübergänge nicht eindeutig festgelegt sind. Die Zustandsübergangsfunktion  $G$

$$G : \mathcal{Z} \rightarrow 2^{\mathcal{Z}} \quad (11.18)$$

ordnet jedem Zustand  $z$  eine Menge  $G(z) \subseteq \mathcal{Z}$  von möglichen Nachfolgezuständen  $z'$  zu:

$$z' \in G(z). \quad (11.19)$$

$2^{\mathcal{Z}}$  bezeichnet die Potenzmenge von  $\mathcal{Z}$  (das ist die Menge aller Untermengen von  $\mathcal{Z}$ ). Als weiterer Unterschied zu deterministischen Automaten wird bei nichtdeterministischen Automaten zugelassen, dass bereits der Anfangszustand nicht genau bekannt ist.  $\mathcal{Z}_0$  bezeichnet eine Menge von Anfangszuständen, für die gilt  $z(0) \in \mathcal{Z}_0$ .



**Abb. 11.12:** Automatengraph eines nichtdeterministischen Automaten

Gleichwertig zur Funktion  $G$  kann man die Dynamik autonomer nichtdeterministischer Automaten durch eine Relation

$$\mathcal{G} \subset \mathcal{Z} \times \mathcal{Z}$$

beschreiben, also durch eine Menge geordneter Paare  $(z', z)$ . Dabei wird mit dem Symbol  $\mathcal{G}$  an Stelle von  $G$  gearbeitet, weil  $\mathcal{G}$  eine Menge ist. Die Darstellungen der Dynamik nichtdeterministischer Automaten durch die Funktion  $G$  und die Relation  $\mathcal{G}$  sind äquivalent in dem Sinne, dass

$$z' \in G(z) \iff (z', z) \in \mathcal{G}$$

gilt. Man spricht deshalb bei beiden Darstellungsformen von einer *Zustandsübergangsrelation*.

Aufgrund der nichtdeterministischen Zustandsübergänge gibt es in der Automatentabelle Zeilen, die sich nicht in  $z$ , aber in  $z'$  unterscheiden, wie das folgende Beispiel zeigt:

$$G =$$

$z'$	$z$
1	1
4	1
5	1
1	2
2	2
3	2
2	3
$\vdots$	$\vdots$

Bei diesem Beispiel haben die Zustände 1 und 2 drei Nachfolgezustände. Im Graphen nichtdeterministischer Automaten gibt es folglich Knoten, von denen mehrere Kanten ausgehen, wie in Abb. 11.12 zu sehen ist. Das Beispiel zeigt auch, dass bei nichtdeterministischen Automaten mehrere Zustände als Anfangszustände markiert sein können:  $\mathcal{Z}_0 = \{1, 4\}$ . Mit den zuvor eingeführten Schreibweisen ist der Automat durch

$$\begin{aligned} G(1) &= \{1, 4, 5\} \\ G(2) &= \{1, 2, 3\} \\ G(3) &= \{2, 3\} \\ G(4) &= \{5\} \\ G(5) &= \{3, 4, 5, 6\} \\ G(6) &= \{3\} \end{aligned}$$

und

$$\mathcal{G} = \{(1, 1), (4, 1), (5, 1), (1, 2), (2, 2), (3, 2), (2, 3), (3, 3), \dots\}$$

beschrieben.

**Verhalten nichtdeterministischer Automaten.** Der nichtdeterministische Automat mit dem Anfangszustand  $z(0) \in \mathcal{Z}_0$  kann jede Zustandsfolge

$$Z(0 \dots k_e) = (z(0), z(1), z(2), \dots, z(k_e))$$

durchlaufen, für die die Beziehung

Zustandsraumdarstellung autonomer nichtdeterministischer Automaten:

$$\mathcal{N} : z(k+1) \in G(z(k)), \quad z(0) \in \mathcal{Z}_0$$

(11.20)

gilt. Da diese Beziehung offenlässt, welchen Zustand aus der Menge  $G(z(k))$  der Automat zum Zeitpunkt  $k + 1$  annimmt, kann man die Zustandsfolge  $Z$  nicht eindeutig vorhersagen, sondern nur eine Menge  $\mathcal{B}$  von möglichen Zustandsfolgen angeben:

$$\mathcal{B} = \{(z(0), z(1), z(2), \dots) \mid z(0) \in \mathcal{Z}_0, z(k+1) \in G(z(k)) \text{ für } k = 0, 1, 2, \dots\}. \quad (11.21)$$

Für den Beispielautomaten umfasst die Menge  $\mathcal{B}$  Zustandsfolgen, die von den Zuständen  $z_0 = 1$  oder  $z_0 = 4$  ausgehen und zu denen für den Zeithorizont  $k_e = 4$  u. a. die hier angegebenen Folgen gehören:

$$\begin{aligned} \mathcal{B} = \{ & (1, 1, 1, 4, 5), (1, 5, 6, 3, 3), (1, 4, 5, 5, 3), \dots \\ & (4, 5, 6, 3, 2), (4, 5, 5, 6, 3), (4, 5, 3, 3, 3), \dots \}. \end{aligned} \quad (11.22)$$

Wenn man sich nur für die Mengen  $\mathcal{Z}(k)$  von Zuständen interessiert, die der Automat zum Zeitpunkt  $k$  annehmen kann, so beschreibt man das Verhalten des Automaten durch die Folge

$$\mathcal{Z}(0 \dots k_e) = (\mathcal{Z}(0), \mathcal{Z}(1), \mathcal{Z}(2), \dots, \mathcal{Z}(k_e)), \quad (11.23)$$

deren Elemente  $\mathcal{Z}(k)$  man rekursiv ermitteln kann:

$$\mathcal{Z}(0) = \mathcal{Z}_0 \quad (11.24)$$

$$\mathcal{Z}(k+1) = \{z' \in G(z) \mid z \in \mathcal{Z}(k)\}, \quad k = 0, 1, 2, \dots \quad (11.25)$$

Für das Beispiel erhält man auf diese Weise

$$\mathcal{Z}(0 \dots 4) = \{\{1, 4\}, \{1, 4, 5\}, \{1, 3, 4, 5, 6\}, \{1, 2, 3, 4, 5, 6\}, \{1, 2, 3, 4, 5, 6\}\}.$$

Diese Darstellung ist kompakter als die Menge (11.22) und sie zeigt, dass sich der Beispielautomat nach drei Zustandsübergängen in jedem seiner Zustände befinden kann. Sie lässt jedoch keine Aussage über einzelne Zustandsfolgen mehr zu. So könnte man aus der Menge  $\mathcal{Z}(0 \dots 4)$  folgern, dass der Automat die Zustandsfolge

$$Z(0 \dots 4) = (1, 4, 1, 2, 1)$$

durchlaufen kann, was allerdings durch die Zustandsübergangsrelation ausgeschlossen ist. Bei der Bildung der Folge  $\mathcal{Z}(0 \dots k_e)$  ist die Information darüber verloren gegangen, dass zum Zustand  $z \in \mathcal{Z}(k)$  nur die Teilmenge  $G(z) \subseteq \mathcal{Z}(k+1)$  von Nachfolgezuständen gehört.

**Matrixdarstellung nichtdeterministischer Automaten.** Wie beim deterministischen Automaten kann man den Zustand zum Zeitpunkt  $k$  durch den Vektor  $p(k)$  beschreiben. Beim nichtdeterministischen Automaten treten jedoch i. Allg. mehrere Einsen auf. Die Matrix  $G$  hat in der  $j$ -ten Spalte mehr als eine Eins, wenn der Zustandsübergang vom Zustand  $j$  nichtdeterministisch ist. Mit diesen Erweiterungen gilt analog zu Gl. (11.10)

$$\mathcal{N}: p(k+1) = G \circ p(k), \quad k = 0, 1, 2, \dots, \quad (11.26)$$

wobei das Zeichen  $\circ$  eine Matrix-Vektor-Multiplikation darstellt, deren Ergebnis auf eins abgerundet wird, wenn es größer als eins ist. Man kann diese Operation auch als logische UND-Verknüpfung interpretieren. Die zum Zeitpunkt  $k$  möglichen Systemzustände werden durch die Folge der Vektoren  $p(k)$  beschrieben, die durch Gl. (11.26) erzeugt wird:

$$\mathbf{Z}(0\dots k_e) = (\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots, \mathbf{p}(k_e)).$$

Diese Folge sagt dasselbe aus wie die in Gl. (11.23) angegebene Folge  $\mathbf{Z}(0\dots k_e)$ , denn die Menge  $\mathbf{Z}(k)$  umfasst genau diejenigen Zustände, für die im Vektor  $\mathbf{p}(k)$  Einsen stehen. Für den Beispielautomaten erhält man die Folge

$$\mathbf{Z}(0\dots 4) = \left( \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right).$$

**Zustand nichtdeterministischer Automaten.** Der Zustand wurde im Kap. 3 für kontinuierliche und im Abschn. 11.2.2 für deterministische ereignisdiskrete Systeme als eine Darstellung derjenigen Informationen eingeführt, die hinreichend dafür sind, dass man das zukünftige Systemverhalten eindeutig vorhersagen kann. Bei nichtdeterministischen Automaten stimmt die Verwendung des Zustands  $z$  nicht mehr mit dieser Begriffsbestimmung überein. Man muss den Begriff dahingehend erweitern, dass der Zustand  $z$  jetzt nicht mehr den Nachfolgezustand, sondern die *Menge* der Nachfolgezustände *eindeutig* bestimmt. Dies wird aus Gl. (11.25) deutlich. Aufgrund dieser Eigenschaft des Zustands nichtdeterministischer Automaten ist die Folge (11.23) von Zustandsmengen eindeutig.

Für die Modellbildung heißt das, dass man bei nichtdeterministischen Systemen zwar keine Möglichkeit hat, im Zustand alle Informationen für eine eindeutige Vorhersage des Nachfolgezustands zusammenzufassen. Man muss die Zustandsmenge aber so definieren, dass man für jeden Zustand die Menge der Nachfolgezustände eindeutig festlegen kann. Dabei ist man im Sinne einer guten Aussagekraft des Modells daran interessiert, die Zustandsmenge so festzulegen, dass der „kleinstmögliche“ Nichtdeterminismus auftritt.

**Interpretation der Zustandsfolge des Automaten.** Das nichtdeterministische Verhalten dynamischer Systeme bereitet oft Schwierigkeiten in Bezug auf die Interpretation der Folge  $\mathbf{Z}$  bzw. der Menge  $\mathcal{B}$ . Das technische System kann natürlich nur genau eine Trajektorie durchlaufen, während das Modell mehrere vorgibt. Mit Hilfe des Modells weiß man, dass die vom System durchlaufene Trajektorie zur Menge  $\mathcal{B}$  gehört, aber man weiß nicht, welches Element von  $\mathcal{B}$  das tatsächliche Systemverhalten beschreibt.

### Beispiel 11.3 Beschreibung einer Stanze unter Berücksichtigung von Fehlermöglichkeiten

Das im Beispiel 11.2 beschriebene Modell einer Stanze soll jetzt so erweitert werden, dass es auch das Verhalten in zwei Fehlerfällen wiedergibt. Erstens bleibt die Stanze stehen, wenn die Blechrolle aufgebraucht ist. Zweitens stoppt die Steuerung den Stanzvorgang, wenn das Werkstück nicht ordnungsgemäß in die Ablage eingeordnet ist. Um diese Fehlerfälle zu berücksichtigen, müssen zwei neue Zustände eingeführt und der bereits verwendete Zustand 1 genauer definiert werden:

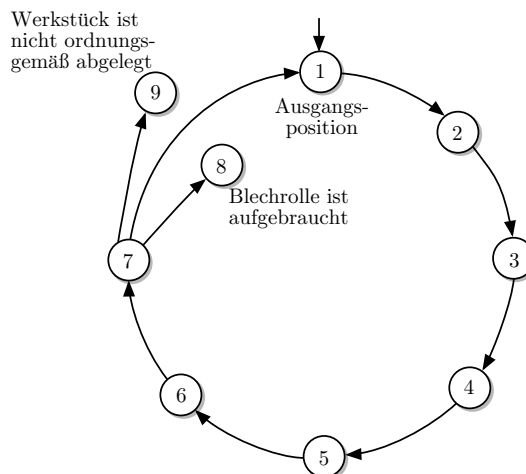
Zustand	Beschreibung
1	Die Stanze steht in der Ausgangsposition und es ist noch Blech vorhanden.
8	Die Stanze steht in der Ausgangsposition, aber die Blechrolle ist aufgebraucht.
9	Das Werkstück ist nicht ordnungsgemäß abgelegt.

Aus der im Beispiel 11.2 angegebenen Zustandsübergangsfunktion wird eine Zustandsübergangsrelation, die folgende nichtdeterministische Übergänge enthält:

$z'$	$z$	Beschreibung
1	7	Die Stanze kehrt ordnungsgemäß in die Ausgangsposition zurück.
8	7	Das Werkstück wird ordnungsgemäß transportiert, aber die Blechrolle ist leer.
9	7	Das Werkstück wird nicht ordnungsgemäß aus der Stanze transportiert.

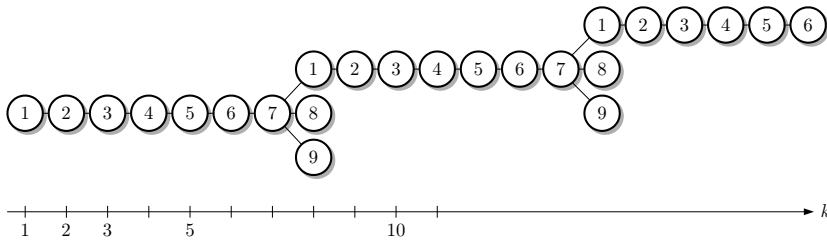
Die beiden Zustandsübergänge  $7 \rightarrow 1$  und  $7 \rightarrow 8$  beschreiben einen ordnungsgemäßen Transport des Werkstücks, aber das System geht in unterschiedliche Zustände über, weil im zweiten Fall die Blechrolle leer ist und die Stanze nicht weiterarbeiten kann. Es spielt sich im System ein Zustandsübergang ab, der die Blechrolle aus dem ausreichend gefüllten in den leeren Zustand überführt. Weil aus den zur Verfügung stehenden Messgrößen nicht eindeutig vorhergesagt werden kann, wann dieser Zustandsübergang eintritt, ist das Verhalten der Stanze nichtdeterministisch. Gleiches gilt für den nicht ordnungsgemäßen Transport des Werkstücks, der gleichermaßen die Stanze blockiert. Auch sein Auftreten kann nicht eindeutig vorhergesagt werden. Allgemein gilt:

|| Nichtdeterministische Vorgänge haben ihre Ursache in unbekannten oder im Modell nicht berücksichtigten Ereignissen, die auch von nicht bekannten äußeren Einflüssen (Störungen) auf das System verursacht werden können.



**Abb. 11.13:** Automatengraph des erweiterten Modells der Stanze

Im Vergleich zu Abb. 11.8 enthält der Automatengraph in Abb. 11.13 zwei neue Zustände und zwei neue Kanten. Der Nichtdeterminismus des Automaten kommt dadurch zum Ausdruck, dass vom Knoten 7 drei Kanten ausgehen. Die im Zustand 7 ("Die Stanze ist nach der Umformung des Werkstücks geöffnet.") enthaltenen Informationen reichen nicht aus, um eindeutig vorherzusagen, in welchen Nachfolgezustand die Stanze übergeht. Deshalb kann man lediglich die Menge  $\mathcal{B}$  der möglichen Zustandsfolgen angeben.



**Abb. 11.14:** Baum der Zustandsfolgen des nichtdeterministischen Automaten

Die Menge  $\mathcal{B}$  enthält alle Zustandsfolgen, die der Automat vom Anfangszustand  $z(0) = 1$  aus erzeugen kann. Wie der Automatengraph zeigt, enthalten diese Trajektorien die Folge

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$$

unterschiedlich oft und enden dann mit dem Übergang zum Zustand 8 oder zum Zustand 9. Man kann dies auch als Baum darstellen (Abb. 11.14), dessen Pfade die möglichen Zustandstrajektorien repräsentieren.

Dieses Beispiel verdeutlicht auch, dass ereignisdiskrete Modelle Trajektorien unterschiedlicher Länge erzeugen können, das Modell also nach einer endlichen Anzahl von Zustandsübergängen „stehenbleiben“ kann. Ein derartiges Verhalten ist von kontinuierlichen Systemen nicht bekannt. Der Grund für dieses neue Phänomen liegt in der Tatsache, dass man sich bei ereignisdiskreten Systemen für die Übergänge zwischen unterschiedlichen Zuständen interessiert und ein Verharren in einem Zustand deshalb keine neuen Informationen bringt. Der nächste Zustandsübergang liegt bei  $t = \infty$ , also außerhalb des Betrachtungshorizonts. Das System verbleibt für  $t < \infty$  in seinem letzten Zustand.

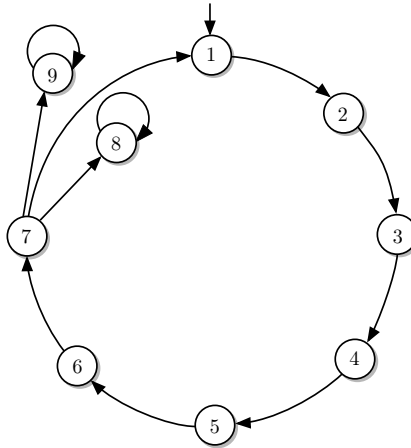
Die Matrixdarstellung dieses Automaten heißt

$$\begin{pmatrix} p_1(k+1) \\ p_2(k+1) \\ p_3(k+1) \\ p_4(k+1) \\ p_5(k+1) \\ p_6(k+1) \\ p_7(k+1) \\ p_8(k+1) \\ p_9(k+1) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \circ \begin{pmatrix} p_1(k) \\ p_2(k) \\ p_3(k) \\ p_4(k) \\ p_5(k) \\ p_6(k) \\ p_7(k) \\ p_8(k) \\ p_9(k) \end{pmatrix}, \quad \begin{pmatrix} p_1(0) \\ p_2(0) \\ p_3(0) \\ p_4(0) \\ p_5(0) \\ p_6(0) \\ p_7(0) \\ p_8(0) \\ p_9(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (11.27)$$

Die siebente Spalte der Matrix  $G$  enthält drei Einsen, weil vom Zustand 7 aus drei Zustandsübergänge möglich sind.

Bei der Interpretation der mit Hilfe von Gl. (11.26) erzeugten Zustandsfolge  $p(k)$  muss man aufpassen, weil nach dem Erreichen der Zustände 8 oder 9 keine weiteren Zustandsübergänge stattfinden.





**Abb. 11.15:** Automatengraph mit Schlingen an den Zuständen 8 und 9

Wenn einer dieser Zustände zum Zeitpunkt  $\bar{k}$  angenommen wird, so gibt es, streng genommen, keinen Nachfolgezeitpunkt  $\bar{k} + 1$ . Da es aber für andere Zustände diesen Nachfolgezeitpunkt gibt, beschreibt man das Verharren des Systems im Zustand 8 bzw. 9 damit, dass man es als Übergang in denselben Zustand deutet, also in den Automatengraphen Schlingen an den Knoten 8 und 9 einträgt bzw. in der Matrix  $G$  in der achten und neunten Zeile die Hauptdiagonalelemente gleich eins setzt. Die Matrix  $G$  heißt dann

$$G = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \quad (11.28)$$

und an Stelle der Abb. 11.13 und 11.14 erhält man die Abb. 11.15 und 11.16. □

#### **Beispiel 11.4** Modell einer Fertigungszelle mit vier Werkzeugmaschinen

Nichtdeterministisches Verhalten entsteht häufig bei parallel ablaufenden Prozessen, die nach unterschiedlicher bzw. nicht genau vorhersagbarer Dauer beendet werden. Die Doppelpfeile in Abb. 11.17 zeigen den Materialfluss zwischen vier Werkzeugmaschinen, der beispielsweise durch Förderbänder realisiert ist. Das von der Maschine  $M_1$  bearbeitete Werkstück wird entweder von der Maschine  $M_2$  oder von der Maschine  $M_3$  weiterverarbeitet, wobei eine Steuerung gewährleistet, dass das Werkstück an die jeweils freie Maschine transportiert wird. Die Maschine  $M_4$  erhält die Werkstücke von den Maschinen  $M_2$  oder  $M_3$  und gibt die bearbeiteten Werkstücke an den nachfolgenden Teil der Fertigungseinrichtung weiter.

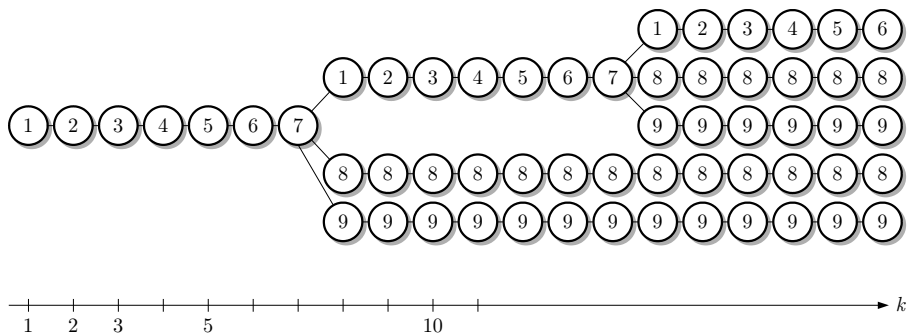


Abb. 11.16: Baum der Zustandsfolgen des nichtdeterministischen Automaten nach der Erweiterung

Das System soll unter der Wirkung einer Steuerung, die die beschriebenen Abläufe erzeugt, modelliert werden. Das Modell soll die Folge der Zustände der Werkzeugmaschinen wiedergeben, wobei jede Maschine zwei Zustände besitzt:

- 0 – Die Maschine arbeitet nicht.
- 1 – Die Maschine arbeitet.

Der Zustand der Fertigungszelle wird durch die vier Maschinenzustände beschrieben, die im Automatengraphen in Abb. 11.18 in derselben Weise wie die Maschinen in Abb. 11.17 angeordnet sind. Der Zustand

$$\begin{matrix} & 1 \\ 0 & 0 \\ & 0 \end{matrix}$$

besagt also, dass die Maschine  $M_2$  arbeitet, während die anderen drei Maschinen im Wartezustand sind.

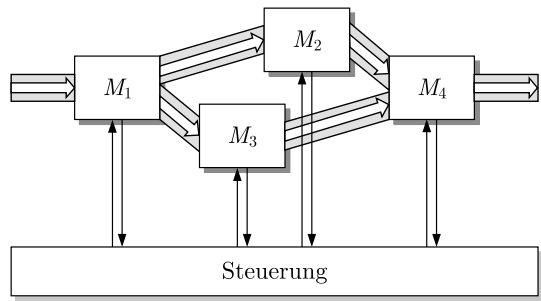
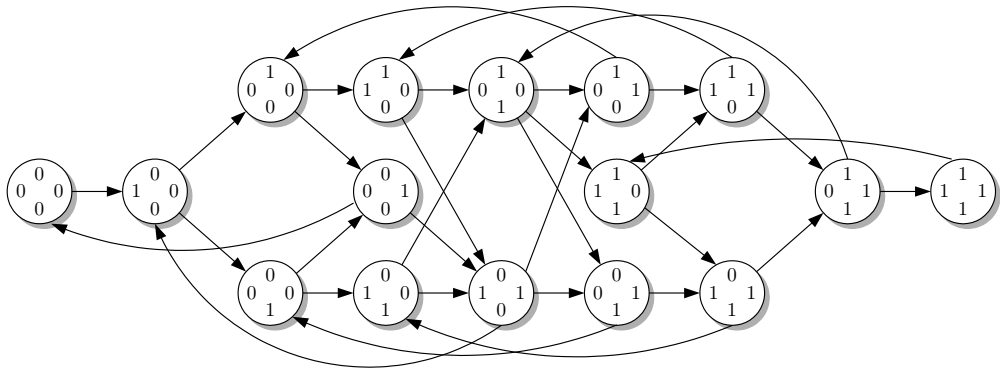


Abb. 11.17: Materialfluss zwischen vier Werkzeugmaschinen

Abbildung 11.18 zeigt den nichtdeterministischen Automaten, der die Arbeitsweise der vier Maschinen beschreibt. Das Verhalten ist nichtdeterministisch, weil bei Betrachtung der logischen Folge der Zustandsübergänge nicht entschieden werden kann, welche Maschine als nächstes ihren Bearbeitungsvorgang abschließen und damit einen Zustandsübergang auslösen wird. Für viele Zustände gibt es deshalb mehrere Nachfolgezustände. Beispielsweise sind die Zustandsübergänge

$$\begin{array}{ccc} 0 & & 0 \\ 0 & 0 \rightarrow & 0 \quad 1 \quad \text{und} \quad 0 \quad 0 \rightarrow & 1 \quad 0 \\ 1 & & 0 & 1 & 1 \end{array}$$

möglich, wobei im ersten Fall die Maschine  $M_3$  ihre Bearbeitung beendet und das Werkstück an die freie Maschine  $M_4$  weitergibt, während im zweiten Fall die Maschine  $M_3$  weiterarbeitet und die Maschine  $M_1$  ein Werkstück erhält und die Bearbeitung beginnt. Dieses Beispiel demonstriert den Nichtdeterminismus paralleler Prozesse.



**Abb. 11.18:** Beschreibung der Arbeitsweise der vier Werkzeugmaschinen durch einen nichtdeterministischen Automaten

Ein weiterer Nichtdeterminismus entsteht dadurch, dass bei freien Maschinen  $M_2$  und  $M_3$  das Werkstück wahlweise zu einer der beiden Maschinen transportiert werden kann, nachdem es von der Maschine  $M_1$  bearbeitet wurde. Dieser im Automaten in Abb. 11.18 erfasste Nichtdeterminismus wird allerdings durch die Steuerung aufgelöst, die eindeutig den Zielort des Werkstücktransports vorgeben muss, wobei sich die Steuerung nach übergeordneten Gesichtspunkten für einen der zur Wahl stehenden Übergänge entscheidet.  $\square$

#### **Aufgabe 11.6** Ereignisdiskrete Beschreibung des Startvorgangs eines Fahrzeugs

Einsteigen und losfahren – wie oft haben Sie Ihr Auto schon gestartet, ohne sich die dabei ablaufende Ereignisfolge im Einzelnen zu vergegenwärtigen? Eine Reihe von Ereignissen müssen abgelaufen sein, bevor Sie losfahren können, z. B. das Türenschließen, das Starten des Motors, das Anlegen des Sicherheitsgurts. Dabei sind nichtdeterministische Folgen zugelassen, denn ob Sie den Gurt vor dem Starten des Motors anlegen oder danach, ist gleichgültig. Andererseits müssen bestimmte Ereignisse in der richtigen Reihenfolge auftreten, beispielsweise das Einlegen des ersten Ganges nach dem Auskuppeln.

Stellen Sie einen autonomen nichtdeterministischen Automaten auf, der alle zulässigen Ereignisfolgen vom Aufschließen der Autotür bis zum Anfahren des Autos erzeugen kann.  $\square$

**Aufgabe 11.7\***    *Modellierung einer Fertigungszelle*

Das Verhalten der in Abb. 1.23 auf S. 39 dargestellten Fertigungszelle einschließlich der Automatisierungseinrichtung soll durch einen Automaten dargestellt werden. Da die Fertigungszelle zusammen mit der Steuerung betrachtet wird, handelt es sich um ein autonomes System. Das Modell soll beschreiben, welche Zustandsfolge dieses System durchläuft. Dafür werden folgende Zustände eingeführt:

Zustand	Beschreibung
<b>Greifer:</b>	
<i>I</i>	Der Greifer ist im Ruhezustand.
<i>T</i> <sub>1</sub>	Der Greifer transportiert ein Werkstück vom Band zur Maschine <i>M</i> <sub>1</sub> .
<i>T</i> <sub>2</sub>	Der Greifer transportiert ein Werkstück von der Maschine <i>M</i> <sub>1</sub> zur Maschine <i>M</i> <sub>2</sub> .
<b>Maschine <i>M</i><sub>1</sub>:</b>	
<i>I</i>	Die Maschine <i>M</i> <sub>1</sub> ist im Ruhezustand.
<i>A</i>	Die Maschine <i>M</i> <sub>1</sub> bearbeitet ein Werkstück.
<i>W</i>	Die Maschine <i>M</i> <sub>1</sub> hat die Bearbeitung beendet und wartet auf den Werkstückwechsel.
<b>Maschine <i>M</i><sub>2</sub>:</b>	
<i>I</i>	Die Maschine <i>M</i> <sub>2</sub> ist im Ruhezustand.
<i>A</i>	Die Maschine <i>M</i> <sub>2</sub> bearbeitet ein Werkstück.

Die Steuerung der Fertigungszelle erfolgt dezentral, d. h., der Greifer und die beiden Maschinen haben eigene Steuerungen, die nach folgenden Regeln arbeiten:

- Greifer:

Wenn die Maschine *M*<sub>1</sub> im Ruhezustand ist, wird ein Teil vom Transportband zu *M*<sub>1</sub> transportiert.

Wenn die Maschine *M*<sub>1</sub> die Bearbeitung beendet hat und die Maschine *M*<sub>2</sub> im Ruhezustand ist, wird das Teil von *M*<sub>1</sub> zu *M*<sub>2</sub> transportiert
- Maschine *M*<sub>1</sub> :

Wenn die Maschine *M*<sub>1</sub> ein neues Werkstück erhält, beginnt sofort dessen Bearbeitung.

Wenn die Maschine *M*<sub>1</sub> die Bearbeitung abgeschlossen hat, wartet sie auf den Werkstückwechsel.

Wenn das Werkstück von der Maschine *M*<sub>1</sub> abtransportiert ist, geht *M*<sub>1</sub> in den Ruhezustand über.
- Maschine *M*<sub>2</sub> :

Wenn die Maschine *M*<sub>2</sub> ein neues Werkstück erhält, beginnt sofort dessen Bearbeitung.

Wenn die Maschine *M*<sub>2</sub> die Bearbeitung abgeschlossen hat, transportiert sie das Werkstück in die Ablage und geht in den Ruhezustand über.

Es wird bei der Modellierung davon ausgegangen, dass stets genügend Teile auf dem Transportband auf die Bearbeitung warten und dass der Transport der Werkstücke durch den Greifer schneller geht als die Bearbeitung der Teile durch die beiden Maschinen.

Beschreiben Sie das Verhalten der Fertigungszelle durch einen Automaten. Ist der Automat deterministisch? □

### 11.3.3 Nichtdeterministische Automaten mit Eingang und Ausgang

Die Dynamik nichtdeterministischer Automaten mit Eingang und Ausgang (nichtdeterministischer E/A-Automat)

Nichtdeterministischer E/A-Automat:  $\mathcal{N} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, L, \mathcal{Z}_0)$

ist durch die *Verhaltensrelation*  $L$  beschrieben

$$L : \mathcal{Z} \times \mathcal{W} \times \mathcal{Z} \times \mathcal{V} \rightarrow \{0, 1\}. \quad (11.29)$$

Die Verhaltensrelation gibt an, welche Zustandsübergänge von  $z$  nach  $z'$  stattfinden *können*, wenn der Automat den Eingang  $v$  erhält und dabei den Ausgang  $w$  erzeugt. Für diese Tupel  $(z', w, z, v)$  hat  $L$  den Wert eins

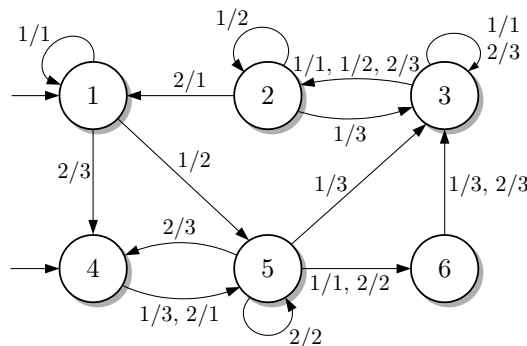
$$L(z', w, z, v) = 1 \quad (11.30)$$

und für die Tupel, die nicht zum Automatenverhalten gehören, den Wert null.

Wie bei der Zustandsübergangsrelation  $G$  des autonomen nichtdeterministischen Automaten resultiert der Begriff Verhaltensrelation aus der Tatsache, dass  $L$  aus der Menge aller möglichen Tupel  $(z', w, z, v)$  diejenigen kennzeichnet, die am Automat auftreten können. Man kann diese Relation deshalb als Menge  $\mathcal{L} \subseteq \mathcal{Z} \times \mathcal{W} \times \mathcal{Z} \times \mathcal{V}$  aufschreiben, die mit der in Gl. (11.29) eingeführten Funktion in folgender Beziehung steht:

$$\mathcal{L} = \{(z', w, z, v) \mid L(z', w, z, v) = 1\}. \quad (11.31)$$

Die Darstellungen der Verhaltensrelation als Menge  $\mathcal{L}$  oder als Funktion  $L$  haben dieselbe Bedeutung. Die mathematische Bezeichnung der Verhaltensrelation bezieht sich auf die Mengendarstellung  $\mathcal{L}$ . In der Mengentheorie wird die Funktion  $L$ , die diese Menge entsprechend Gl. (11.31) definiert, als charakteristische Funktion der Menge  $\mathcal{L}$  bezeichnet.



**Abb. 11.19:** Nichtdeterministischer Automat mit Eingang und Ausgang

Im Automatengraphen werden die zu den Zustandsübergängen gehörenden Paare aus Eingang  $v$  und Ausgang  $w$  wie beim deterministischen Automaten an die Kanten geschrieben. Abbildung 11.19 zeigt ein Beispiel. Die Funktion  $L$  hat für diejenigen Tupel den Wert eins, für die der Automatengraph eine Kante besitzt:

$$L(1, 1, 1, 1) = 1$$

$$L(5, 2, 1, 1) = 1$$

$$L(6, 2, 5, 2) = 1$$

$$L(6, 1, 5, 1) = 1$$

$$L(3, 3, 2, 1) = 1$$

usw.

### Beispiel 11.5 Modell eines Regensors

Mit Hilfe eines Regensors soll der Scheibenwischer an der Frontscheibe eines Personewagens in Abhängigkeit vom Regen gesteuert werden. Das Messprinzip beruht auf der Bestimmung des Brechungsindex, wofür eine Diode Licht ausstrahlt und das an der Scheibe reflektierte Licht misst. Ist die Änderung des Brechungsindex größer als ein Schwellwert, so wird Regenwasser auf der Windschutzscheibe detektiert.

Dieses Messprinzip ist anfällig gegenüber Licht, das von außen auf die Scheibe fällt, sowie Verschmutzungen der Scheibe durch Staub, Fliegen oder Schneematsch. Deshalb ist beispielsweise Spritzwasser schlecht erkennbar, während der Sensor normalen Regen gut detektiert. Der Regensor verhält sich deshalb nichtdeterministisch.

Um den Regensor durch einen nichtdeterministischen Automaten beschreiben zu können, werden folgende Größen definiert:

$v$	Bedeutung
1	Es regnet nicht.
2	Es regnet.
$w$	Bedeutung
1	Anzeige: „Kein Regen“
2	Anzeige: „Regen“
$z$	Bedeutung
1	Der Sensor zeigt keinen Regen an.
2	Der Sensor zeigt Regen an.

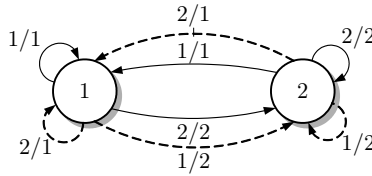
Abbildung 11.20 gibt den Graphen eines nichtdeterministischen E/A-Automaten wieder. Für eine gute Funktion des Sensors versucht man, die gestrichelt gezeichneten Zustandsübergänge zu vermeiden, wofür man z. B. die Sensorschwellwerte geeignet festlegen muss.  $\square$

Die Bewegung des Automaten wird durch alle Tupel  $(z(k+1), w(k), z(k), v(k))$  beschrieben, die die Relation

Zustandsraumdarstellung nichtdeterministischer E/A-Automaten:

$$\mathcal{N} : (z(k+1), w(k), z(k), v(k)) \in \mathcal{L}, \quad z(0) \in \mathcal{Z}_0$$

(11.32)



**Abb. 11.20:** Beschreibung eines Regensensors durch einen nichtdeterministischen Automaten

erfüllen. Gleichung (11.32) ersetzt also die beiden Gleichungen der Zustandsraumdarstellung (11.12) des deterministischen Automaten. Sie kann auch in der Form

$$L(z(k+1), w(k), z(k), v(k)) = 1, \quad z(0) \in \mathcal{Z}_0 \quad (11.33)$$

geschrieben werden.

Die Darstellung (11.32) des nichtdeterministischen Automaten ist im Vergleich zu allen bisher eingeführten Modellen insofern neuartig, als dass der Nachfolgezustand und die Ausgabe gemeinsam durch *eine* Relation  $\mathcal{L}$  beschrieben werden, während sowohl bei den kontinuierlichen Systemen als auch bei den bisher betrachteten ereignisdiskreten Systemen die Zustandsänderung und die Ausgabe in zwei getrennten Gleichungen festgelegt wurden. Deshalb erhebt sich die Frage, warum man jetzt auf eine neue Beschreibungsform übergehen muss.

Zur Beantwortung dieser Frage kann man sich zunächst überlegen, dass man auch für deterministische Automaten eine Funktion  $L_d$  definieren kann, die mit den in Gl. (11.12) vorkommenden Funktionen  $G$  und  $H$  in der Beziehung

$$L_d(z', w, z, v) = 1 \iff \begin{aligned} z' &= G(z, v) \\ w &= H(z, v) \end{aligned} \quad (11.34)$$

steht. Die Verhaltensrelation  $\mathcal{L}_d$  erhält man dann analog zu Gl. (11.31) aus der Beziehung

$$\mathcal{L}_d = \{(z', w, z, v) \mid L_d(z', w, z, v) = 1\}.$$

Offenbar gilt für den deterministischen Automaten die Beziehung (11.32):

$$(z(k+1), w(k), z(k), v(k)) \in \mathcal{L}_d.$$

Eine wichtige Eigenschaft von  $\mathcal{L}_d$  besteht darin, dass diese Menge für jedes Zustand-Eingangs-Paar  $(z(k), v(k))$  den Nachfolgezustand  $z(k+1)$  und die Ausgabe  $w(k)$  eindeutig festlegt. Da dies so ist, kann man das Verhalten deterministischer Automaten durch die beiden Gleichungen auf der rechten Seite von (11.34) beschreiben und muss nicht die Verhaltensrelation  $L_d$  verwenden.

Dies ist bei nichtdeterministischen Automaten anders. Da die Verhaltensrelation  $\mathcal{L}$  einem Zustands-Eingangs-paar  $(z(k), v(k))$  mehrere Nachfolgezustände  $z(k+1)$  und mehrere Ausgaben  $w(k)$  zuordnen kann, muss das Modell nichtdeterministischer Automaten beschreiben, welcher der möglichen Zustandsübergänge  $z(k) \rightarrow z(k+1)$  mit welcher Ausgabe  $w(k)$  gemeinsam auftritt. Man braucht deshalb eine Funktion  $\tilde{L}$ , die dem Paar  $(z(k), v(k))$  die Menge der möglichen Paare  $(z(k+1), w(k))$  zuordnet

$$\mathcal{N} : (z(k+1), w(k)) \in \tilde{L}(z(k), v(k)). \quad (11.35)$$

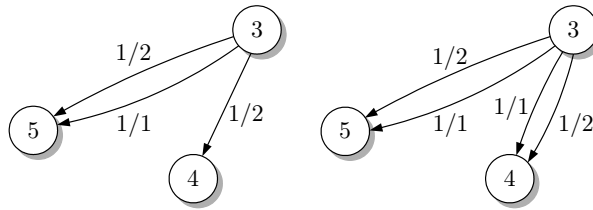
Es ist also, zumindest im allgemeinen Fall, nicht möglich, den Zustandsübergang und die Ausgabe durch zwei getrennte Gleichungen zu beschreiben, die bei nichtdeterministischen Automaten die Form

$$\mathcal{N} : \begin{cases} z(k+1) \in G(z(k), v(k)), & z(0) \in \mathcal{Z}_0 \\ w(k) \in H(z(k), v(k)) \end{cases} \quad (11.36)$$

hätten.

An Stelle von  $\tilde{L}$  in Gl. (11.35) arbeitet man mit der Relation  $\mathcal{L}$  und stellt das Verhalten durch Gl. (11.32) dar, was nicht die Aussage, sondern nur die mathematische Form dieser Aussage verändert.

**Nichtdeterministischer Mealy-Automat.** Ein nichtdeterministischer Mealy-Automat unterscheidet sich von dem bisher angegebenen nichtdeterministischen Automaten dadurch, dass die Zustandsübergangsrelation  $G$  von der Ausgaberation  $H$  getrennt werden kann. Es gelten für diese spezielle Automatenklasse also gerade die Gleichungen (11.36).



**Abb. 11.21:** Ausschnitte aus zwei Automatengraphen

Abbildung 11.21 zeigt, welche spezielle Eigenschaft nichtdeterministische Mealy-Automaten haben. Alle Kanten der Automatengraphen gelten für  $v = 1$ . Der linke Automat kann bei den beiden Zustandsübergängen  $3 \rightarrow 4$  und  $3 \rightarrow 5$  die Ausgabe  $w = 2$  erzeugen, aber nur beim Übergang  $3 \rightarrow 5$  die Ausgabe  $w = 1$ . Diese Tatsache kann nicht als Mealy-Automat (11.36) dargestellt werden, denn dann müssten die Funktionen  $G$  und  $H$  die Beziehungen

$$\begin{aligned} G(3, 1) &= \{4, 5\} \\ H(3, 1) &= \{1, 2\} \end{aligned}$$

erfüllen, die zulassen, dass der Automat auch beim Zustandsübergang  $3 \rightarrow 4$  die Ausgabe  $w = 1$  erzeugt. Andererseits kann man diesen Teil des Automatengraphen durch eine Verhaltensrelation darstellen, wenn die Menge  $\mathcal{L}$  die drei zu den gezeigten Kanten gehörenden Tupel beinhaltet

$$\mathcal{L} = \{...(5, 2, 3, 1), (5, 1, 3, 1), (4, 2, 3, 1), ...\}$$

und  $(4, 1, 3, 1)$  nicht zu  $\mathcal{L}$  gehört. Der rechte Teil der Abbildung zeigt einen nichtdeterministischen Zustandsübergang, der durch einen nichtdeterministischen Mealy-Automat beschrieben



werden kann. Hier treten die Ausgaben 1 und 2 bei beiden Übergängen auf. Auf diesen Automatengraphen kommt man, wenn man für den linken Automaten die Funktionen  $G$  und  $H$  aufstellt.

**Beispiel 11.5 (Forts.)** *Modell eines Regensors*

Der in Abb. 11.20 gezeigte Automat wird durch die Verhaltensrelation

$$L = \begin{array}{c|c|c|c} z' & w & z & v \\ \hline 1 & 1 & 1 & 1 \\ 2 & 2 & 1 & 1 \\ 1 & 1 & 1 & 2 \\ 2 & 2 & 1 & 2 \\ 1 & 1 & 2 & 1 \\ 2 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{array}$$

dargestellt. Im Zustand  $z = 2$  wird bei der Eingabe  $v = 1$  dem Zustandswechsel nach  $z' = 1$  die Ausgabe  $w = 1$  und dem Wechsel nach  $z' = 2$  die Ausgabe  $w = 2$  zugeordnet. Es handelt sich bei diesem Automaten also nicht um einen nichtdeterministischen Mealy-Automaten.  $\square$

**Verhalten nichtdeterministischer E/A-Automaten.** Für die Eingangsfolge

$$V(0...k_e) = (v(0), v(1), \dots, v(k_e))$$

kann der Automat die Ausgangsfolge

$$W(0...k_e) = (w(0), w(1), \dots, w(k_e))$$

erzeugen, wenn es eine Zustandsfolge

$$Z(0...k_e+1) = (z(0), z(1), \dots, z(k_e+1))$$

gibt, mit der Gl. (11.33) für alle  $k = 0, 1, \dots, k_e$  gilt. Zusammengehörige E/A-Folgen werden zur Menge  $\mathcal{B}$  zusammengefasst, die das Verhalten des Automaten beschreibt:

$$\mathcal{B} = \{((v(0), v(1), \dots, v(k_e)), (w(0), w(1), \dots, w(k_e))) \mid \exists(z(0), z(1), \dots, z(k_e+1)) : \\ L(z(k+1), w(k), z(k), v(k)) = 1 \text{ für } k = 0, 1, \dots, k_e\}. \quad (11.37)$$

**Matrixdarstellung nichtdeterministischer Automaten.** Die Matrixdarstellung (11.26) kann auf Automaten mit Eingängen und Ausgängen verallgemeinert werden:

$$\mathcal{N} : \begin{cases} \mathbf{p}(k+1) = \mathbf{G}(v(k)) \circ \mathbf{p}(k), & k = 0, 1, 2, \dots, \\ \mathbf{p}_w(k) = \mathbf{H}(v(k)) \circ \mathbf{p}(k), \end{cases} \quad (11.38)$$

wobei  $G$  und  $H$  zwei Matrizen sind, die vom Eingang  $v$  abhängen.  $p_w(k)$  ist ein Vektor, dessen  $i$ -tes Element gleich eins ist, wenn der Ausgang  $w$  den Wert  $i$  annehmen kann. Bei dieser Matrixdarstellung ist der Zusammenhang zwischen den Zustandsübergängen und den dabei erzeugten Ausgaben verloren gegangen. Sie ist also ohne Informationsverlust nur für nichtdeterministische Mealy-Automaten anwendbar.

#### **Aufgabe 11.8\*** Funktion einer Raumbeleuchtung

Es soll ein Modell aufgestellt werden, das die Funktionsfähigkeit einer Raumbeleuchtung beschreibt. Die Zustände der Beleuchtungseinrichtung werden durch Aussagen wie „Die Beleuchtungseinrichtung ist außer Betrieb.“, „Die Beleuchtungseinrichtung ist betriebsbereit.“ oder „Die Lampe ist defekt.“ charakterisiert und die Ausgabe der Raumbeleuchtung durch die Aussagen „Die Lampe brennt.“ und „Die Lampe brennt nicht.“ Die Eingabe beschreibt das Heraus- bzw. Hereinschrauben einer Sicherung und das Ein- oder Ausschalten des Schalters. Nehmen Sie zur Vereinfachung der Modellbildung an, dass die Lampe nicht während des Betriebs, sondern nur beim Schließen des Stromkreises durchbrennen kann.

Stellen Sie die Zustandsübergangsrelation  $G$  auf und geben Sie die Matrixdarstellung des Automaten an. Welche Zustandsfolge erzeugt der Automat, wenn als erstes die Sicherung hereingeschraubt wird, dann der Schalter zweimal ein- und ausgeschaltet wird und zuletzt die Sicherung herausgeschraubt wird?

Wie muss das Modell für den Fall erweitert werden, dass der Kontakt im Schalter beim Einschalten nicht funktioniert, der Schalter aber beim nächsten Ein- und Ausschalten wieder funktioniert? ☐

#### **Aufgabe 11.9** Beschreibung eines Kartentelefons

Beschreiben Sie durch einen Automaten den Ablauf des Telefonierens mit einem Kartentelefon, wobei Sie die Tätigkeiten der telefonierenden Person als Eingaben und die vom Telefon an den Nutzer ausgegebenen Informationen als Ausgaben auffassen. Wäre es bei diesem Beispiel wichtig, die zeitlichen Abstände der Ereignisse im Modell zu erfassen? ☐

## 11.4 Stochastische Automaten

### 11.4.1 Stochastische Prozesse

Nichtdeterministische Automaten beschreiben Systeme, deren Zustandsfolgen nicht eindeutig vorhergesagt werden können. Sie führen entsprechend Gl. (11.25) auf die Menge  $\mathcal{Z}(k)$  der zur Zeit  $k$  möglichen Zustände. Allerdings sind keine Aussagen darüber möglich, welche Zustände dieser Menge häufiger und welche weniger häufig angenommen werden.

In vielen technischen Anwendungen kann man beim Auftreten eines derartigen Nichtdeterminismus Aussagen über die Häufigkeit machen, mit denen die nichtdeterministischen Zustandsübergänge erfolgen. So wird die Stanze aus Beispiel 11.2 bei ordnungsgemäßer Justage und Steuerung fast immer in den Ausgangszustand zurückkehren und nur sehr selten einen der beiden Fehlerzustände annehmen. Bei der im Folgenden behandelten Erweiterung nichtdeterministischer zu stochastischen Automaten wird mit der Auftrittswahrscheinlichkeit der Zustände ein Maß eingeführt, das diese Unterscheidung möglich macht.

Die Beschreibung diskreter Systeme durch stochastische Automaten beruht auf der Interpretation des Systems als ein *stochastischer Prozess*, also als ein Prozess, dessen Verhalten durch den Zufall mitbestimmt wird. Der stochastische Prozess in Abb. 11.22 hat den Eingang  $V(k)$ , den Zustand  $Z(k)$  und den Ausgang  $W(k)$ . Die hier verwendeten großen Buchstaben zeigen, dass es sich bei diesen Größen um *stochastische Variablen* handelt, also um Variablen, deren Werte vom Zufall abhängen. So ist  $V(k)$  die zufällige Variable, die die Systemeingabe zur Zeit  $k$  beschreibt. Ihre möglichen Werte werden durch den kleinen Buchstaben  $v \in \mathcal{V}$  symbolisiert.



Abb. 11.22: Stochastischer Prozess

**Autonomer stochastischer Prozess.** Um die Idee des stochastischen Prozesses zu erläutern, wird zunächst wieder ein System ohne Eingang betrachtet, der sich im Anfangszustand  $Z(0) = z_0$  befindet. Der Nachfolgezustand  $Z(1)$  kann nicht eindeutig vorhergesagt werden, weil der Übergang vom Anfangszustand zum Nachfolgezustand vom Zufall abhängt. Gleiches gilt für alle weiteren Zustände  $Z(k)$ , ( $k \geq 1$ ). Die Beschreibung des Systemverhaltens durch die Wahrscheinlichkeit  $\text{Prob}(Z(k) = z)$  beruht auf der Vorstellung, dass man mit dem stochastischen Prozess sehr oft „experimentiert“, indem man ihn in einen Anfangszustand bringt und die Zustandsfolge  $Z(0), Z(1), Z(2), \dots$  beobachtet. Die Wahrscheinlichkeit  $\text{Prob}(Z(k) = z)$  sagt dann aus, mit welcher Häufigkeit der Prozess zum Zeitpunkt  $k$  im Zustand  $z$  war. Im Sinne der Wahrscheinlichkeitstheorie ist damit genauer die relative Häufigkeit

$$\frac{N_z(k)}{N} \quad (11.39)$$

gemeint, wobei  $N$  die Gesamtanzahl der durchgeführten „Experimente“ bezeichnet und  $N_z(k)$  die Anzahl der Experimente, in denen sich der Prozess zum Zeitpunkt  $k$  im Zustand  $z$  befunden hat. Führt man unendlich viele Experimente aus, so geht diese relative Häufigkeit in die Wahrscheinlichkeit über, mit der der Zustand  $z$  zum Zeitpunkt  $k$  angenommen wird:

$$\frac{N_z(k)}{N} \xrightarrow{N \rightarrow \infty} \text{Prob}(Z(k) = z). \quad (11.40)$$

Die Erweiterung nichtdeterministischer zu stochastischen Automaten soll es also möglich machen, für jeden Zeitpunkt  $k$  die Häufigkeit  $\text{Prob}(Z(k) = z)$  vorherzusagen, mit der das betrachtete System die Zustände  $z \in \mathcal{Z}$  annimmt. Dafür braucht man den Wert der Wahrscheinlichkeit  $\text{Prob}(Z(k) = z)$  nicht nur für einen Zustand  $z$ , sondern für alle Zustände der Zustandsmenge  $\mathcal{Z}$  und für alle Zeitpunkte  $k$  im betrachteten Zeitintervall  $0 \dots k_e$ . Für eine festgelegte Zeit  $k$  ist die Menge aller Werte  $\text{Prob}(Z(k) = z)$  für  $z \in \mathcal{Z}$  die Zustandswahrscheinlichkeitsverteilung. Da sich der Automat in genau einem Zustand  $z \in \mathcal{Z}$  befindet, gilt

$$\sum_{z \in \mathcal{Z}} \text{Prob}(Z(k) = z) = 1, \quad k = 0, 1, 2, \dots$$

Um die Bewegung des Systems zu beschreiben, muss man diese Wahrscheinlichkeitsverteilung für alle Zeitpunkte  $k = 0, 1, 2, \dots$  bestimmen.

Diese ausführliche Erläuterung soll zeigen, was durch einen Wert der Wahrscheinlichkeit ausgesagt wird und was nicht.  $\text{Prob}(Z(k) = z)$  beschreibt die relative Häufigkeit, mit der das betrachtete System zum Zeitpunkt  $k$  im Zustand  $z$  ist, wenn man unendlich oft mit dem System experimentiert. Damit weiß man mehr, als wenn man das System durch einen nichtdeterministischen Automaten beschreiben würde, mit dem nur die Mengen  $\mathcal{Z}(k)$ , ( $k = 0, 1, \dots, k_e$ ) derjenigen Zustände berechenbar sind, die das System zu den Zeitpunkten  $k \in [0, k_e]$  annehmen kann. Wenn man einen nichtdeterministischen Automaten zu einem stochastischen Automaten erweitert, gilt die Beziehung

$$\mathcal{Z}(k) = \{z \mid \text{Prob}(Z(k) = z) > 0\}.$$

Der stochastische Automat führt also auf dieselben Mengen  $\mathcal{Z}(k)$ , sagt aber darüber hinaus etwas über die Häufigkeit aus, mit der die Elemente dieser Mengen den Zustand repräsentieren.

Alle diese Aussagen gelten im Mittel für eine große Anzahl von Experimenten. Was in einem konkreten Experiment geschieht, kann man auch mit der zusätzlichen Information, die in der Zustandswahrscheinlichkeitsverteilung steckt, nicht genauer vorhersagen als mit einem nichtdeterministischen Automaten. Die Wahrscheinlichkeitsbetrachtungen lassen keine Verkleinerung der Menge  $\mathcal{Z}(k)$  zu.

**Zustandsübergangswahrscheinlichkeit stochastischer Prozesse.** Für den stochastischen Prozess soll ein Modell aufgestellt werden, das es ermöglicht, die Folge

$$\text{Prob}(Z(0) = z), \text{Prob}(Z(1) = z), \text{Prob}(Z(2) = z), \dots, \text{Prob}(Z(k_e) = z), \quad z \in \mathcal{Z}$$

von Zustandswahrscheinlichkeitsverteilungen zu berechnen. Dafür muss die Modellform „nicht-deterministischer Automat“ zur Modellform „stochastischer Automat“ erweitert werden, indem man die Verhaltensrelation durch die *Zustandsübergangswahrscheinlichkeit* ersetzt.

Für einen autonomen stochastischen Prozess ist die Zustandsübergangswahrscheinlichkeit die bedingte Wahrscheinlichkeit

$$g_{z'z} = \text{Prob}(Z(k+1) = z' \mid Z(k) = z), \quad (11.41)$$

mit der der Automat zum Zeitpunkt  $k+1$  den Zustand  $z'$  annimmt, wenn er zum Zeitpunkt  $k$  im Zustand  $z$  war. Diese Wahrscheinlichkeit muss für alle Zustandspaare  $z, z'$  bekannt sein. Sie beschreibt das dynamische Verhalten des stochastischen Prozesses durch die relative Häufigkeit, mit der das System den Zustandsübergang  $z \rightarrow z'$  ausführt, wenn man es unendlich oft in den Zustand  $z$  bringt.

Bei der Erweiterung des in Abb. 11.12 auf S. 357 gezeigten nichtdeterministischen Automaten will man in das Modell beispielsweise die Kenntnis aufnehmen, dass der Zustand 4 dem Zustand 1 in 80% aller Fälle folgt, in denen das System im Zustand 1 war, während nur in 15% dieser Fälle der Zustand 5 als Nachfolgezustand auftritt und in 5% der Zustand 1 wiederholt wird. Dann gilt

$$\begin{aligned}
g_{41} &= \text{Prob}(Z(k+1) = 4 \mid Z(k) = 1) = 0,8 \\
g_{51} &= \text{Prob}(Z(k+1) = 5 \mid Z(k) = 1) = 0,15 \\
g_{11} &= \text{Prob}(Z(k+1) = 1 \mid Z(k) = 1) = 0,05.
\end{aligned}$$

Es wird vorausgesetzt, dass sich die Wahrscheinlichkeit  $g_{z'z}$  nicht mit der Zeit  $k$  ändert, so dass Gl. (11.41) für beliebige Zeitpunkte  $k$ , insbesondere auch für  $k = 0$  gilt:

$$g_{z'z} = \text{Prob}(Z(1) = z' \mid Z(0) = z). \quad (11.42)$$

Da sich der stochastische Prozess bei jedem Zustandsübergang in einen Nachfolgezustand bewegt, muss

$$\sum_{z' \in \mathcal{Z}} g_{z'z} = 1, \quad z \in \mathcal{Z} \quad (11.43)$$

gelten.

Die Anwendung der Zustandsübergangswahrscheinlichkeitsverteilung (11.42) kann man sich so vorstellen, dass ein Würfel für jeden Zustand  $Z(k)$  den Nachfolgezustand  $Z(k+1)$  festlegt. Der Würfel soll allerdings nach der Zustandsübergangswahrscheinlichkeit vorgehen, also nur Zustände  $Z(1) = z'$  auswählen, für die die Beziehung

$$g_{z'z_0} = \text{Prob}(Z(1) = z' \mid Z(0) = z_0) > 0$$

gilt, und dabei diejenigen Nachfolgezustände bevorzugen, für die die Zustandsübergangswahrscheinlichkeit besonders groß ist. Wenn der Würfel auf diese Weise auch die nachfolgenden Zustandsübergänge festlegt, entsteht eine Zustandstrajektorie

$$Z = (z(0), z(1), z(2), \dots),$$

die für das aktuelle Experiment eindeutig ist, aber beim nächsten Experiment anders aussehen kann.

**Gesteuerte stochastische Prozesse.** Die bisher für autonome stochastische Prozesse durchgeführten Betrachtungen können in drei Richtungen verallgemeinert werden. Erstens muss der Anfangszustand nicht bekannt sein, sondern nur eine Wahrscheinlichkeitsverteilung

$$p_0(z) = \text{Prob}(Z(0) = z), \quad z \in \mathcal{Z}, \quad (11.44)$$

die angibt, mit welcher Wahrscheinlichkeit der Anfangszustand  $Z(0)$  jeden der Werte  $z \in \mathcal{Z}$  annimmt. Diese Wahrscheinlichkeitsverteilung wird im Folgenden mit  $p_0(z)$  abgekürzt.

Zweitens kann der stochastische Prozess durch einen Eingang beeinflusst werden, der selbst wieder eine stochastische Größe  $V(k)$  ist. Auch der Eingang kann dann nur durch eine Wahrscheinlichkeitsverteilung

$$\text{Prob}(V(k) = v), \quad v \in \mathcal{V}$$

beschrieben werden, nach der bei einem Experiment mit dem stochastischen Prozess der aktuelle Wert  $v(k)$  ausgewürfelt wird.

Drittens kann der stochastische Prozess auch einen Ausgang besitzen, dessen Wert in Abhängigkeit vom aktuellen Zustand und der aktuellen Eingabe zufällig ausgewählt wird. Alle diese Erweiterungen sind bei dem im folgenden Abschnitt eingeführten stochastischen Automaten berücksichtigt.

### 11.4.2 Autonome stochastische Automaten

Der autonome stochastische Automat

Autonomer stochastischer Automat:  $\mathcal{S} = (\mathcal{Z}, G, p_0(z))$

ist beschrieben durch seine Zustandsmenge  $\mathcal{Z}$ , seine Zustandsübergangswahrscheinlichkeitsverteilung

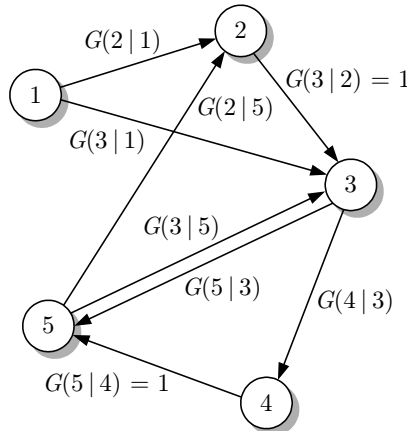
$$G : \mathcal{Z} \times \mathcal{Z} \rightarrow [0, 1]$$

und die Wahrscheinlichkeitsverteilung  $p_0(z)$  des Anfangszustands nach Gl. (11.44). In der älteren Literatur bezeichnet man den stochastischen Automaten auch als Wahrscheinlichkeitsautomaten, in der mathematischen Literatur als *Markovkette*.

Die Wahrscheinlichkeitsverteilung  $G$  setzt sich aus den in Gl. (11.42) definierten Zustandsübergangswahrscheinlichkeiten für alle Paare  $(z, z')$  zusammen:

$$G(z' | z) = g_{z'z} = \text{Prob}(z(k+1) = z' | z(k) = z), \quad z', z \in \mathcal{Z}.$$

Um hervorzuheben, dass  $G$  eine bedingte Wahrscheinlichkeit ist, wird  $G(z' | z)$  an Stelle von  $G(z', z)$  geschrieben und im Folgenden für diese Wahrscheinlichkeit eine der drei in der letzten Gleichung angegebenen Symbole verwendet.



**Abb. 11.23:** Automatengraph eines autonomen stochastischen Automaten

Die Erweiterung des nichtdeterministischen zum stochastischen Automaten wird im Automatengraphen an der Bewertung der Kanten durch die Zustandsübergangswahrscheinlichkeiten erkennbar (Abb. 11.23). Wenn der Zustandsübergang eindeutig ist wie im Beispiel die Übergänge  $2 \rightarrow 3$  und  $4 \rightarrow 5$ , so hat die zugehörige Übergangswahrscheinlichkeit den Wert eins.

Bei nichtdeterministischen Übergängen wichtet  $G(z' | z)$  die einzelnen Übergänge bezüglich der Häufigkeit ihres Auftretens, wobei entsprechend Gl. (11.43)

$$\sum_{z' \in \mathcal{Z}} G(z' | z) = 1, \quad z \in \mathcal{Z}$$

die Summe der Gewichte an allen von einem Knoten  $z$  ausgehenden Kanten gleich eins ist. Im abgebildeten Automaten kann also beispielsweise  $G(2 | 1) = 0,3$  und  $G(3 | 1) = 0,7$  gelten.

**Verhalten stochastischer Automaten.** Der Automat kann jede Zustandsfolge durchlaufen, die in einem Zustand  $z_0$  beginnt, den der Automat mit einer nicht verschwindenden Wahrscheinlichkeit annimmt ( $p_0(z_0) \neq 0$ ) und bei der aufeinanderfolgende Zustände  $z', z$  eine nicht verschwindende Übergangswahrscheinlichkeit haben:

$$G(z' | z) = \text{Prob}(Z(k+1) = z' | Z(k) = z) \neq 0.$$

Die Menge aller derartiger Zustandsfolgen ist

$$\mathcal{B} = \{(z(0), z(1), z(2), \dots) \mid p_0(z(0)) \neq 0, \\ G(z(k+1) | z(k)) \neq 0, \quad k = 0, 1, 2, \dots\}. \quad (11.45)$$

Diese Gleichung erweitert die für nichtdeterministische Automaten geltende Gl. (11.21) auf stochastische Automaten.

Außer der Menge der möglichen Trajektorien kann für den stochastischen Automaten angegeben werden, mit welcher Wahrscheinlichkeit sich der Automat zur Zeit  $k$  in den einzelnen Zuständen der Menge  $\mathcal{Z}(k)$  befindet. Das Verhalten des stochastischen Automaten ist deshalb durch die Folge der Wahrscheinlichkeiten  $p_z(k) = \text{Prob}(Z(k) = z)$ , ( $k = 0, 1, 2, \dots$ ) beschrieben, die man aus der Beziehung

$$\text{Prob}(Z(k+1) = z') = \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' | Z(k) = z) \cdot \text{Prob}(Z(k) = z)$$

erhält (vgl. Satz der vollständigen Wahrscheinlichkeit). Mit den oben eingeführten Bezeichnungen kann man diese Gleichung kürzer als

Zustandsraumdarstellung autonomer stochastischer Automaten:

$$\mathcal{S} : \begin{cases} \text{Prob}(Z(k+1) = z') = \sum_{z \in \mathcal{Z}} G(z' | z) \cdot \text{Prob}(Z(k) = z) \\ \text{Prob}(Z(0) = z) = p_0(z) \end{cases} \quad (11.46)$$

schreiben. Diese Beziehung wird als *Chapman-Kolmogorov-Gleichung*<sup>2</sup> bezeichnet. Sie ist die Zustandsraumdarstellung stochastischer Prozesse, denn mit ihr kann aus der für alle  $z \in \mathcal{Z}$  bekannten Aufenthaltswahrscheinlichkeit  $\text{Prob}(Z(k) = z)$  für den Zeitpunkt  $k$  die Aufenthaltswahrscheinlichkeit  $\text{Prob}(Z(k+1) = z)$  für alle  $z \in \mathcal{Z}$  im Nachfolgezeitpunkt  $k+1$

<sup>2</sup> ANDREJ N. KOLMOGOROV (1903 – 1987), russischer Mathematiker, schuf die axiomatische Begründung der Wahrscheinlichkeitstheorie

bestimmt werden. Für den Anfangszeitpunkt  $k = 0$  ist die Wahrscheinlichkeitsverteilung entsprechend Gl. (11.44) vorgegeben.

**Matrixdarstellung stochastischer Automaten.** In den bereits für deterministische und nicht-deterministische Automaten verwendeten Vektor  $\mathbf{p}(k)$  werden jetzt die Zustandswahrscheinlichkeiten  $\text{Prob}(Z(k) = z)$  eingetragen, so dass für das  $i$ -te Element die Beziehung

$$p_i(k) = \text{Prob}(Z(k) = i) \quad (11.47)$$

gilt. In der Matrix  $\mathbf{G}$  stehen die in Gl. (11.42) definierten Größen  $g_{z'z}$ . Wegen Gl. (11.43) ist  $\mathbf{G}$  eine stochastische Matrix, d. h., alle Spaltensummen sind gleich eins.

Der autonome stochastische Automat ist folglich durch die Gleichung

$$\mathcal{S}: \mathbf{p}(k+1) = \mathbf{G}\mathbf{p}(k), \quad \mathbf{p}(0) = \mathbf{p}_0 \quad (11.48)$$

beschrieben, die dasselbe wie Gl. (11.46) ausdrückt.  $\mathbf{p}_0$  ist die vektorielle Darstellung der Wahrscheinlichkeitsverteilung  $p_0(z)$  des Anfangszustands.

#### Beispiel 11.6 Würfelspiel mit vielen Personen

Um die Bedeutung der Übergangswahrscheinlichkeit  $g_{z'z}$  zu veranschaulichen, soll ein Würfelspiel mit einer großen Zahl von Personen durchgeführt werden. Die Anzahl der gewürfelten Sechsen beschreibt den „Zustand“ jeder Person. Die Zustände sind deshalb mit null beginnend durchnummeriert

$$\mathcal{Z} = \{0, 1, \dots, N\},$$

wobei  $N$  in Abhängigkeit von der Spieldauer hinreichend groß gewählt wird. Der Prozentsatz der Personen, die sich zum Zeitpunkt  $k$  im Zustand  $z$  befinden, gibt (näherungsweise) die Wahrscheinlichkeit  $\text{Prob}(Z(k) = z)$  an. Offensichtlich gilt  $\text{Prob}(Z(0) = 0) = 1$ , also

$$p_0(z) = \begin{cases} 1 & \text{für } z = 0 \\ 0 & \text{sonst.} \end{cases}$$

Es soll ein stochastischer Automat aufgestellt werden, der das Würfelspiel beschreibt, wobei jede Person in jedem Zeittakt einmal würfelt. In jedem Zeitschritt geht durchschnittlich  $\frac{1}{6}$  der Personen in ihren Nachfolgezustand über. Deshalb gilt

$$\text{Prob}(Z(k+1) = i+1 \mid Z(k) = i) = \frac{1}{6}.$$

Da diese Übergangswahrscheinlichkeit von der Zeit  $k$  unabhängig ist, kann man auch

$$g_{i+1,i} = \frac{1}{6}$$

schreiben. Auf Grund der Spielregeln gilt ferner

$$g_{ij} = 0 \quad \text{für } j \neq i-1, j \neq i.$$

Aus Gl. (11.43) erhält man

$$g_{ii} = 1 - g_{i+1,i} = \frac{5}{6}.$$



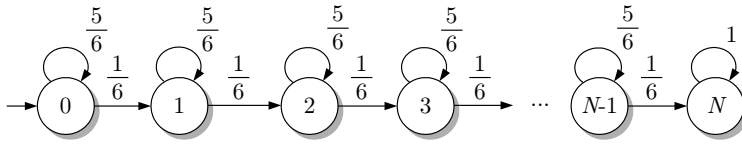


Abb. 11.24: Stochastischer Automat, der das Würfelspiel beschreibt

Personen, die  $N$  Sechsen gewürfelt haben, sollen nicht mehr mitspielen. Diese Personen verbleiben im Zustand  $N$ , so dass  $g_{NN} = 1$  gilt.

Der Automatengraph ist in Abb. 11.24 zu sehen. Auf Grund der Spielregeln sind nur Zustandsübergänge  $i \rightarrow i + 1$  und  $i \rightarrow i$  möglich.

Wichtig ist, dass die Übergangswahrscheinlichkeit nicht die Absolutzahl der gewürfelten Sechsen, sondern den Anteil der Personen im Zustand  $z$  beschreibt, der eine Sechs würfelt. Die Aufenthaltswahrscheinlichkeit  $p_i(k) = \text{Prob}(Z(k) = i)$  des Automaten im Zustand  $i$  zur Zeit  $k$  gibt für alle Spieler die Wahrscheinlichkeit an, nach  $k$  Würfeln  $i$  Sechsen gewürfelt zu haben.

Als Matrixdarstellung des Würfelspiels erhält man

$$\text{Würfelspiel: } \begin{pmatrix} p_0(k+1) \\ p_1(k+1) \\ p_2(k+1) \\ p_3(k+1) \\ \vdots \\ p_N(k+1) \end{pmatrix} = \begin{pmatrix} \frac{5}{6} & 0 & 0 & \dots & 0 \\ \frac{1}{6} & \frac{5}{6} & 0 & \dots & 0 \\ 0 & \frac{1}{6} & \frac{5}{6} & \dots & 0 \\ 0 & 0 & \frac{1}{6} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \begin{pmatrix} p_0(k) \\ p_1(k) \\ p_2(k) \\ p_3(k) \\ \vdots \\ p_N(k) \end{pmatrix}, \quad \begin{pmatrix} p_0(0) \\ p_1(0) \\ p_2(0) \\ p_3(0) \\ \vdots \\ p_N(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Die Nummerierung der Elemente des Vektors  $\mathbf{p}(k)$  beginnt wie die Nummerierung der Zustände bei null.

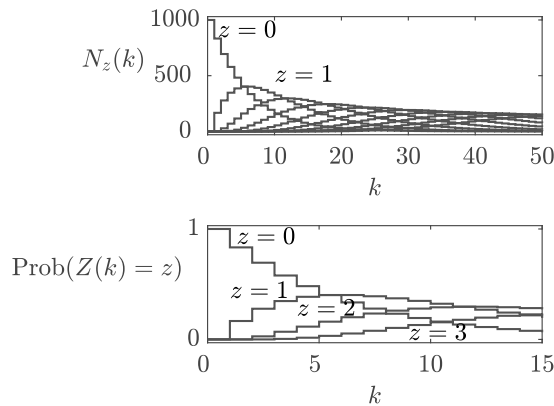


Abb. 11.25: Verhalten des Würfelspiels

Die Trajektorie des Automaten sagt den mittleren Spielverlauf voraus. Im unteren Teil von Abb. 11.25 ist der Verlauf von  $p_z(k) = \text{Prob}(Z(k) = z)$  für die ersten 15 Würfe ( $0 \leq k \leq 15$ ) und die Zustände  $z = 0, 1, 2, 3$  dargestellt. Erwartungsgemäß nimmt die Wahrscheinlichkeit  $p_0(k) = \text{Prob}(Z(k) =$

0), dass eine Person noch keine Sechs gewürfelt hat, vom Anfangswert  $\text{Prob}(Z(0) = 0) = 1$  nach dem ersten Wurf auf den Wert  $\text{Prob}(Z(1) = 0) = \frac{5}{6}$  ab, weil durchschnittlich  $\frac{1}{6}$  aller Spieler eine Sechs würfelt. Gleichzeitig steigt die Wahrscheinlichkeit, genau eine Sechs gewürfelt zu haben, von  $\text{Prob}(Z(0) = 1) = 0$  auf  $\text{Prob}(Z(1) = 1) = \frac{1}{6}$ .

Diese Vorgänge wiederholen sich im zweiten Wurf, wobei wiederum  $\frac{1}{6}$  der Spieler vom Zustand 0 in den Zustand 1 wechseln. Gleichzeitig erreichen aber erste Spieler den Zustand 2. Im weiteren Verlauf des Spiels nimmt die Zahl der Spieler im Zustand 1 durch Spieler zu, die die erste Sechs würfeln, und durch Spieler ab, die die zweite Sechs würfeln. Dabei hat die Wahrscheinlichkeit, genau eine Sechs geworfen zu haben, nach  $k = 6$  Würfeln ihr Maximum.

Diese Wahrscheinlichkeiten lassen sich mit Gl. (11.39) (näherungsweise) in die Anzahl von Personen umrechnen, die  $z$  Sechsen gewürfelt haben. Sei  $N = 1000$  die Anzahl der Mitspieler und  $N_z(k)$  die Anzahl der Personen, die nach  $k$  Würfeln  $z$  Sechsen geworfen haben. Wegen Gl. (11.40) gilt (näherungsweise)

$$N_z(k) = N \cdot \text{Prob}(Z(k) = z).$$

Das heißt, dass der im unteren Teil der Abbildung dargestellte Verlauf von  $\text{Prob}(Z(k) = z)$  durch Multiplikation mit  $N = 1000$  in  $N_z(k)$  umgerechnet werden kann, was für 50 Würfe im oberen Teil der Abbildung aufgetragen ist.

Bei diesem Beispiel ist das Würfelspiel der stochastische Prozess, bei dem der Würfel als Ausdruck des Zufalls auch physisch vorhanden ist. Jedes Spiel hat einen anderen Ausgang. Die Zahl der Personen mit  $z$  Sechsen ändert sich entsprechend dem Ergebnis jedes Wurfes. Der stochastische Automat, der diesen stochastischen Prozess beschreibt, zeigt, wie sich der Spielverlauf im Mittel gestaltet. Mit ihm kann man vorhersagen, wie viele Spieler durchschnittlich nach  $k$  Schritten  $z$  Sechsen gewürfelt haben. Der stochastische Automat als Modell des Würfelspiels sagt jedoch nichts darüber aus, wie viel oder gar welche Spieler sich in einem konkreten Spiel im Zustand  $z$  befinden.

Der Zustand  $z$  ist trotz des Nichtdeterminismus des Spiels ein Zustand im systemtheoretischen Sinne, denn wenn man ihn kennt, kann man die Einzelergebnisse aller bisherigen Würfe vergessen. Der nach der Zeit  $k$  liegende Spielverlauf wird nur durch den Zustand  $Z(k)$  bzw. die Wahrscheinlichkeitsverteilung  $\text{Prob}(Z(k) = z)$ , ( $z \in \mathcal{Z}$ ) und die zukünftigen Würfe bestimmt.  $\square$

### Beispiel 11.7 Beschreibung der Stanze unter Berücksichtigung der Fehlerwahrscheinlichkeit

Das Modell einer Stanze aus Beispiel 11.3 soll jetzt so erweitert werden, dass mit ihm die Ausfallwahrscheinlichkeit der Stanze berechnet werden kann. Dazu wird angenommen, dass von einer Blechrolle 1000 Teile gestanzt werden können und dass der Fehler beim Ablegen des Werkstücks durchschnittlich alle 2000 Arbeitszyklen einmal auftritt. Dementsprechend gilt

$$\begin{aligned} g_{87} &= 0,001 \\ g_{97} &= 0,0005. \end{aligned}$$

Die Matrix  $\mathbf{G}$  entsteht aus der in Gl. (11.28) angegebenen, indem man die Elemente  $g_{87} = 1$  und  $g_{97} = 1$  durch  $g_{87} = 0,001$  bzw.  $g_{97} = 0,0005$  ersetzt. Auf Grund der Bedingung (11.43) muss ferner das Element  $g_{17}$  den Wert

$$g_{17} = 1 - g_{87} - g_{97} = 0,9985$$

erhalten und die Diagonalelemente  $g_{88}$  und  $g_{99}$  gleich eins gesetzt werden. Der letzte Schritt entspricht der im Beispiel 11.3 behandelten Erweiterung, durch die der Automat unaufhörlich weiterschalten kann. Der Anfangszustand ist derselbe wie der in Gl. (11.27) rechts angegebene.

Mit dem stochastischen Automaten kann man beispielsweise ausrechnen, in welchem Zustand sich die Stanze nach 1500 Arbeitsschritten befindet. Man erhält nach 1500-maliger Anwendung der Rekursionsbeziehung (11.48) das Ergebnis

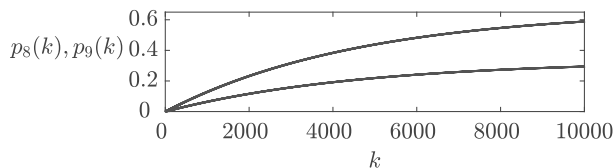
$$\mathbf{p}(1500) = \mathbf{G}^{1500} \mathbf{p}_0 = (0 \ 0 \ 0,7252 \ 0 \ 0 \ 0 \ 0 \ 0,1832 \ 0,0916)^T,$$

das sich folgendermaßen interpretieren lässt:

Die Stanze befindet sich im Zustand...	... mit der Wahrscheinlichkeit
3 („Das Blech ist abgetrennt.“)	$p_3(1500) = 0,7252$
8 („Die Stanze steht in Ausgangsposition, aber Blechrolle ist aufgebraucht.“)	$p_8(1500) = 0,1832$
9 („Das Werkstück ist nicht ordnungsgemäß abgelegt.“)	$p_9(1500) = 0,0916$

Würde man die aufgetretenen Fehler nicht beheben, so stünde die Stanze zur Zeit  $k = 1500$  mit der Wahrscheinlichkeit  $p_8(1500) + p_9(1500) = 0,1832 + 0,0916 = 0,2748$  still, weil im Zeitraum  $0 \leq k \leq 1500$  einer der beiden Fehler aufgetreten ist. Diese Aussage ist die bestmögliche, die man mit den im Modell enthaltenen Informationen machen kann. Da das Modell den Blechverbrauch nicht im Detail nachrechnet, kann insbesondere nur eine Wahrscheinlichkeitsaussage darüber abgegeben werden, ob der Zustand 8 eingetreten ist. Im Modell wird angenommen, dass zur Zeit  $k = 0$  eine beliebig „gefüllte“ Blechrolle vorhanden ist. Deshalb kann die Stanze mit der Wahrscheinlichkeit 0,001 bereits nach dem ersten Durchlauf aller Arbeitsschritte stehen bleiben.

Die Maschine muss nach spätestens 1000 Stanzzyklen, also 7000 Zustandsänderungen, zum Stillstand kommen, weil die Blechrolle aufgebraucht ist. Da anstelle dieser Information nur die „schwächere“ Wahrscheinlichkeitsaussage  $p_{87} = 0,001$  im stochastischen Automaten enthalten ist, erhält man für die Wahrscheinlichkeit des Stillstandes den relativ kleinen Wert  $p_8(7000) = 0,518$ .



**Abb. 11.26:** Fehlerwahrscheinlichkeit der Stanze

Abbildung 11.26 zeigt die über die Zeit zunehmende Fehlerwahrscheinlichkeit, die durch die Zustandsaufenthaltswahrscheinlichkeiten  $p_8(k)$  und  $p_9(k)$  ausgedrückt wird. Nach 10000 Arbeitsschritten beträgt die Wahrscheinlichkeit für den Stillstand der Stanze auf Grund des Fehlers „Die Blechrolle ist aufgebraucht“ 59%, für den Stillstand auf Grund des Fehlers „Das Werkstück nicht ordnungsgemäß abgelegt“ 28,5%. Die monotone Zunahme der Fehlerwahrscheinlichkeiten ist dadurch begründet, dass der stochastische Automat die Reparatur der Stanze nicht mit beschreibt.  $\square$

**Aufgabe 11.10** Beschreibung der Zuverlässigkeit einer Werkzeugmaschine

Eine Werkzeugmaschine, die Werkstücke fräst, transportiert ein Werkstück zunächst in die Werkstückaufnahme, klemmt es dort fest, bearbeitet es mit dem Fräser und transportiert das Werkstück dann in eine Ablage. Die Zuverlässigkeit der Maschine hängt vom reibungslosen Ablauf dieser Vorgänge ab. Überlegen Sie sich, welche Fehler zum Stillstand der Maschine führen können und beschreiben Sie den gewünschten Ablauf sowie den Übergang in Fehlerzustände durch einen stochastischen Automaten. Welche Eigenschaften der Maschine werden durch die Zustandsübergangswahrscheinlichkeiten ausgedrückt?  $\square$

### 11.4.3 Stochastische Automaten mit Eingang und Ausgang

Der stochastische Automat mit Eingang  $v(k)$  und Ausgang  $w(k)$  (stochastischer E/A-Automat)

$$\text{Stochastischer E/A-Automat: } \mathcal{S} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, L, p_0(z)) \quad (11.49)$$

wird durch die Verhaltensrelation  $L$  beschrieben

$$L : \mathcal{Z} \times \mathcal{W} \times \mathcal{Z} \times \mathcal{V} \longrightarrow [0, 1],$$

die die bedingte Wahrscheinlichkeit dafür angibt, dass der Automat zum Zeitpunkt  $k$  aus dem Zustand  $z$  in den Zustand  $z'$  übergeht, wenn er die Eingabe  $v$  erhält und dabei die Ausgabe  $w$  erzeugt:

$$L(z', w, z, v) = \text{Prob}(Z(k+1) = z', W(k) = w \mid Z(k) = z, V(k) = v). \quad (11.50)$$

Um den Charakter der bedingten Wahrscheinlichkeit hervorzuheben, schreibt man an Stelle von  $L(z', w, z, v)$  häufig  $L(z', w \mid z, v)$ . Die Verhaltensrelation hat die Eigenschaften

$$0 \leq L(z', w \mid z, v) \leq 1, \quad \forall z', z \in \mathcal{Z}, v \in \mathcal{V}, w \in \mathcal{W} \quad (11.51)$$

$$\sum_{z' \in \mathcal{Z}} \sum_{w \in \mathcal{W}} L(z', w \mid z, v) = 1, \quad \forall (z, v) \in \mathcal{Z} \times \mathcal{V}, \quad (11.52)$$

denn sie ist eine bedingte Wahrscheinlichkeit. Die beiden Randverteilungen

$$G(z' \mid z, v) = \sum_{w \in \mathcal{W}} L(z', w \mid z, v) \quad (11.53)$$

$$H(w \mid z, v) = \sum_{z' \in \mathcal{Z}} L(z', w \mid z, v) \quad (11.54)$$

ersetzen die Funktionen  $G$  und  $H$  des deterministischen Automaten bzw. des nichtdeterministischen Mealy-Automaten. Für sie gelten die Beziehungen

$$G(z' \mid z, v) = \text{Prob}(Z(k+1) = z' \mid Z(k) = z, V(k) = v) \quad (11.55)$$

$$H(w \mid z, v) = \text{Prob}(W(k) = w \mid Z(k) = z, V(k) = v) \quad (11.56)$$

und

$$\sum_{z' \in \mathcal{Z}} G(z' \mid z, v) = 1 \quad (11.57)$$

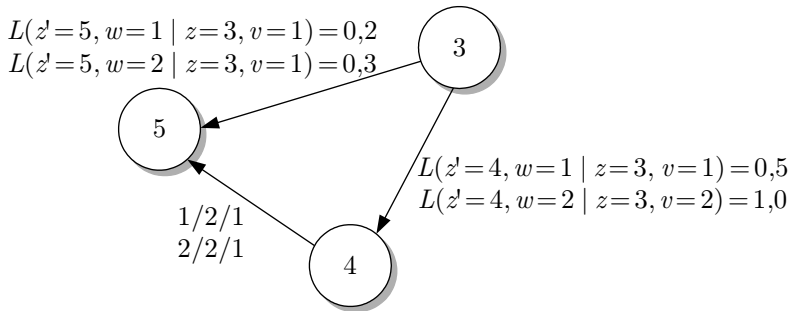
$$\sum_{w \in \mathcal{W}} H(w \mid z, v) = 1. \quad (11.58)$$

Wie für den nichtdeterministischen Mealy-Automaten erläutert wurde, bietet die Beschreibung des Automatenverhaltens durch die Verhaltensrelation  $L$  größere Flexibilität als die Verwendung der beiden Verteilungen  $G$  und  $H$ . Das ist für den stochastischen Automaten durch die

wahrscheinlichkeitstheoretische Tatsache begründet, dass man aus einer Verteilung zwar die Randverteilungen (11.53) und (11.54) berechnen, aber nicht umgekehrt aus den Randverteilungen  $G$  und  $H$  die Verteilung  $L$  zurückgewinnen kann. Nur für den Spezialfall der *stochastischen Mealy-Automaten*, für die die Beziehung

$$L(z', w | z, v) = H(w | z, v) \cdot G(z' | z, v) \quad (11.59)$$

gilt, sind beide Darstellungsformen äquivalent.



**Abb. 11.27:** Teil des Automatengraphen eines stochastischen Automaten mit Eingängen und Ausgängen

Im Automatengraphen werden die Kanten jetzt, wie in Abb. 11.27 angegeben, durch die Verhaltensrelation gewichtet. Zur Vereinfachung der Darstellung wurde jede Kante nur einmal gezeichnet und gegebenenfalls mit mehreren Wichtungen versehen. Bei dem angegebenen Beispiel ist der Zustandsübergang  $3 \rightarrow 4$  eindeutig bestimmt, wenn der Automat im Zustand  $z = 3$  die Eingabe  $v = 2$  erhält. Bei der Eingabe  $v = 1$  gibt es mehrere mögliche Übergänge, wobei sich die Wahrscheinlichkeit des Übergangs  $3 \rightarrow 5$  in Abhängigkeit von der dabei erzeugten Ausgabe weiter unterscheidet. Für den Zustandsübergang  $3 \rightarrow 5$  bei gleichzeitiger Erzeugung der Ausgabe  $w = 1$  gibt die Verhaltensrelation die Wahrscheinlichkeit 0,2 an, während derselbe Übergang mit der Ausgabe  $w = 2$  mit der Wahrscheinlichkeit 0,3 auftritt.

Bei größeren Automaten wird die Verhaltensrelation nicht mehr explizit an die Kanten geschrieben, sondern nur das aus Eingang  $v$ , Ausgang  $w$  und Übergangswahrscheinlichkeit bestehende Tripel, wie es in Abb. 11.27 für den Zustandsübergang  $4 \rightarrow 5$  gezeigt wird. Die Tripel bedeuten

$$\begin{aligned} L(5, 2 | 4, 1) &= 1 \\ L(5, 2 | 4, 2) &= 1. \end{aligned}$$

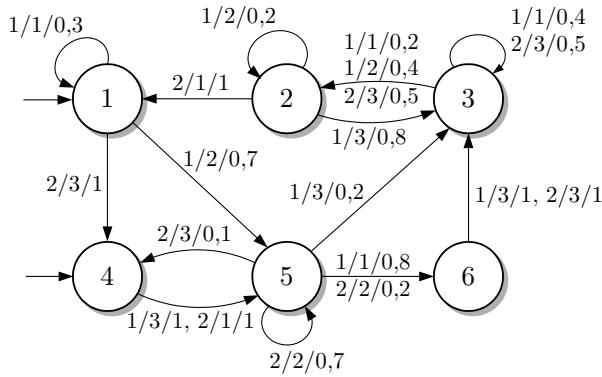


Abb. 11.28: Stochastischer Automat

**Beispiel 11.8** Stochastischer E/A-Automat

Abbildung 11.28 zeigt den um Übergangswahrscheinlichkeiten erweiterten Automatengraphen aus Abb. 11.19. Offensichtlich gilt die Beziehung (11.52) für  $z = 3$  und  $v = 1$  bzw.  $v = 2$ , denn für die Summe erhält man, wenn man nur die von null verschiedenen Summanden aufschreibt, die folgenden Beziehungen

$$\begin{aligned}
 \sum_{z' \in \mathcal{Z}} \sum_{w \in \mathcal{W}} L(z', w | 3, 1) &= L(2, 1 | 3, 1) + L(2, 2 | 3, 1) + L(3, 1 | 3, 1) \\
 &= 0,2 + 0,4 + 0,4 = 1 \\
 \sum_{z' \in \mathcal{Z}} \sum_{w \in \mathcal{W}} L(z', w | 3, 2) &= L(2, 3 | 3, 2) + L(3, 3 | 3, 2) \\
 &= 0,5 + 0,5 = 1.
 \end{aligned}$$

Wenn man sich nur dafür interessiert, in welchen Nachfolgezustand der Automat in Abhängigkeit von der Eingabe übergeht, aber dabei die Ausgabe ignoriert, so berechnet man die Zustandsübergangswahrscheinlichkeit entsprechend Gl. (11.53):

$$\begin{aligned}
 G(2 | 3, 1) &= \sum_{w \in \mathcal{W}} L(2, w | 3, 1) = 0,2 + 0,4 = 0,6 \\
 G(3 | 3, 1) &= \sum_{w \in \mathcal{W}} L(3, w | 3, 1) = 0,4 \\
 G(2 | 3, 2) &= \sum_{w \in \mathcal{W}} L(2, w | 3, 2) = 0,5 \\
 G(3 | 3, 2) &= \sum_{w \in \mathcal{W}} L(3, w | 3, 2) = 0,5.
 \end{aligned}$$

Die erste Zeile besagt, dass der Automat vom Zustand  $z = 3$  bei der Eingabe  $v = 1$  mit einer Wahrscheinlichkeit von 0,5 in den Zustand  $z' = 4$  übergeht. Es wird nichts über die dabei erzeugte Ausgabe ausgesagt.  $\square$

**Matrixdarstellung stochastischer E/A-Automaten.** Eine Erweiterung der Matrixdarstellung (11.38) auf stochastische Automaten ist nur für stochastische Mealy-Automaten möglich. Für diese gilt

$$\begin{aligned} \mathbf{p}(k+1) &= \mathbf{G}(v(k)) \cdot \mathbf{p}(k) \\ \mathbf{p}_w(k) &= \mathbf{H}(v(k)) \cdot \mathbf{p}(k), \end{aligned}$$

wobei die Vektoren  $\mathbf{p}(k)$  und  $\mathbf{p}_w(k)$  entsprechend Gl. (11.47) die Wahrscheinlichkeiten enthalten, mit denen zum Zeitpunkt  $k$  die einzelnen Zustände bzw. Ausgaben auftreten.

**Aufgabe 11.11\*** *Erweiterung des Modells einer Raumbelichtung*

Erweitern Sie das Modell aus Aufgabe 11.8 zum stochastischen Automaten. Welche zusätzlichen Informationen benötigen Sie dafür? Wie sieht die Matrixdarstellung aus?  $\square$

**Aufgabe 11.12\*** *Beschreibung eines Regensors durch einen stochastischen Automaten*

Für den im Beispiel 11.5 beschriebenen Regensor wurden folgende Übergangswahrscheinlichkeiten ermittelt, die sich auf die auf S. 368 eingeführten Abkürzungen für die Eingaben und Zustände beziehen:

$$\begin{aligned} \text{Prob}(Z(k+1) = 2 \mid Z(k) = 1, V(k) = 2) &= 0,95 \\ \text{Prob}(Z(k+1) = 1 \mid Z(k) = 1, V(k) = 1) &= 0,98 \\ \text{Prob}(Z(k+1) = 1 \mid Z(k) = 2, V(k) = 1) &= 0,90 \\ \text{Prob}(Z(k+1) = 2 \mid Z(k) = 2, V(k) = 2) &= 1. \end{aligned}$$

Was besagen diese Angaben? Erweitern Sie den nichtdeterministischen Automaten aus Abb. 11.20 auf S. 369 zu einem stochastischen Automaten.  $\square$

#### 11.4.4 Markoveigenschaft dynamischer Systeme

Bei allen bisher behandelten Modellformen kann der Zustand zum Zeitpunkt  $k+1$  aus alleiniger Kenntnis des Zustands zum Zeitpunkt  $k$  berechnet werden. Dieser Sachverhalt wird aus den rekursiven Beziehungen (11.8), (11.20) und (11.46) offensichtlich. Dabei wird beim deterministischen Automaten der eindeutige Nachfolgezustand berechnet, beim nichtdeterministischen Automaten die Menge der möglichen Nachfolgezustände und beim stochastischen Automaten die Wahrscheinlichkeit, mit der die möglichen Nachfolgezustände angenommen werden. Allen Beziehungen ist gemeinsam, dass das Ergebnis nur vom aktuellen Zustand, der Menge der möglichen aktuellen Zustände bzw. der Wahrscheinlichkeitsverteilung für den aktuellen Zustand ermittelt wird. Die weiter zurückliegenden Zustände  $z(k-1)$ ,  $z(k-2)$  usw. haben keinen Einfluss. Diese Eigenschaft dynamischer Systeme nennt man die *Markoveigenschaft*<sup>3</sup>.

<sup>3</sup> ANDREJ ANDREJEWITSCH MARKOV (1856 – 1922), russischer Mathematiker

Um die Bedeutung dieser Eigenschaft zu verstehen, muss man sich überlegen, dass der zum Zeitpunkt  $k + 1$  angenommene Zustand  $i$ . Allg. von allen vorherigen Zuständen abhängig sein kann, was durch die bedingte Wahrscheinlichkeit

$$\text{Prob}(Z(k+1) = z' \mid Z(k) = z, Z(k-1) = z_{k-1}, Z(k-2) = z_{k-2}, \dots, Z(0) = z_0)$$

zum Ausdruck gebracht wird, in der alle dem Zustand  $Z(k+1) = z'$  vorhergehenden Zustände im Bedingungsteil stehen. Die Markoveigenschaft besagt nun, dass diese bedingte Wahrscheinlichkeit nicht von den Zuständen  $z_{k-1}, z_{k-2}, \dots, z_0$  beeinflusst wird:

Markoveigenschaft ereignisdiskreter Systeme:

$$\text{Prob}(Z(k+1) = z' \mid Z(k) = z, \dots, Z(0) = z_0) = \text{Prob}(Z(k+1) = z' \mid Z(k) = z).$$

(11.60)

Ein System besitzt die Markoveigenschaft, wenn diese Beziehung für beliebige Zustandsfolgen  $z', z, z_{k-1}, \dots, z_0$  erfüllt ist, die mit positiver Wahrscheinlichkeit auftreten können:

$$\text{Prob}(Z(k+1) = z' \mid Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0) > 0.$$

Die Eigenschaft eines Prozesses, dass das Verhalten nach dem Zeitpunkt  $k$  nicht vom Verhalten vor dem Zeitpunkt  $k$  abhängig ist, wird häufig auch noch anders formuliert. Man sagt, dass die Vergangenheit und die Zukunft eines Prozesses bei bekannter Gegenwart voneinander unabhängig sind, und man sagt, dass ein Prozess, der die Markoveigenschaft besitzt, kein Gedächtnis hat bzw. ohne Nachwirkung ist. Diese Formulierungen muss man richtig interpretieren. Die Markovkette besitzt insofern ein Gedächtnis, als dass der Zustand  $Z(k+1)$  vom Zustand  $Z(k)$  beeinflusst wird, aber er speichert keine weiter zurückliegenden Zustände (vgl. Beispiel 11.9).

**Markoveigenschaft stochastischer Automaten.** Im Folgenden wird gezeigt, dass die Beziehung (11.46) für das Verhalten des stochastischen Automaten nur unter der Voraussetzung gilt, dass der stochastische Automat die Markoveigenschaft (11.60) besitzt. Da der stochastische Automat eine zufällige Folge

$$(Z(0), Z(1), \dots, Z(k-1), Z(k), Z(k+1))$$

erzeugt, muss man für die Berechnung der Wahrscheinlichkeit  $\text{Prob}(Z(k+1) = z')$  die Wahrscheinlichkeitsverteilung

$$\text{Prob}(Z(k+1) = z', Z(k) = z, Z(k-1) = z_{k-1}, Z(k-2) = z_{k-2}, \dots, Z(0) = z_0)$$

über alle  $z', z, z_{k-1}, \dots, z_0 \in \mathcal{Z}$  zu Grunde legen. Die gesuchte Aufenthaltswahrscheinlichkeit für den  $(k+1)$ -ten Zustand erhält man als Randverteilung

$$\text{Prob}(Z(k+1) = z') = \tag{11.61}$$

$$\sum_{z, z_{k-1}, \dots, z_0 \in \mathcal{Z}} \text{Prob}(Z(k+1) = z', Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0).$$



Diese Beziehung kann man in

$$\begin{aligned} \text{Prob}(Z(k+1) = z') = \\ \sum_{z, z_{k-1}, \dots, z_0 \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' \mid Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0) \cdot \\ \cdot \text{Prob}(Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0) \end{aligned}$$

umformen. Nur wenn der betrachtete Prozess die Markoveigenschaft (11.60) besitzt, kann man die nächsten beiden Umformungsschritte durchführen

$$\begin{aligned} \text{Prob}(Z(k+1) = z') &= \sum_{z, z_{k-1}, \dots, z_0 \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' \mid Z(k) = z) \cdot \\ &\cdot \text{Prob}(Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0) \\ &= \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' \mid Z(k) = z) \cdot \\ &\cdot \sum_{z_{k-1}, \dots, z_1, z_0 \in \mathcal{Z}} \text{Prob}(Z(k) = z, Z(k-1) = z_{k-1}, \dots, Z(0) = z_0) \\ &= \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' \mid Z(k) = z) \cdot \text{Prob}(Z(k) = z), \end{aligned}$$

die auf die Beziehung (11.46) führen.

Das heißt, man kann die Wahrscheinlichkeitsverteilungen  $\text{Prob}(Z(k) = z)$  für steigende Zeit  $k$  nur dann mit der Chapman-Kolmogorov-Gleichung (11.46) berechnen, wenn das betrachtete System die Markoveigenschaft besitzt. Für stochastische Automaten wird diese Eigenschaft als a-priori gültig angenommen. Ob ein stochastischer Automat eine geeignete Modellform für ein ereignisdiskretes System ist, hängt folglich von der Frage ab, ob das zu beschreibende System tatsächlich die Markoveigenschaft besitzt.

**Bedeutung der Markoveigenschaft für die Modellbildung.** Die wichtigste Konsequenz der Markoveigenschaft für Anwendungen ist die, dass zur Aufstellung des Modells nur die in *einem* Schritt erfolgenden Zustandsübergänge betrachtet werden müssen und mit dem Modell dann beliebig lange Ketten von Zustandsübergängen berechnet werden können. Die in den Modellen (11.8), (11.20) und (11.46) enthaltenen Größen  $G$  bzw.  $L$  beschreiben gerade solche Einschrittübergänge.

Auf Grund der Markoveigenschaft gibt es für nichtdeterministische Systeme eine Analogie zur Zustandsraumdarstellung deterministischer Systeme

$$\begin{aligned} x(k+1) &= g(x(k)) && \text{(zeitdiskrete kontinuierliche Systeme)} \\ z(k+1) &= G(z(k)), && \text{(deterministische ereignisdiskrete Systeme)} \end{aligned}$$

die für autonome Systeme die Form

$$z(k+1) \in G(z(k)) \quad \text{(nichtdeterministische ereignisdiskrete Systeme)}$$

hat. Es unterscheiden sich lediglich die Darstellungen des Systemzustands, der bei deterministischen kontinuierlichen Systemen ein Vektor  $x(k) \in \mathbb{R}^n$  und bei nichtdeterministischen Systemen ein Element  $z(k)$  einer endlichen oder abzählbar unendlichen Menge  $\mathcal{Z}$  ist. Bei beiden Systemen kann die Trajektorie rekursiv aus dem Anfangszustand berechnet werden.

### Beispiel 11.9 Roulette und andere Glücksspiele

Viele Glücksspiele beruhen auf der Annahme, dass der Ausgang eines Spiels vollkommen unabhängig vom Ausgang des vorhergehenden Spiels ist. Dies trifft auf einen Würfel genauso zu wie auf Roulette. Diese stochastische Unabhängigkeit wird durch die Beziehung

$$\text{Prob}(Z(k+1) = z') = \text{Prob}(Z(k+1) = z' \mid Z(k) = z), \quad z, z' \in \mathcal{Z}$$

zum Ausdruck gebracht. Wenn man sich das Ergebnis des  $k$ -ten Spiels merkt, so kann man den Ausgang des  $(k+1)$ -ten Spiels nicht besser vorhersagen als wenn man diese Information nicht besitzt. Das Roulettespiel besitzt also „noch nicht einmal“ die Markoveigenschaft.

Wenn man dennoch derartige Spiele als Markovkette auffasst und sie durch einen autonomen stochastischen Automaten darstellt, dessen Zustand den Ausgang eines Spiels repräsentiert, so gilt für die Zustandsübergangswahrscheinlichkeit die Beziehung

$$\text{Prob}(Z(k+1) = z' \mid Z(k) = z) = \frac{1}{n},$$

wobei  $n$  die Zahl der möglichen Spielergebnisse beschreibt, also  $n = 6$  beim Würfel bzw.  $n = 37$  beim Roulette. Im Automatengraphen sind alle Zustände mit allen anderen über Kanten mit der Bewertung  $\frac{1}{n}$  verbunden. Da man zum Zeitpunkt  $k = 0$  mit einer Gleichverteilung über alle Zustände beginnen muss, weil man auch das erste Spielergebnis nicht kennt, erzeugt der Automat gleichverteilte Zustandswahrscheinlichkeiten

$$\text{Prob}(Z(k) = i) = \frac{1}{n} \quad \text{für } i = 1, 2, \dots, n, \quad k = 0, 1, \dots$$

Mit dem Modell kann man nichts über den Spielausgang aussagen.

**Diskussion.** Als Spieler hofft man, wenigstens aus dem letzten Spielausgang auf das nächste Spielergebnis schließen zu können. Man setzt beim Roulette auf die Farbe rot, wenn das letzte Ergebnis schwarz war, weil beide Farben gleichberechtigt sind und man deshalb einen Farbwechsel voraussieht. Wenn sich die Roulettekugel das letzte Ergebnis „merken“ würde, besäße sie die Markoveigenschaft. Nur dann wäre die beschriebene Spielstrategie sinnvoll.

In Casinos sieht man Spieler, die das Roulettespiel lange Zeit verfolgen, um Regelmäßigkeiten in den Ergebnisfolgen festzustellen und dementsprechend ihren Einsatz zu planen. Wenn dies möglich wäre, würde das Spiel ein längeres Gedächtnis besitzen und damit die Markoveigenschaft nicht erfüllen, denn dann würde das nächste Ergebnis sogar von der *Folge* der vorhergehenden Spielausgänge abhängen. Die Kugel müsste sich z. B. die vier letzten Ergebnisse merken. Man könnte das Spiel dann durch ein Markovmodell beschreiben, indem man die Ergebnisse der letzten vier Läufe in einem Vektor zusammenfasst und diesen Vektor als Zustand des Spiels interpretiert.

Spielautomaten, die eine bestimmte Gewinnchance garantieren müssen, verändern die Auftretenswahrscheinlichkeit der „ausgewürfelten“ Spielergebnisse in Abhängigkeit von der Folge der letzten Spielergebnisse. Sie besitzen die Markoveigenschaft in Bezug zu dem durch den Vektor der letzten Spielergebnisse ausgedrückten Zustand. Wer weiß, wie viele der letzten Spielergebnisse das zukünftige Spielergebnis beeinflussen und mit welchen Auftretenswahrscheinlichkeiten deshalb in den nächsten Spielzügen zu rechnen ist, kann das Spielergebnis besser vorhersagen und seinen Einsatz dementsprechend besser platzieren als jemand, der von der Unabhängigkeit der Spielzüge ausgeht.  $\square$

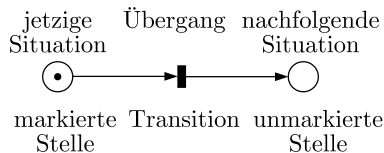
## 11.5 Petrinetze

### 11.5.1 Autonome Petrinetze

Automaten beschreiben diskrete Systeme durch die Zustände, die diese Systeme annehmen können, und die Übergänge, die zwischen diesen Zuständen möglich sind. Dabei nimmt der Automat definitionsgemäß zu jedem Zeitpunkt  $k$  genau einen Zustand an. Man kann sich das Verhalten von Automaten deshalb auch dadurch veranschaulichen, dass man den aktuellen Zustand im Automatengraphen mit einer Marke versieht, die unter dem Einfluss der Eingangsfolge entlang der Kanten von Zustand zu Zustand wandert. Zu jedem Zeitpunkt ist genau ein Zustand markiert.

Die Systembeschreibung durch Automaten wird komplex, weil diskrete Systeme typischerweise eine sehr große Zahl unterschiedlicher Zustände annehmen können. Wie das Beispiel 11.4 gezeigt hat, wird insbesondere bei nebenläufigen Prozessen die Anzahl der möglichen Zustände dadurch sehr groß, dass die unabhängig voneinander arbeitenden Systemteile in unterschiedlicher Reihenfolge ihren neuen Zustand erreichen und jede mögliche Reihenfolge durch einen eigenen Zustand des Automaten, der den Gesamtprozess beschreibt, dargestellt werden muss.

Die in diesem Abschnitt eingeführten Petrinetze<sup>4</sup> reduzieren die Modellkomplexität dadurch, dass parallel ablaufende Prozesse durch parallele Wege in der grafischen Darstellung repräsentiert werden. Der Zustand des Systems wird nicht mehr durch einen einzigen markierten Knoten, sondern durch eine Menge markierter Knoten beschrieben. Da im Graphen mehrere Marken auftreten können, die sich mit unterschiedlicher Geschwindigkeit bewegen, können vor allem parallele Prozesse einfacher durch Petrinetze beschrieben werden als durch Automaten.



**Abb. 11.29:** Elemente eines Petrinetzes

**Beschreibung der Netzelemente.** Im Folgenden werden zunächst autonome Petrinetze behandelt, also Petrinetze ohne Eingangsgrößen. Wie Abb. 11.29 zeigt, gibt es in ihnen zwei unterschiedliche Arten von Knoten:

- **Stellen** (Plätze), die durch Kreise dargestellt werden und den Systemzustand beschreiben,
- **Transitionen**, die durch schwarze Rechtecke dargestellt werden und Zustandsübergänge (Ereignisse) beschreiben.

Die Mengen dieser Netzelemente bezeichnet man mit  $\mathcal{P}$  und  $\mathcal{T}$ . Die markierten Stellen sind in den Abbildungen durch einen Punkt („Marke“) gekennzeichnet. Das Verhalten des Petrinetzes

<sup>4</sup> benannt nach CARL ADAM PETRI (1926 – 2010), der diese Netze 1962 einführte

wird durch die Bewegung der Marken beschrieben, die sich nach den im Folgenden erläuterten Regeln vollzieht.

Wie bereits erwähnt, beschreibt nicht jede Stelle für sich, sondern die Menge aller gleichzeitig markierten Stellen den aktuellen Systemzustand. Jeder Stelle kann deshalb nicht ein Zustand, sondern nur eine Situation zugeordnet werden, wobei mehrere, möglicherweise in unterschiedlicher Kombination auftretende Situationen den Systemzustand kennzeichnen. Gerichtete Kanten, die von einer Stelle über eine Transition zu einer anderen Stelle führen, können nach bestimmten Regeln eine Marke von einer Stelle zur nächsten führen. Dies bedeutet für das System, dass eine existierende Situation von einer nachfolgenden Situation abgelöst wird.

Ein Petrinetz wird durch einen bipartiten Graphen dargestellt, also einen Graphen, bei dem sich die beiden Knotenarten stets abwechseln, wenn man den Graphen entlang der gerichteten Kanten durchläuft. Vom Standpunkt der Transitionen unterscheidet man zwischen

- **Präkanten**, den Kanten von den Stellen  $p \in \mathcal{P}$  zu den Transitionen  $t \in \mathcal{T}$  und
- **Postkanten**, den Kanten von den Transitionen  $t \in \mathcal{T}$  zu den Stellen  $p \in \mathcal{P}$ .

Die Mengen dieser Kanten bezeichnet man mit  $\mathcal{P}rae$  bzw.  $\mathcal{P}ost$ . Formal kann man mit diesen Bezeichnungen ein Petrinetz als Tupel

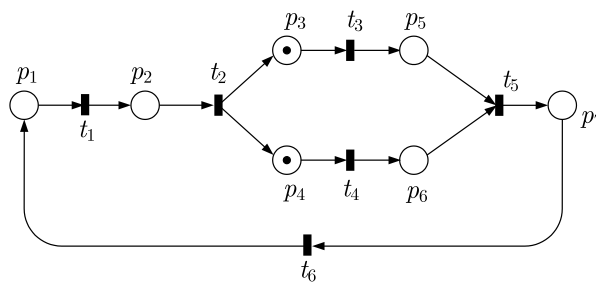
$$\text{Petrinetz: } \mathcal{PN} = (\mathcal{P}, \mathcal{T}, \mathcal{P}rae, \mathcal{P}ost, M_0)$$

schreiben, wobei die Beziehungen

$$\mathcal{P}rae \subseteq \mathcal{P} \times \mathcal{T}$$

$$\mathcal{P}ost \subseteq \mathcal{T} \times \mathcal{P}$$

gelten.  $M_0$  bezeichnet die Markierung des Netzes zur Zeit  $k = 0$ .

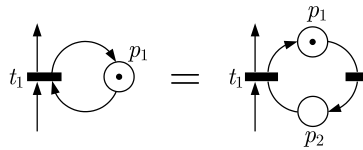


**Abb. 11.30:** Petrinetz mit zwei parallelen Prozessen

Abbildung 11.30 zeigt ein Petrinetz, bei dem die Stellen  $p_3$  und  $p_5$  bzw.  $p_4$  und  $p_6$  zusammen mit den sie verbindenden Transitionen zwei parallele Prozesse beschreiben. Das Netz zeigt, dass von Transitionen mehrere Kanten ausgehen oder mehrere Kanten in Transitionen münden können. Gleiches gilt für Stellen, auch wenn dies in der Abbildung nicht vorkommt.

Unter den *Prästellen* (Vorgängerstelle) einer Transition versteht man die Menge der Stellen, von denen aus eine gerichtete Kante zur betreffenden Transition führt. *Poststellen* (Nachfolgestellen) sind die Stellen, zu denen von der Transition gerichtete Kanten führen. Bezüglich der Stellen bezeichnet man die davor liegenden Transitionen, von denen Marken auf die Stelle fließen können, als *Eingangstransitionen* und die dahinter liegenden als *Ausgangstransitionen*.

Im Folgenden wird vorausgesetzt, dass das zu beschreibende Petrinetz keine Schlingen besitzt. Diese Voraussetzung kann stets dadurch erfüllt werden, dass man Schlingen durch Einführung einer neuen Transition und einer neuen Stelle eliminiert, wie es in Abb. 11.31 gezeigt ist. Petrinetze ohne Schlingen werden als reine Netze bezeichnet.



**Abb. 11.31:** Schlinge, die durch Einführung einer zusätzlichen Stelle und einer Transition ersetzt werden kann

**Markenfluss.** Das Verhalten des Petrinetzes wird durch einen Markenfluss beschrieben. Die Markierung  $M(k)$  kennzeichnet, welche Stellen zum Zeitpunkt  $k$  eine Marke besitzen. Sie wird meistens dadurch notiert, dass man die zur Zeit  $k$  markierten Stellen als Menge aufschreibt. Bei der später behandelten algebraischen Darstellung wird ein Vektor  $\mathbf{p}(k)$  eingeführt, dessen  $i$ -tes Element gleich eins ist, wenn die Stelle  $p_i$  zur Zeit  $k$  markiert ist, und andernfalls verschwindet.

Bei den hier verwendeten Petrinetzen darf jede Stelle höchstens eine Marke besitzen – die Markierung ist also binär. Wichtig ist, dass i. Allg. mehrere Stellen gleichzeitig markiert sind. Darin liegt gerade der Vorteil der Petrinetze gegenüber der Systembeschreibung durch Automaten. Die Markierung ändert sich mit der Zeit, was durch das zusätzliche Argument  $k$  gekennzeichnet wird:  $M(k)$ .

Das Netz hat die Initialmarkierung  $M_0$ , die in den Abbildungen gekennzeichnet ist. Die Marken wandern entlang der gerichteten Kanten von ihren Stellen über Transitionen zu Nachfolgestellen, was als Schalten oder Feuern von Transitionen bezeichnet wird und die Markierung  $M$  verändert. Dabei ist die folgende Schaltregel einzuhalten:

**Schaltregel:** Eine Transition ist *aktiviert*, wenn

- alle ihre Prästellen markiert und
- alle Poststellen nicht markiert sind.

Beim Schalten aktivierter Transitionen wird allen Prästellen die Marke entzogen und alle Poststellen werden markiert.

Eine aktivierte Transition kann schalten, muss es aber nicht. Die Menge der bei der Markierung  $M(k)$  aktivierten Transitionen wird mit  $\mathcal{T}_{\text{akt}}(M(k))$  bezeichnet. Enthält diese Menge

mehr als eine Transition, so verhält sich das Netz nichtdeterministisch. Welche der aktivierten Transitionen schaltet, kann nicht mit Hilfe des Petrinetzes bestimmt werden. So wie nichtdeterministische Automaten in mehr als einen Nachfolgezustand übergehen können, können Petrinetze von einer Markierung in mehr als eine Nachfolgemarkierung wechseln.

Man kann die Folgemarkierung  $M'$  von  $M$  mit dem Apostroph kennzeichnen, wie es bei Automaten für Nachfolgezustände schon getan wurde. Führt das Schalten der Transition  $t \in \mathcal{T}_{\text{akt}}(M)$  von der Markierung  $M$  auf die Nachfolgemarkierung  $M'$ , so schreibt man dies in der Form

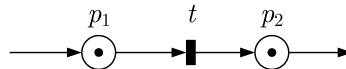
$$M \xrightarrow{t} M'.$$

Aus einer Anfangsmarkierung  $M(0) = M_0$  entsteht durch mehrfaches Schalten des Petrinetzes eine Markierungsfolge

$$M = (M(0), M(1), M(2), \dots),$$

die das dynamische Verhalten des Netzes beschreibt. Aufgrund des Nichtdeterminismus des Markenflusses erhält man für Petrinetze typischerweise nicht eine eindeutige Markierungsfolge, sondern eine Menge von Markierungsfolgen.

Die Anzahl der sich im Netz befindenden Marken kann sich beim Schalten ändern. Wenn beispielsweise die Stelle  $p_2$  in Abb. 11.30 markiert und die Stellen  $p_3$  und  $p_4$  nicht markiert sind, so ist die Transition  $t_2$  aktiviert. Das Feuern dieser Transition entfernt die Marke von der Stelle  $p_2$  und führt zwei neue Marken ein, je eine für die Stellen  $p_3$  und  $p_4$ , wodurch der in Abb. 11.30 gezeigte Systemzustand entsteht. Nach dem Schalten der Transition  $t_2$  hat das Netz also eine Marke mehr als vorher. Wenn andererseits die Stellen  $p_5$  und  $p_6$  markiert sind und die Stelle  $p_7$  nicht und deshalb die Transition  $t_5$  aktiviert ist, so verschwinden beim Schalten der Transition  $t_5$  die beiden Marken von den Plätzen  $p_5$  und  $p_6$  und erscheinen als eine Marke in der Stelle  $p_7$ , wodurch das Netz eine Marke verliert.



**Abb. 11.32:** Kontakt

Die angegebene Schaltregel berücksichtigt auch die Situation, dass sowohl die Prästellen als auch die Poststellen markiert sind (Abb. 11.32). Diese Situation nennt man einen *Kontakt*. Die Schaltregel besagt, dass in dieser Situation die Transition  $t$  nicht aktiviert ist. Die Markierung der Poststellen blockiert also den Fluss der nachfolgenden Marken.

Als Beispiel für den Markenfluss wird das Petrinetz in Abb. 11.30 für eine Initialmarkierung untersucht, bei der nur die Stelle  $p_1$  markiert ist. Durch Anwendung der angegebenen Schaltregel kann man dann folgende Markierungsfolgen erhalten, bei denen für die ersten sieben Zeitpunkte die jeweils markierten Stellen aufgeführt sind:

$$p_1, \quad p_2, \quad \begin{pmatrix} p_3 \\ p_4 \end{pmatrix}, \quad \begin{pmatrix} p_3 \\ p_6 \end{pmatrix}, \quad \begin{pmatrix} p_5 \\ p_6 \end{pmatrix}, \quad p_7, \quad p_1, \dots$$

$$\begin{aligned}
 & p_1, \quad p_2, \quad \begin{pmatrix} p_3 \\ p_4 \end{pmatrix}, \quad \begin{pmatrix} p_5 \\ p_4 \end{pmatrix}, \quad \begin{pmatrix} p_5 \\ p_6 \end{pmatrix}, \quad p_7, \quad p_1, \dots \\
 & p_1, \quad p_2, \quad \begin{pmatrix} p_3 \\ p_4 \end{pmatrix}, \quad \begin{pmatrix} p_5 \\ p_6 \end{pmatrix}, \quad p_7, \quad p_1, \quad p_2, \dots
 \end{aligned} \tag{11.62}$$

In der letzten Markierungsfolge schalten die Transitionen  $t_3$  und  $t_4$  gleichzeitig, während die anderen beiden Folgen durch das einzelne Feuern dieser beiden Transitionen entstehen.

**Modellbildung mit Petrinetzen.** Die Beschreibung technischer Systeme durch Petrinetze wird dadurch erleichtert, dass wichtige Elemente der Modellbildung durch entsprechende Netzelemente explizit ausgedrückt werden können:

- **Prozessorientierte Modellbildung:** Das Systemverhalten wird durch eine Aufeinanderfolge von Teilprozessen dargestellt. Das dynamische Verhalten des Systems besteht aus Abschnitten, in denen einer oder mehrere parallel ablaufende Teilprozesse aktiviert sind. Ereignisse kennzeichnen den Beginn oder den Abschluss eines oder mehrerer Teilprozesse. Die Teilprozesse können direkt als parallele Pfade eines Petrinetzes repräsentiert werden.
- **Zustandsraum:** Der Systemzustand wird durch die Markierung des Petrinetzes bestimmt, der Zustandsraum also durch die Menge aller möglichen Markierungen, die von der Netzstruktur abhängt.
- **Ereignisse:** Die das Systemverhalten charakterisierenden Ereignisse werden durch das Schalten von Transitionen nachgebildet.
- **Kausalität:** Ereignisse können nur eintreten, wenn alle dafür erforderlichen Bedingungen erfüllt sind. Diese Bedingungen werden in Petrinetzen durch Stellen repräsentiert. Eine Bedingung ist erfüllt, wenn die zugehörige Stelle markiert ist.
- **Nichtdeterminismus:** Ereignisse in parallel ablaufenden Teilprozessen können in beliebiger Folge zueinander auftreten, woraus ein nichtdeterministisches Systemverhalten resultiert. Petrinetze beschreiben diesen Nichtdeterminismus.
- **Dynamik:** Das dynamische Verhalten des Netzes wird durch den Markenfluss charakterisiert, wodurch eine Folge von Ereignissen entsteht.

Dementsprechend wird bei der Verwendung von Petrinetzen im 1. Schritt des Modellbildungsalgorithmus 11.1 nicht der gesamte Zustandsraum explizit aufgeschrieben, sondern es werden die Situationen identifiziert, in denen Teilprozesse beginnen bzw. enden. Die Bedingungen werden im Petrinetz durch Stellen und die Teilprozesse durch Transitionen oder Ketten von Stellen und Transitionen repräsentiert und entsprechend ihrer kausalen Abhängigkeit durch Kanten miteinander verbunden. Der in der Netztheorie gebräuchliche Begriff Bedingungs/Ereignis-Netz für die in diesem Abschnitt eingeführten Petrinetze weist auf den beschriebenen Modellbildungsweg hin, Bedingungen zu definieren, unter denen Ereignisse eintreten können.

Ob ein Teilprozess in einem Petrinetz durch eine einzige Transition oder durch eine Kette von Stellen und Transitionen dargestellt wird, hängt vor allem von seiner Kopplung mit anderen Teilprozessen ab. Gibt es keinen mit diesem Teilprozess verkoppelten zweiten Prozess, so reicht

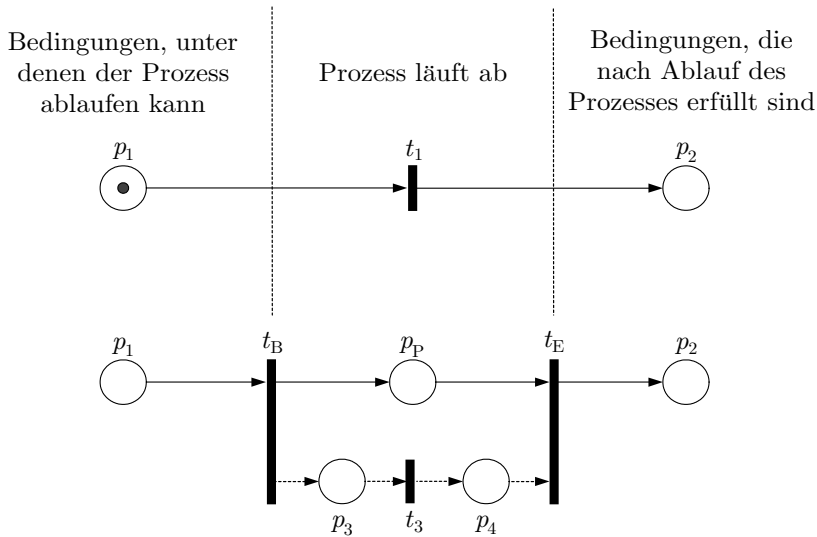


Abb. 11.33: Prozessorientierte Modellbildung mit Petrinetzen

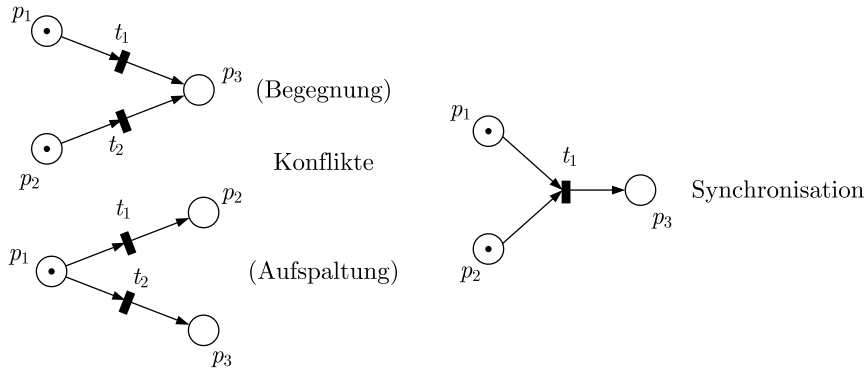
eine Transition wie die Transition  $t_1$  in Abb. 11.33 für seine Darstellung aus. In dem abgebildeten Petrinetz beschreibt die Stelle  $p_1$  die Situation, in der der betrachtete Teilprozess beginnen kann, und die Stelle  $p_2$  die Bedingungen, die nach dem Ablauf des Teilprozesses erfüllt sind.

Wird der Teilprozess mit einem zweiten gekoppelt, der erst nach Beginn des ersten Teilprozesses starten kann und vor dem Ende des ersten abgeschlossen ist, so ist die Transition  $t_1$  eine zu grobe Darstellung. Im unteren Teil der Abbildung wurden die Transitionen  $t_B$  zur Kennzeichnung des Beginns und die Transition  $t_E$  zur Darstellung des Endes des ersten Teilprozesses in das Netz eingefügt. Wenn die Stelle  $p_P$  markiert ist, ist der Teilprozess aktiv. Durch diese Erweiterung ist es möglich, durch das Petrinetz den Sachverhalt zu beschreiben, dass der zweite Teilprozess, der hier durch zwei weitere Stellen und eine weitere Transition repräsentiert wird, nach dem ersten startet und vor dem ersten beendet ist.

**Spezielle Petrinetzelemente.** Mit den angegebenen Petrinetzelementen kann man „Bausteine“ bilden, die mit gleicher Bedeutung in vielen Modellen wiederkehren. Drei davon sind in Abb. 11.34 gezeigt. Der linke Teil stellt zwei *Konflikte* (Alternativen) dar. Auf Grund der Schaltregel sind im oberen Teil die beiden Transitionen  $t_1$  und  $t_2$  aktiviert, jedoch nur eine von beiden kann schalten. Dies ist ein Konflikt, weil durch die Schaltregel nicht entschieden wird, welche der beiden aktivierten Transitionen feuert und weil nach dem Schalten der einen Transition auf Grund des dadurch entstehenden Kontakts die andere Transition nicht mehr aktiviert ist. Diese Konfliktsituation kann auf mehr als zwei Plätze und Transitionen erweitert werden.

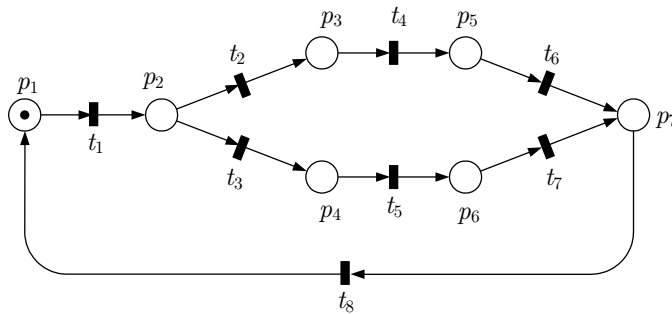
Ein anderer Konflikt tritt in Petrinetzen auf, wenn, wie bei nichtdeterministischen Automaten, mehrere mögliche Zustandsübergänge auftreten. Diese Situation liegt beispielsweise in dem in Abb. 11.35 gezeigten Petrinetz an der Stelle  $p_2$  vor, deren Markierung entweder das Schalten der Transition  $t_2$  oder der Transition  $t_3$  auslösen kann.





**Abb. 11.34:** Darstellung von Konflikt und Synchronisation in einem Petrinetz

Konflikte werden hier wie nichtdeterministische Zustandsübergänge von Automaten behandelt. Es wird davon ausgegangen, dass eine beliebige der aktivierten Transitionen schaltet. Petrinetze mit dieser Art der Konfliktlösung werden auch als *Free-Choice-Netze* bezeichnet.



**Abb. 11.35:** Petrinetz mit Konflikt an der Stelle  $p_2$ , der zur Blockierung führt

Der in Abb. 11.34 rechts dargestellte Teil eines Netzes beschreibt eine *Synchronisation*, wie sie z. B. im rechten Teil von Abb. 11.30 auftritt. Die Transition  $t_1$  ist nur dann aktiviert, wenn beide Prästellen markiert sind. In Abb. 11.30 liegen vor dieser Synchronisation zwei parallele Pfade im Netz, die gleichzeitig ablaufende Prozesse repräsentieren. Für das Schalten der Transition  $t_5$  ist es gleichgültig, in welchem Pfad die Marke schneller läuft. Erst wenn Marken beide Pfade durchlaufen haben und die Stellen  $p_5$  und  $p_6$  markiert sind, ist diese Transition aktiviert. Die Transition  $t_5$  synchronisiert also Teilprozesse, was diesem Netzelement seinen Namen gibt.

**Matrixdarstellung autonomer Petrinetze.** Petrinetze können durch eine Netzmatrix  $N$  beschrieben werden, die die Verschiebung der Marken beim Schalten der Transitionen angibt. Die Matrix  $N$  hat für jede Stelle  $p_i$  eine Zeile. Die  $j$ -te Spalte gehört zur Transition  $t_j$ . Das Element  $n_{ij}$  hat den Wert  $+1$ , wenn es eine Kante von der Transition  $t_j$  zur Stelle  $p_i$  gibt und folglich beim Schalten der Transition  $t_j$  eine Marke über diese Kante auf den Platz  $p_i$  fließt. Es hat den Wert  $-1$ , wenn eine Kante von der Stelle  $p_i$  zur Transition  $t_j$  führt, auf der die Marke von  $p_i$  abfließt. Diese Definition der Matrixelemente kann man sich abgekürzt in der Form

$$n_{ij} = \begin{cases} +1 & \text{wenn } t_j \rightarrow p_i \\ -1 & \text{wenn } p_i \rightarrow t_j \end{cases}$$

merken. Alle anderen Elemente sind gleich null. Für die grafische Darstellung des Petrinetzes ist  $N$  die Inzidenzmatrix, also die Matrix, die die Zusammengehörigkeit der Knoten und Kanten beschreibt.

Die Netzmatrix ist nur für reine Netze eine zur grafischen Darstellung äquivalente Beschreibung, weil bei Schlingen von der Stelle  $p_i$  über die Transition  $t_j$  zurück zur Stelle  $p_i$  das zugehörige Element  $n_{ij}$  gleich null sein muss, um den Markenfluss richtig zu beschreiben, und Schlingen deshalb nicht in der Netzmatrix erscheinen. Bevor man zu dieser Matrixrepräsentation übergeht, muss man Schlingen deshalb entsprechend Abb. 11.31 eliminieren.

Zur Beschreibung der zum Zeitpunkt  $k$  vorhandenen Markierung  $M(k)$  führt man einen  $N$ -dimensionalen Vektor  $\mathbf{p}(k)$  ein, wobei  $N$  die Anzahl der Stellen im Petrinetz bezeichnet. Es wird dieselbe Notation wie bei Automaten verwendet: Das  $i$ -te Element  $p_i(k)$  des Vektors  $\mathbf{p}(k)$  ist gleich eins, wenn die Stelle  $p_i$  zur Zeit  $k$  markiert ist, andernfalls gleich null.  $\mathbf{p}(k)$  heißt Markierungsvektor.

Die zum Zeitpunkt  $k$  schaltenden Transitionen werden durch den Transitionsvektor (Schaltvektor)  $\mathbf{t}(k)$  beschrieben, dessen  $i$ -tes Element gleich eins ist, wenn die Transition  $t_i$  zur Zeit  $k$  schaltet, andernfalls gleich null. Aufgrund der Schaltregel dürfen im Schaltvektor nur aktivierte Transitionen mit eins belegt werden, was durch die Beziehung

$$\mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k))$$

zum Ausdruck gebracht wird.

Mit diesen Größen kann man das Petrinetz durch die folgende Beziehung darstellen:

Zustandsraumdarstellung eines Petrinetzes  $\mathcal{PN}$ :

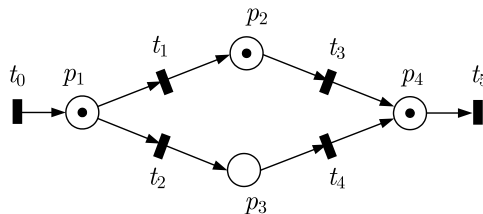
$$\mathcal{PN} : \begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + N \mathbf{t}(k) & \text{für } \mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k)) \\ \mathbf{p}(0) = \mathbf{p}_0. \end{cases}$$

(11.63)

Sie ist eine rekursive Darstellung der Markierungsfolge  $\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots$ , die den in den Gln. (11.10) und (11.26) angegebenen Beziehungen für autonome deterministische bzw. nicht-deterministische Automaten ähnlich ist.

**Beispiel 11.10** Beschreibung der Kopplung von vier Werkzeugmaschinen durch ein Petrinetz

Für die Beschreibung der im Beispiel 11.4 auf S. 363 betrachteten Werkzeugmaschinen führt man für jede Maschine eine Stelle ein mit der Interpretation, dass die Maschine arbeitet, wenn die zugehörige Stelle markiert ist (Abb. 11.17). Das Schalten der Transitionen verschiebt die Markierung, was in diesem Beispiel als Bewegung eines Werkstücks interpretiert werden kann. Abbildung 11.36 zeigt das Petrinetz.



**Abb. 11.36:** Petrinetz zur Beschreibung der Arbeitsweise von vier Maschinen

Da die Maschine  $M_1$  durch ein von außen ankommendes Werkstück aktiviert wird und ständig Werkstücke auf die Bearbeitung warten, hat das Petrinetz an der linken Seite eine Transition ohne Prästelle. Diese Transition  $t_0$  ist immer dann aktiviert, wenn die Stelle  $p_1$  nicht markiert ist. Sie sorgt dafür, dass die Stelle  $p_1$  immer erneut markiert wird, wenn die Marke von  $p_1$  auf  $p_2$  oder  $p_3$  abgeflossen ist.

Auf der rechten Seite führt die Transition  $t_5$  eine Marke von der Stelle  $p_4$  in die Umgebung ab, was bedeutet, dass die Maschine  $M_4$  ihre Bearbeitung beendet hat.

Das Netz kann dieselben Zustände annehmen und dieselben Zustandsübergänge erzeugen wie der nichtdeterministische Automat in Abb. 11.18 auf S. 365, aber das Netz ist offensichtlich viel kleiner als der Automatengraph. Wenn man die Markierung so wie die Automatenzustände beschreibt, wobei eine Eins jetzt das Vorhandensein einer Marke auf der entsprechenden Stelle kennzeichnet, dann kann man das Verhalten des Automaten direkt mit dem des Petrinetzes vergleichen. Die in Abb. 11.36 gezeigte Markierung entspricht dem Automatenzustand

$$\begin{pmatrix} 1 \\ 1 & 1 \\ 0 \end{pmatrix},$$

der in der oberen rechten Ecke des Automatengraphen zu finden ist. Es gibt zwei Nachfolgezustände im Automatengraphen, die im Petrinetz durch das Schalten der Transitionen  $t_5$  bzw.  $t_2$  entstehen:

$$\begin{pmatrix} 1 \\ 1 & 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 1 \\ 0 & 1 \\ 1 \end{pmatrix}.$$

Verfolgt man die Bewegung der Marken weiter, so erhält man dieselben Zustände wie beim Automaten.

Das nichtdeterministische Verhalten des Petrinetzes entsteht einerseits durch die unterschiedliche Reihenfolge, in der Transitionen auf parallelen Pfaden im Petrinetz schalten können. Andererseits entstehen weitere nichtdeterministische Übergänge durch Konflikte (vgl. Abb. 11.34). Wenn die Stellen  $p_2$  und  $p_3$  markiert sind und die Stelle  $p_4$  keine Marke hat, tritt der Konflikt auf, dass als nächstes entweder

die Transition  $t_3$  oder die Transition  $t_4$  schalten kann. Dadurch entstehen zwei mögliche Nachfolgemarkierungen.

Wenn andererseits von der abgebildeten Initialmarkierung ausgehend zuerst die Transition  $t_5$  und dann die Transition  $t_3$  schaltet, so entsteht auf der linken Seite des Netzes der Konflikt, dass die beiden Transitionen  $t_1$  und  $t_2$  aktiviert sind, wodurch zwei unterschiedliche Ereignisfolgen erzeugt werden können.

Die Netzmatrix  $N$  hat folgendes Aussehen

$$N = \begin{array}{c} p_1 \\ p_2 \\ p_3 \\ p_4 \end{array} \begin{array}{c} t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5 \\ \left( \begin{array}{cccccc} 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{array} \right), \end{array}$$

wobei zur Erläuterung der Einträge die Zeilen und Spalten mit den entsprechenden Stellen und Transitionen gekennzeichnet sind. Die Eins in der oberen linken Ecke besagt, dass beim Schalten der Transition  $t_0$  die Stelle  $p_1$  eine Marke erhält. In der zweiten Spalte stehen eine  $-1$  und eine  $1$ , weil beim Schalten der Transition  $t_1$  eine Marke von der Stelle  $p_1$  zur Stelle  $p_2$  wandert.

Das Verhalten des Netzes bei der Initialmarkierung  $M_0$ , die durch den Vektor  $\mathbf{p}(0) = (1 \ 0 \ 0 \ 0)^T$  beschrieben ist, erhält man durch Anwendung der Gl. (11.63) folgendermaßen. Bei der Initialmarkierung sind die Transitionen  $t_1$  und  $t_2$  auf Grund der Marke in der Stelle  $p_1$  aktiviert. Wenn  $t_1$  schaltet, hat der Vektor  $\mathbf{t}(0)$  in dem zu  $t_1$  gehörigen zweiten Element eine Eins, sonst Nullen:

$$\mathbf{t}(0) = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \begin{array}{l} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{array}$$

Damit erhält man für die Nachfolgemarkierung aus Gl. (11.63)

$$\begin{aligned} \mathbf{p}(1) &= \mathbf{p}(0) + N \mathbf{t}(0) \\ &= \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \end{aligned}$$

bei der die Stelle  $p_2$  markiert ist. □

**Synchronisationsgraphen.** Bei der als Synchronisationsgraphen bezeichneten speziellen Form von Petrinetzen hat jede Stelle genau eine Eingangstransition und eine Ausgangstransition. Andererseits können die Transitionen mehrere Präkanten und Postkanten besitzen. Abbildung 11.37 zeigt ein Beispiel.

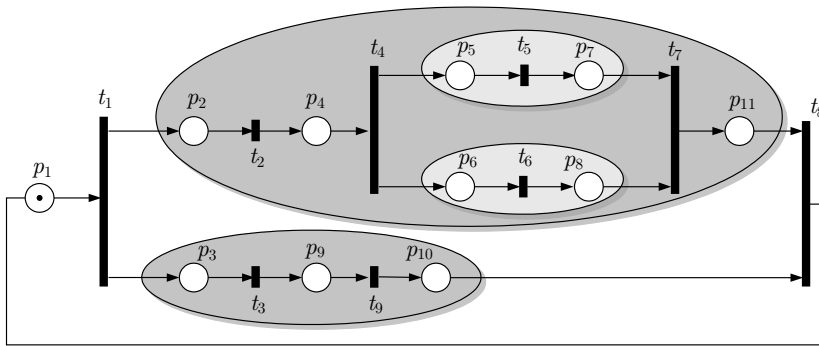


Abb. 11.37: Beispiel für einen Synchronisationsgraphen

Synchronisationsgraphen entstehen vor allem bei der Modellierung von Systemen, in denen mehrere Prozesse parallel ablaufen. Das Ende dieser Teilprozesse kann durch eine gemeinsame Transition synchronisiert werden, woraus auch die Bezeichnung dieser speziellen Petrinetze resultiert. So wird der durch den Synchronisationsgraphen aus Abb. 11.37 beschriebene Prozess zwischen den Transitionen  $t_1$  und  $t_8$  in zwei Teilprozesse zerlegt, die dunkel hinterlegt sind. Zwischen den Transitionen  $t_4$  und  $t_7$  wird der obere der beiden Teilprozesse noch einmal in zwei Teilprozesse aufgeteilt. Wie dieses Beispiel zeigt, eignen sich Synchronisationsgraphen vor allem zur Beschreibung von Systemen, deren Bewegung sich aus einer genau bekannten Folge von Teilschritten zusammensetzt, wobei Teilschritte auch parallel ablaufen können. Diese Teilschritte können eine unterschiedliche Dauer haben, auch eine nicht genau bekannte Dauer, so dass sie in unterschiedlicher Reihenfolge enden und Nachfolgeprozesse anregen.

Charakteristisch für Systeme, die durch Synchronisationsgraphen beschrieben werden, ist die Tatsache, dass die Folge von Teilprozessen an bestimmten Stellen synchronisiert wird, wobei die früher beendeten Teilprozesse auf die später zu Ende gehende Teilprozesse warten und erst nach Abschluss aller Teilprozesse der Folgeprozess beginnt. Synchronisationsgraphen entstehen also insbesondere dort, wo sich das bestimmungsgemäße Verhalten eines Systems aus einer Folge regelmäßig wiederholter Teilprozesse zusammensetzt.

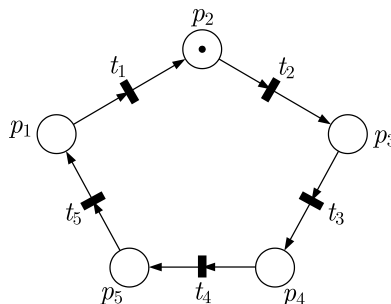
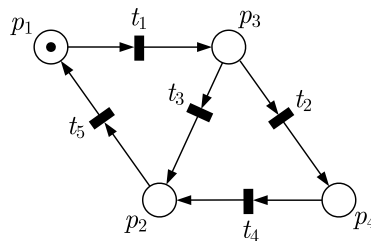


Abb. 11.38: Petrinetz, das zum Automaten in Abb. 11.6 äquivalent ist

**Zustandsmaschinen.** Eine andere spezielle Klasse von Petrinetzen ist dadurch ausgezeichnet, dass jede Transition genau eine Vorgängerstelle und eine Nachfolgestelle besitzt. Derartige Petrinetze werden als Zustandsmaschinen bezeichnet. Die Bezeichnung kommt vom englischen Begriff *state machine*, der für Automaten gebraucht wird. Tatsächlich kann man aus jedem Automaten durch Einfügen je einer Transition zwischen alle benachbarten Zustandsknoten eine Zustandsmaschine machen. Abbildung 11.38 zeigt das auf diesem Wege aus dem Automaten in Abb. 11.6 gewonnene Petrinetz. Da für den Automaten  $z_0 = 2$  gilt, ist die Stelle  $p_2$  markiert. Für Zustandsmaschinen besitzt die Netzmatrix  $N$  in jeder Spalte genau eine  $+1$  und eine  $-1$ .



**Abb. 11.39:** Zustandsmaschine mit nichtdeterministischem Verhalten

Die Bedingung, dass jede Transition genau eine Prästelle und eine Poststelle besitzt, lässt nichtdeterministisches Verhalten durch Aufspaltungen und Begegnungen der Markenflüsse zu (vgl. Abb. 11.34). Abbildung 11.39 zeigt als Beispiel dafür eine Zustandsmaschine mit einem Konflikt in der Stelle  $p_3$ .

Zustandsmaschinen (und demzufolge auch Automaten) eignen sich zur Beschreibung von Systemen, deren Verhalten durch eine Ereignisfolge beschrieben wird, die sich gegebenenfalls verzweigt. Parallel ablaufende Prozesse können nicht explizit dargestellt werden. Wenn die Anfangsmarkierung einer Zustandsmaschine eine Marke enthält, so ist die Markierung zu jedem Zeitpunkt durch genau eine Marke beschrieben. Die einzige markierte Stelle beschreibt dann den Systemzustand, wie es auch bei Automaten der Fall ist.

**Aufgabe 11.13\*** Beschreibung eines Batchprozesses durch ein Petrinetz

Der in Abb. 11.40 gezeigte Batchprozess wird eingesetzt, um in den Behältern  $B_1$  und  $B_2$  Flüssigkeiten vorzulegen, die nach dem Entleeren des Behälters  $B_3$  in diesen gefüllt werden. Wenn der Behälter  $B_3$  gefüllt ist, wird der Rührer angeschaltet und die Flüssigkeit erhitzt, während gleichzeitig die Behälter  $B_1$  und  $B_2$  über die Pumpe  $P$  bzw. das Ventil  $V$  wieder aufgefüllt werden. Beschreiben Sie diesen Vorgang durch ein Petrinetz. □

### 11.5.2 Petrinetze mit Eingang und Ausgang

Die Erweiterung der Petrinetze um Eingänge und Ausgänge soll es ermöglichen, das Schalten der Transitionen von außen durch Eingaben zu beeinflussen und den Stellen Ausgaben zuzuordnen. Man spricht hierbei auch von einer *steuerungstechnischen Interpretation* des Petrinetzes

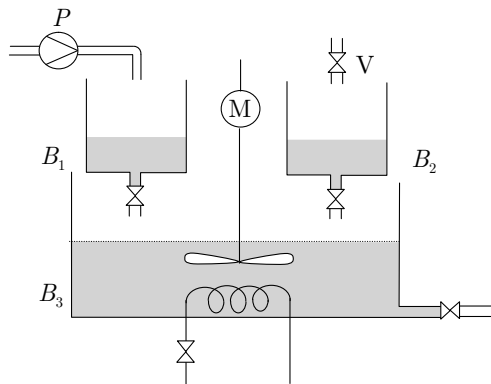


Abb. 11.40: Batchprozess

bzw. von einem *steuerungstechnisch interpretierten Petrinetz*, weil man den abstrakten Netzelementen „Stellen“ und „Transitionen“ eine steuerungstechnische Bedeutung zuweist.

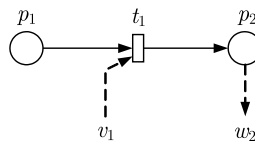


Abb. 11.41: Elemente eines steuerungstechnisch interpretierten Petrinetzes

**Eingänge.** In einem steuerungstechnisch interpretierten Petrinetz wird der Transition  $t_i$  die boolesche Eingangsgröße  $v_i$  (mit demselben Index  $i$ ) zugeordnet, was in Abb. 11.41 durch den Pfeil dargestellt ist. Um die durch eine Steuerung beeinflussbaren Transitionen hervorzuheben, werden diese Transition durch ein nicht ausgefülltes Rechteck dargestellt. Die Transition  $t_i$  kann nur schalten, wenn sie nach der Schaltregel für autonome Petrinetze aktiviert ist und wenn gleichzeitig für die Eingangsgröße die Beziehung  $v_i = 1$  gilt. Die Zusatzbedingung erweitert die Schaltregel folgendermaßen:

**Schaltregel für Transitionen mit Eingang:** Eine Transition mit Eingang ist *aktiviert*, wenn

- alle ihre Prästellen markiert sind,
- alle Poststellen nicht markiert sind und
- die zugeordnete Eingangsgröße den Wert eins hat.

Für die algebraische Darstellung des erweiterten Petrinetzes fasst man die Eingangsgrößen aller Transitionen zum  $R$ -dimensionalen Vektor  $v$  zusammen, wobei  $R$  die Anzahl der Transi-

tionen angibt. Wenn eine Transition keine Eingangsgröße besitzt, wird in ihrer grafischen Darstellung der Pfeil weggelassen und für sie gilt die alte Schaltregel. Das entsprechende Element im Vektor  $\mathbf{v}$  hat dann stets den Wert eins.

Die Menge der aktivierten Transitionen ist jetzt nicht nur vom Markierungsvektor  $\mathbf{p}(k)$ , sondern auch von der Eingabe  $\mathbf{v}(k)$  abhängig:  $\mathcal{T}_{\text{akt}}(\mathbf{p}(k), \mathbf{v}(k))$ . Die Beschreibung (11.63) des Petrinetzes wird deshalb auf

$$\mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{N} \mathbf{t}(k) \quad \text{für } \mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k), \mathbf{v}(k)) \quad (11.64)$$

erweitert.

In einer allgemeineren Darstellung kann man den Transitionen  $t_i$  einen Schaltausdruck  $q_i(\mathbf{v})$  zuweisen. Dies ist ein boolescher Ausdruck in den Eingangsvariablen  $v_1, v_2, \dots, v_R$ . Die Transition  $t_i$  ist nur aktiviert, wenn auch ihr Schaltausdruck den Wert eins hat. Die Verwendung von Schaltausdrücken macht es möglich, mehrere Transitionen durch dieselben Eingangsgrößen zu beeinflussen. Diese Erweiterung ist jedoch bei den im Folgenden betrachteten Beispielen nicht notwendig.

Durch die Wahl des Eingangs kann man nicht nur festlegen, ob eine Transition schalten kann, sondern auch den Zeitpunkt bestimmen, an dem die Transition schaltet. Transitionen, denen ein Schaltausdruck zugeordnet ist, werden deshalb *steuerbar* genannt. Dann heißt auch das durch die Transition repräsentierte Ereignis steuerbar.

**Ausgänge.** Die zweite Erweiterung des Petrinetzes betrifft die Ausgabe. Wenn man jeder Stelle  $p_i$  ein boolesches Ausgangssignal  $w_i(k)$  mit demselben Index zuordnet, das genau dann den Wert eins hat, wenn die zugehörige Stelle markiert ist, stimmt der Vektor  $\mathbf{w}(k)$  aller Ausgangsgrößen mit dem Markierungsvektor  $\mathbf{p}(k)$  überein ( $\mathbf{w}(k) = \mathbf{p}(k)$ ) und man kann den Markierungszustand des Netzes anhand der Ausgabe vollständig erkennen. Allerdings ist dies bei technischen Anwendungen häufig nicht der Fall. Die Ausgabe  $\mathbf{w}(k)$  enthält typischerweise nur einen Teil der Informationen von  $\mathbf{p}(k)$ , was durch die Funktion  $\mathbf{h}$  ausgedrückt wird, mit der die Beziehung

$$\mathbf{w}(k) = \mathbf{h}(\mathbf{p}(k))$$

gilt. Ähnlich wie beim Moore-Automaten bestimmt die aktuelle Markierung  $\mathbf{p}(k)$  die aktuelle Ausgabe  $\mathbf{w}(k)$ . Dies lässt zu, dass zu unterschiedlichen Markierungen derselbe Wert ausgegeben wird.

In der grafischen Darstellung wird die Ausgabe mit einer gestrichelt eingetragenen Kante, die von der betreffenden Stelle ausgeht, gekennzeichnet (Abb. 11.41).

Zusammengefasst werden Systeme  $\mathcal{PN}$ , die durch ein Petrinetz mit Eingang und Ausgang dargestellt sind, durch die folgenden Gleichungen beschrieben:

Zustandsraumdarstellung von Petrinetzen mit Eingang und Ausgang:

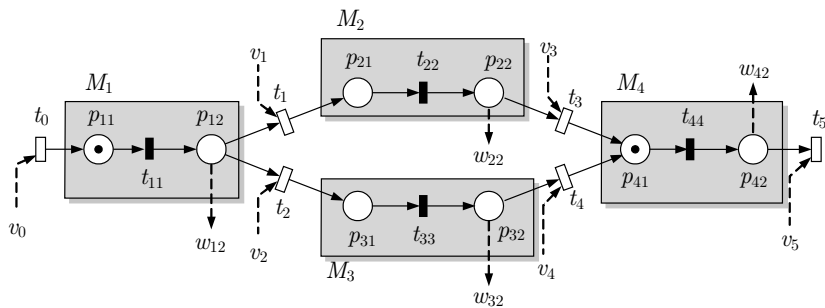
$$\mathcal{PN} : \begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{N} \mathbf{t}(k) & \text{für } \mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k), \mathbf{v}(k)) \\ \mathbf{w}(k) = \mathbf{h}(\mathbf{p}(k)) \\ \mathbf{p}(0) = \mathbf{p}_0. \end{cases} \quad (11.65)$$



**Beispiel 11.11** Beschreibung von vier Werkzeugmaschinen für die Materialflussteuerung

In den Beispielen 11.4 und 11.10 wurden vier Werkzeugmaschinen unter der Wirkung ihrer Steuerung betrachtet, wobei die Maschinen zusammen mit der Steuerung ein autonomes System bildeten. Jetzt sollen die vier Maschinen als Steuerstrecke modelliert werden, die über folgende Eingangs- und Ausgangsgrößen mit der Steuereinrichtung kommuniziert:

Eingangsgröße	Wirkung der Eingangsgröße (wenn diese Eingangsgröße den Wert eins hat)
$v_0$	Ein Werkstück soll der Maschine $M_1$ von außen zugeführt werden.
$v_1$	Ein Werkstück soll von der Maschine $M_1$ zur Maschine $M_2$ transportiert werden.
$v_2$	Ein Werkstück soll von der Maschine $M_1$ zur Maschine $M_3$ transportiert werden.
$v_3$	Ein Werkstück soll von der Maschine $M_2$ zur Maschine $M_4$ transportiert werden.
$v_4$	Ein Werkstück soll von der Maschine $M_3$ zur Maschine $M_4$ transportiert werden.
$v_5$	Ein Werkstück soll von der Maschine $M_4$ abtransportiert werden.
Ausgangsgröße	Bedeutung der Ausgangsgröße (wenn diese Ausgangsgröße den Wert eins hat)
$w_{12}$	Die Maschine $M_1$ hat die Bearbeitung des Werkstücks abgeschlossen.
$w_{22}$	Die Maschine $M_2$ hat die Bearbeitung des Werkstücks abgeschlossen.
$w_{32}$	Die Maschine $M_3$ hat die Bearbeitung des Werkstücks abgeschlossen.
$w_{42}$	Die Maschine $M_4$ hat die Bearbeitung des Werkstücks abgeschlossen.



**Abb. 11.42:** Petrinetz zur Beschreibung der Werkzeugmaschinen als Steuerstrecke

Die Steuerung bezieht sich also auf den Materialfluss, während angenommen wird, dass die Bearbeitungsschritte der Werkstücke durch unterlagerte, hier nicht betrachtete Steuereinrichtungen ausgelöst werden.

Die Wirkung der Eingangsgrößen sieht man an folgendem Beispiel. Indem die Steuereinrichtung die Eingangsgröße  $v_1$  auf den Wert eins setzt, legt sie fest, dass das Werkstück von der Maschine  $M_1$  zur Maschine  $M_2$  transportiert wird. Allerdings bestimmt nicht der Wert der Eingangsgröße allein, sondern nur in Verbindung mit den Markierungen das Schalten der Transition  $t_1$ . Nur wenn die Stelle  $p_{12}$  markiert und die Stelle  $p_{21}$  nicht markiert ist, kann die Transition  $t_1$  schalten, was dem gewünschten Transport des Werkstücks entspricht.

Um die Abhängigkeit der Werkzeugmaschinen von der Steuereinrichtung in der beabsichtigten Weise darstellen zu können, muss bei der Modellierung – in Erweiterung des in Abb. 11.36 angegebenen

Modells – zwischen dem Beginn und dem Ende der Bearbeitung der Werkstücke unterschieden werden. Die Eingangsgrößen können nur wirksam sein, wenn die Maschinen für den Werkstücktransport frei sind. Die Ausgaben treten erst dann auf, wenn die Maschinen ihre Arbeit beendet haben.

Deshalb wurden in Abb. 11.42 die Stellen  $p_1$ ,  $p_2$ ,  $p_3$  und  $p_4$  aus Abb. 11.36 durch je zwei Stellen ersetzt und durch je eine Transition verbunden. Das Schalten der neu eingeführten Transitionen entspricht dem Beenden der Werkstückbearbeitung an der entsprechenden Maschine. Die Transitionen  $t_0$  und  $t_5$  wurden eingeführt, damit die Werkstückzuführung bzw. -abführung für die beiden äußeren Maschinen von der Steuereinrichtung beeinflusst werden kann.  $\square$

#### Aufgabe 11.14\* Steuerung einer Digitalkamera

Eine Digitalkamera fokussiert bei halb gedrücktem Auslöser auf das sich im Mittelpunkt des Suchers befindende Objekt und bestimmt die Belichtungszeit. Erst nach Beendigung beider Vorgänge und bei vollständigem Durchdrücken des Auslösers wird ein Bild belichtet. Die Kamera speichert anschließend das Bild als Datei ab.

1. Schreiben Sie die im Fotoapparat ablaufenden Prozesse auf. Geben Sie an, in welchem Teil der Kamera die Prozesse ablaufen und unter welchen Bedingungen die Prozesse beginnen können.
2. Zeichnen Sie ein Petrinetz, das die Prozesse zum gewünschten Ablauf des Fotografierens verknüpft.
3. Das von Ihnen gezeichnete Petrinetz soll als Beschreibung für eine Steuerung verwendet werden, die die Eingangssignale an den Fotoapparat sendet, um die einzelnen Prozesse auszulösen, und die die Ausgangssignale des Fotoapparates empfängt, die die Beendigung der Teilprozesse anzeigen. Markieren Sie diese Signale als Eingaben und Ausgaben des Petrinetzes.
4. Zeichnen Sie ein Blockschaltbild, das das Zusammenwirken der Komponenten des Fotoapparates mit der Steuerung zeigt.  $\square$

#### Aufgabe 11.15\* Beschreibung eines Streckenabschnitts der Eisenbahn

Der in Abb. 11.43 gezeigte Ausschnitt des Streckennetzes der Eisenbahn enthält zwei einspurige Abschnitte, die durch eine Ausweichstelle verbunden sind, auf der ein Zug einem anderen begegnen kann. Die Freigabe der einspurigen Streckenabschnitte erfolgt durch Signale. Es soll angenommen werden, dass die Strecke viel befahren ist und deshalb in den Stationen A und B stets Züge darauf warten, nach B bzw. A zu fahren.

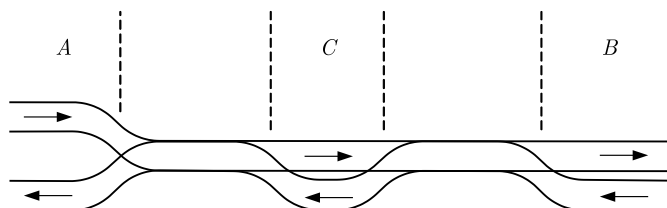


Abb. 11.43: Streckenabschnitt einer Eisenbahnverbindung

Beschreiben Sie den Eisenbahnbetrieb auf diesem Streckenabschnitt durch ein Petrinetz, dessen Eingaben Fahrtwünsche für die einzelnen Streckenabschnitte darstellen.  $\square$

### 11.5.3 Beziehungen zwischen Petrinetzen und Automaten

Bei der Einführung des Petrinetzes im Abschn. 11.5.1 wurde bereits darauf hingewiesen, dass Petrinetze zu einer kompakteren Beschreibung nichtdeterministischer Systeme führen können. Hier soll die Frage untersucht werden, ob Petrinetze damit auch neue, mit Automaten nicht darstellbare Phänomene beschreiben können.

Als erste Antwort wird zunächst gezeigt, dass sich Automaten in Petrinetze und umgekehrt Petrinetze in Automaten überführen lassen. Die erste Richtung ist einfach und bei der Einführung der Zustandsmaschinen im Abschn. 11.5.1 bereits behandelt worden: Wird im Graphen eines endlichen Automaten in jede Kante eine Transition eingefügt, entsteht ein Petrinetz. Wenn der Kante eine Eingabe  $v$  zugeordnet ist, wird der Transition ein Schaltausdruck zugeordnet, der genau dann den Wert eins hat, wenn die Eingabe  $v$  erscheint.

Umgekehrt gilt: Ein Petrinetz, in dem jede Transition höchstens eine Prästelle und höchstens eine Poststelle besitzt, kann durch Weglassen der Transitionen in einen deterministischen Automaten überführt werden. Für andere Petrinetze ist diese Transformation schwieriger. Um einen Automaten zu erhalten, der dasselbe Verhalten wie das in Abb. 11.30 gezeigte Petrinetz besitzt, muss man alle möglichen Markierungen ermitteln, denn jede mögliche Markierung des Petrinetzes entspricht einem Systemzustand, der auch im Automaten vorkommen muss. Wie die Folgen in Gl. (11.62) zeigen, gibt es außer den alleinigen Markierungen der Stellen  $p_1$ ,  $p_2$  und  $p_7$  vier Kombinationsmöglichkeiten:

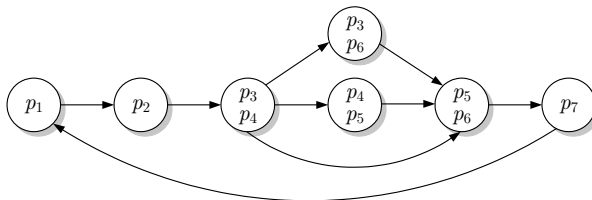
$$\begin{pmatrix} p_3 \\ p_4 \end{pmatrix}, \begin{pmatrix} p_3 \\ p_6 \end{pmatrix}, \begin{pmatrix} p_4 \\ p_5 \end{pmatrix}, \begin{pmatrix} p_5 \\ p_6 \end{pmatrix}.$$

Deshalb muss der Automat Zustände besitzen, die diesen Markierungen des Petrinetzes entsprechen. Abbildung 11.44 zeigt diesen Automaten. Die Zustände sind so bezeichnet, dass man erkennt, welchen Markierungen des Petrinetzes sie entsprechen.

Die hier angegebenen Beziehungen zeigen:

|| Automaten und Petrinetze (in den hier eingeführten Grundformen) beschreiben dieselbe Klasse ereignisdiskreter Systeme.

Alle im Folgenden für Automaten entwickelten Methoden können deshalb auf Petrinetze übertragen werden und umgekehrt.



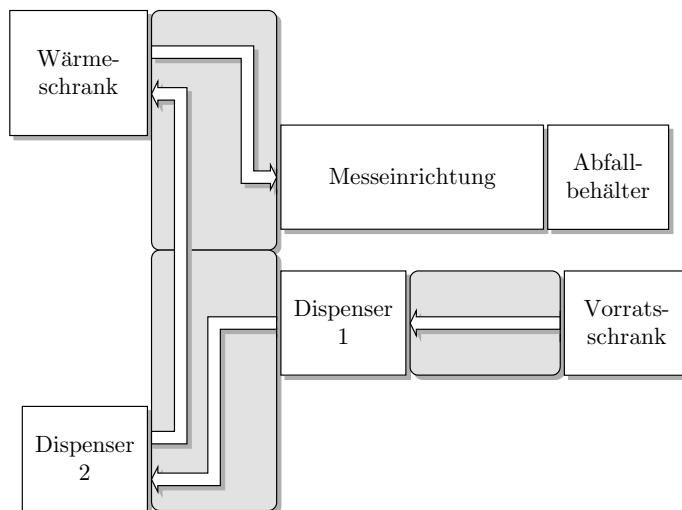
**Abb. 11.44:** Automat mit demselben Verhalten wie das Petrinetz aus Abb. 11.30

**Aufgabe 11.16** Beschreibung der Arbeitsweise eines Verbrennungsmotors

Beschreiben Sie die zyklische Arbeitsweise eines Verbrennungsmotors durch ein Petrinetz. Ist das Netz ein Synchronisationsgraph oder eine Zustandsmaschine? □

**Aufgabe 11.17** Steuerung einer Screening-Anlage

Ein wichtiges Forschungsziel in der pharmazeutischen Industrie ist die Entdeckung neuer Wirkstoffe. Da man hierfür wenige systematische Wege kennt, werden Millionen von Substanzen daraufhin untersucht, ob sie die Aktivität von krankheitsrelevanten Eiweißkörpern wie Enzymen oder Zelloberflächenrezeptoren hemmen bzw. aktivieren. Dieser Vorgang („Screening“) muss automatisiert ablaufen.



**Abb. 11.45:** Aufbau der Screening-Anlage

Abbildung 11.45 zeigt den prinzipiellen Aufbau einer Anlage, mit der Stoffe getestet werden können. Die einzelnen Komponenten der Anlage sind durch zwei Transportbänder verbunden, die in der Abbildung grau gezeichnet sind. Die Pfeile kennzeichnen die im Folgenden erwähnten Transportvorgänge.

Die zu testenden Substanzen sind in Mikrotiterplatten verteilt, die sich im Vorratschrank befinden. Jede dieser Platten besitzt 1536 kleine Vertiefungen, in die die Substanzen nach einer vorgegebenen Anordnung hineingespritzt wurden. Für das Screening wird durch den Dispenser 1 in alle Vertiefungen ein Substrat *A* zugegeben und die Platte dann zum Dispenser 2 transportiert, der in jede Vertiefung ein Enzym spritzt, dessen Aktivität unter der Wirkung des sich in der Vertiefung bereits befindenden Stoffes getestet werden soll. Die Platte wird dann in einem Wärmeschrank eine bestimmte Zeit lang aufbewahrt. Dabei katalysiert das Enzym den chemischen Umsatz des optisch inaktiven Substrats *A* in das fluoreszente Produkt *B*. Die Reaktion kann leicht über die zeitliche Zunahme des Fluoreszenzsignals, das proportional zur Konzentration des Produkts *B* ist, quantifiziert werden. Wenn das Enzym die Reaktion – wie gewünscht – inhibiert, wird weniger fluoreszentes Produkt *B* pro Zeiteinheit gebildet und der Wert des Messsignals nach einer genau festgelegten Reaktionszeit ist dementsprechend

kleiner. Der Transport der Platten vom Wärmeschrank zur Messeinrichtung muss deshalb nach einer genau festgelegten Zeit erfolgen. Die Mikrotiterplatten werden nach der Messung entsorgt.

1. Stellen Sie einen Automaten bzw. ein Petrinetz auf, das die gesteuerte Screening-Anlage beschreibt. Nehmen Sie dazu an, dass die Anlage über zwei Transportbänder verfügt und zwar eines zwischen dem Vorratsschrank und dem Dispenser 1 und eines zwischen Wärmeschrank, Messeinrichtung und den beiden Dispensern. Das Modell soll den Ablauf des oben beschriebenen Screenings mit möglichst wenigen Modellelementen wiedergeben.
2. Inwiefern kann die Anlage schneller arbeiten, wenn das in der Abbildung links dargestellte Transportband an der gezeigten Stelle geteilt wird, so dass sich der obere und der untere Teil getrennt voneinander bewegen können? Begründen Sie Ihre Antwort anhand einer entsprechenden Veränderung Ihres Modells.  $\square$

## 11.6 Kompositionale Modellbildung diskreter Systeme

### 11.6.1 Modellbildungsaufgabe

Modelle werden häufig nicht für das gesamte betrachtete System auf einmal aufgestellt, sondern man bildet zunächst die Modelle der Teilsysteme und verknüpft diese dann zum Modell des Gesamtsystems. Dieses Vorgehen soll hier für ereignisdiskrete Systeme erklärt werden, die durch deterministische E/A-Automaten beschrieben sind.

Gegeben sind die beiden Automaten

$$\mathcal{A}_1 = (\mathcal{Z}_1, \mathcal{V}_1, \mathcal{W}_1, G_1, H_1, z_{10}) \text{ und } \mathcal{A}_2 = (\mathcal{Z}_2, \mathcal{V}_2, \mathcal{W}_2, G_2, H_2, z_{20}).$$

Gesucht ist ein Automat

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, G, H, z_0),$$

der das Verhalten der Zusammenschaltung von  $\mathcal{A}_1$  und  $\mathcal{A}_2$  beschreibt. Dabei ist die Frage zu beantworten, wie sich in Abhängigkeit von der betrachteten Zusammenschaltung der beiden Automaten die Zustandsmenge  $\mathcal{Z}$  des Gesamtmodells aus den Zustandsmengen  $\mathcal{Z}_1$  und  $\mathcal{Z}_2$  der Teilautomaten zusammensetzt, wie dementsprechend  $\mathcal{V}$  aus  $\mathcal{V}_1$  und  $\mathcal{V}_2$  bzw.  $\mathcal{W}$  aus  $\mathcal{W}_1$  und  $\mathcal{W}_2$  gebildet wird und wie die Funktionen  $G$  und  $H$  aus  $G_1, G_2, H_1$  und  $H_2$  berechnet werden können.

Da der Automat  $\mathcal{A}$  die *gleichzeitige* Bewegung der Teilautomaten  $\mathcal{A}_1$  und  $\mathcal{A}_2$  beschreiben soll, sind – unabhängig von der Zusammenschaltung – im Prinzip alle Zustandskombinationen beider Automaten möglich. Die Zustandsmenge von  $\mathcal{A}$  ist folglich das kartesische Produkt der Zustandsmengen von  $\mathcal{A}_1$  und  $\mathcal{A}_2$ :

$$\mathcal{Z} = \mathcal{Z}_1 \times \mathcal{Z}_2.$$

Dies führt auf eine vektorielle Darstellung

$$z(k) = \begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}$$

für den Zustand  $z(k)$  des Gesamtautomaten. Für den Anfangszustand von  $\mathcal{A}$  gilt dementsprechend

$$z_0 = \begin{pmatrix} z_{10} \\ z_{20} \end{pmatrix}.$$

Wenn man zu einer skalaren Zustandsdarstellung zurückkehren will, muss man die in  $\mathcal{Z}_1 \times \mathcal{Z}_2$  enthaltenen Zustände entsprechend kodieren. Dies wird im Folgenden meist nicht getan, um den Zusammenhang zwischen den Zuständen des Gesamtautomaten und denen der Teilautomaten zu zeigen.

In Abhängigkeit von den dynamischen Eigenschaften der Teilautomaten und den Koppelbedingungen kann es vorkommen, dass der Gesamtautomat nicht alle Zustände  $z \in \mathcal{Z}_1 \times \mathcal{Z}_2$  annehmen kann. In diesem Fall streicht man die nicht erreichbaren Zustände aus der Zustandsmenge  $\mathcal{Z}$  heraus.

Für die Verknüpfung von E/A-Automaten gibt es drei Grundstrukturen, die als Synchronisation, Reihenschaltung und Rückführschaltung bezeichnet werden. Auf diese drei Strukturen wird im Folgenden eingegangen. Dabei wird die Zusammenschaltung von nur jeweils zwei Automaten betrachtet, weil man für Systeme mit mehr als zwei Teilsystemen den Gesamtautomaten durch schrittweise Anwendung der hier behandelten Kompositionsregeln bilden kann.

### 11.6.2 Synchronisation von Automaten

Die erste Art der Zusammenschaltung betrifft die in Abb. 11.46 gezeigte Struktur, bei der die beiden Teilautomaten auf den ersten Blick gar nichts miteinander zu tun haben. Jeder Automat schaltet für sich in Abhängigkeit von seiner Eingabe  $v_i$  und erzeugt dabei seine Ausgabe  $w_i$ .

Diese Zusammenschaltung ist jedoch insofern bedeutungsvoll, als dass sie beide Automaten zwingt, synchron zu schalten. Dafür müssen beide Eingaben gleichzeitig auftreten und gleichzeitige Zustandsübergänge in beiden Automaten hervorrufen, die mit einem gleichzeitigen Erzeugen der Ausgaben verknüpft sind. Man spricht deshalb bei dieser Zusammenschaltung von einer Synchronisation.

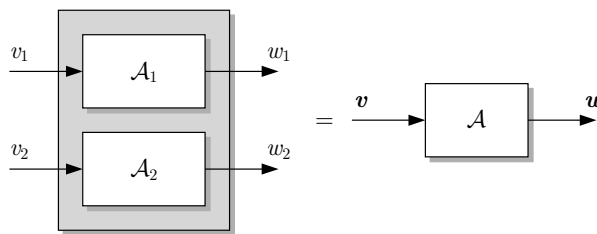


Abb. 11.46: Synchronisation zweier Automaten

Wenn nur ein Automat eine Eingabe erhält, wird die Eingabe des anderen Automaten durch ein leeres Symbol  $\varepsilon$  ergänzt, für das der betreffende Teilautomat stehen bleibt und ein leeres Ausgabesymbol generiert. In diesem Fall schaltet also nur der Teilautomat mit der „richtigen“ Eingabe.

Damit ergeben sich folgende

**Kompositionsregeln für die Synchronisation zweier Automaten:**

$$\mathcal{Z} = \mathcal{Z}_1 \times \mathcal{Z}_2$$

$$\mathcal{V} = \mathcal{V}_1 \times \mathcal{V}_2$$

$$\mathcal{W} = \mathcal{W}_1 \times \mathcal{W}_2$$

$$G \left( \begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}, \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix} \right) = \begin{pmatrix} G_1(z_1(k), v_1(k)) \\ G_2(z_2(k), v_2(k)) \end{pmatrix} \quad (11.66)$$

$$H \left( \begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}, \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix} \right) = \begin{pmatrix} H_1(z_1(k), v_1(k)) \\ H_2(z_2(k), v_2(k)) \end{pmatrix}. \quad (11.67)$$

Die Zustandsübergangsfunktion  $G$  und die Ausgabefunktion  $H$  des Gesamtautomaten setzen sich aus den beiden Funktionen der Teilautomaten zusammen, ohne dass sich diese gegenseitig beeinflussen. Da entsprechend den Gln. (11.13), (11.14) die Teilautomaten auf leere Eingaben nicht reagieren, gelten die o. a. Kompositionsregeln auch dann, wenn nur einer der beiden Teilautomaten eine Eingabe erhält.

**Beispiel 11.12** *Synchronisation zweier Automaten*

Es wird die Synchronisation der in Abb. 11.47 links gezeigten Automaten betrachtet, deren Eingaben, Zustände und Ausgaben binär sind. Der auf der rechten Seite gezeigte Automat entsteht durch die Anwendung der Gln. (11.66), (11.67), wobei nur diejenigen Kanten dargestellt sind, die zum gleichzeitigen Eintreffen einer Eingabe an beide Teilautomaten gehören. Wenn nur ein Teilautomat eine Eingabe erhält, schaltet nur der betreffende Teilautomat, was auf zusätzliche Kanten im Automatengraphen führt, die hier der Übersichtlichkeit halber weggelassen wurden. Die Zustände des Gesamtautomaten sind so angeordnet, dass eine horizontale Zustandsänderung den Automaten  $\mathcal{A}_1$  und eine senkrechte Bewegung den Automaten  $\mathcal{A}_2$  betrifft.

Die ganz links dargestellte Kante kommt folgendermaßen zustande. Der Automat befindet sich zur Zeit  $k$  im Zustand

$$\mathbf{z}(k) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \text{also in } z_1(k) = 0, z_2(k) = 0$$

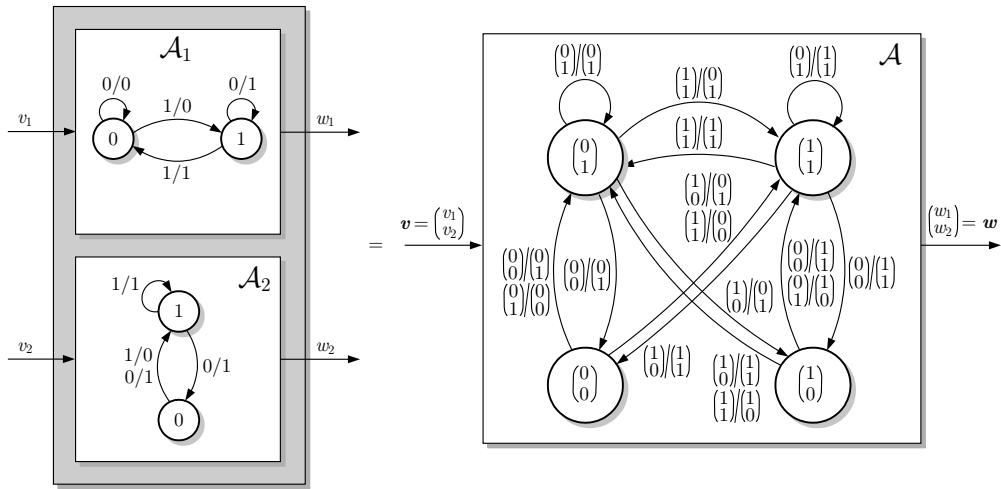
und erhält die Eingabe

$$\mathbf{v}(k) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \text{also } v_1(k) = 0, v_2(k) = 0.$$

Aus Gl. (11.66) erhält man dafür

$$\mathbf{z}(k+1) = \begin{pmatrix} z_1(k+1) \\ z_2(k+1) \end{pmatrix} = \begin{pmatrix} G_1(0, 0) \\ G_2(0, 0) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

und aus Gl. (11.67)

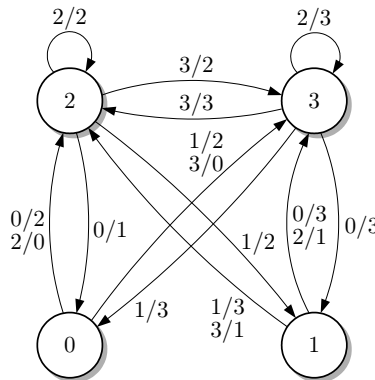


**Abb. 11.47:** Beispiel für die Synchronisation zweier Automaten

$$\mathbf{w}(k) = \begin{pmatrix} w_1(k) \\ w_2(k) \end{pmatrix} = \begin{pmatrix} H_1(0, 0) \\ H_2(0, 0) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Die hier eingesetzten Werte der Funktionen  $G_1$ ,  $G_2$ ,  $H_1$  und  $H_2$  für die angegebenen Argumente kann man aus den Automatengraphen der beiden links gezeichneten Teilautomaten ablesen.

Wenn der Automat im selben Ausgangszustand ist, aber die Eingabe  $\mathbf{v}(k) = (0 \ 1)^T$  erscheint, geht er in denselben Nachfolgezustand über und erzeugt die Ausgabe  $\mathbf{w}(k) = (0 \ 0)^T$ . Deshalb gilt die betrachtete Kante für zwei unterschiedliche E/A-Paare. Der Gesamtautomat entsteht, indem man für alle vier Zustände jeweils alle vier möglichen Kombinationen von Eingaben untersucht.



**Abb. 11.48:** Vereinfachte Darstellung der synchronisierten Automaten

Der erhaltene Automat kann einfacher dargestellt werden, wenn man die vier Zustände, Eingaben und Ausgaben durchnummeriert und damit wieder zu einer skalaren Darstellung zurückkehrt. Hier



wurde eine Kodierung verwendet, bei der die vektoriellen Größen als zweistellige Binärzahlen mit dem ersten Vektorelement als rechte Stelle und dem zweiten Vektorelement als linke Stelle interpretiert wurden, wobei für die Zustände folgende Korrespondenzen gelten:

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \rightarrow 0, \quad \begin{pmatrix} 1 \\ 0 \end{pmatrix} \rightarrow 1, \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \rightarrow 2, \quad \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rightarrow 3.$$

Dadurch entsteht der in Abb. 11.48 gezeigte Automat, der einfacher zu lesen ist, bei dem allerdings die Zusammenhänge zu den gegebenen Teilautomaten nicht mehr explizit zu sehen sind.  $\square$

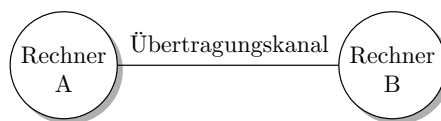
### Beispiel 11.13 Beschreibung einer Rechnerkommunikation

Zwei Rechner kommunizieren über einen Bus (Abb. 11.49). Dabei erhalten die Rechner sporadisch Informationen, die über den Bus zu dem anderen Rechner zu übertragen sind. Für die Rechner kann man deshalb folgende Zustände definieren:

Zustand	Beschreibung
$I$ (idle)	Der Rechner will keine Informationen übertragen.
$W$ (waiting)	Der Rechner hat eine Information, die er übertragen will.
$T$ (transmitting)	Der Rechner sendet seine Information über den Bus an den anderen Rechner.

Die Zustandsübergangsfunktion  $G$  jedes Automaten beschreibt folgende Zustandsübergänge:

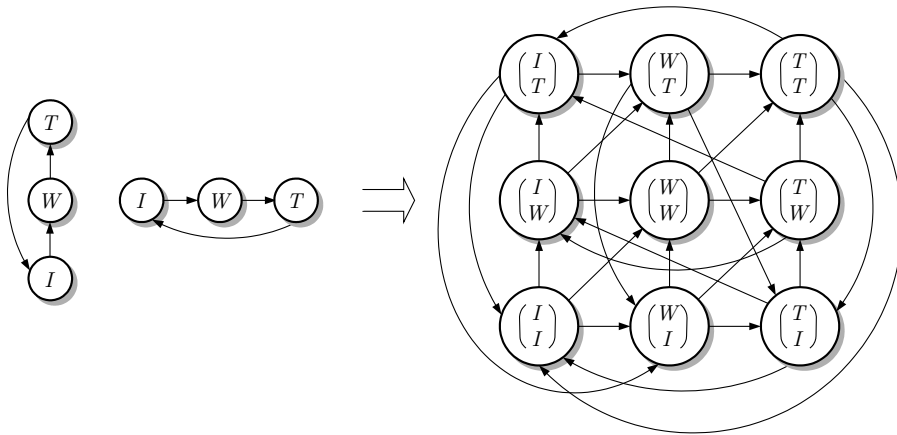
Zustandsübergang	Beschreibung
$I \rightarrow W$	Der Rechner stellt die zu übertragende Information bereit.
$W \rightarrow T$	Der Rechner erhält die Freigabe zur Übertragung.
$T \rightarrow I$	Die Übertragung ist beendet.



**Abb. 11.49:** Kommunikation zweier Rechner über einen Übertragungskanal

Abbildung 11.50 zeigt links die beiden Automaten, von denen jeder einen Rechner beschreibt. Jeder Rechner kann für sich die Zustände  $I$ ,  $W$  und  $T$  zyklisch durchlaufen. Wenn die Rechner nicht von außen in ihrem Verhalten beeinflusst werden, sind sie autonome Systeme, die ihre Zustandsübergänge zu beliebigen Zeiten ausführen. Das Verhalten des Gesamtsystems entspricht dann der Synchronisation beider Automaten.

Um die Komponenten des Gesamtzustands den einzelnen Rechnern besser zuordnen zu können, wurde im linken Teil der Abbildung der Automatengraph für den Rechner 2 in senkrechter Richtung



**Abb. 11.50:** Beschreibung der ungesteuerten Rechnerkommunikation durch Automaten

und der für den Rechner 1 in waagerechter Richtung angeordnet. Im Gesamtautomat beschreibt die erste Komponente des als Vektor notierten Zustands den Zustand des Rechners 1 und die zweite Komponente den des Rechners 2. Eine waagerechte Kante stellt deshalb eine Zustandsänderung des Rechners 1, eine senkrechte Kante eine Zustandsänderung des Rechners 2 dar. Bei „schrägen“ Kanten ändern sich beide Zustände.

Der Automatengraph zeigt, welche Zustandsänderungen auftreten können, wenn die beiden Rechner den Bus in unkoordinierter Weise nutzen wollen und dabei einzelne oder gemeinsame Zustandsänderungen ausführen. Es kann insbesondere der Zustand  $(T \ T)^T$  auftreten, bei dem beide Rechner den Bus zur Übertragung ihrer Daten nutzen. Dies muss verhindert werden, weil in diesem Zustand keine geordnete Kommunikation möglich ist. Die durch das Protokoll vorgeschriebene Koordination der Aktivitäten beider Rechner, die bei diesem Modell nicht berücksichtigt wird, verhindert, dass dieser Zustand eintreten kann.  $\square$

**Aufgabe 11.18\*** *Synchronisierte Darstellung der Steuerstrecke einer Rolltreppensteuerung*

Eine Rolltreppensteuerung soll die Rolltreppe anschalten, wenn eine Lichtschranke einen Impuls ausstet, der die Ankunft einer Person darstellt, und die Treppe ausschalten, wenn sie nicht mehr benutzt wird. Die Steuerung reagiert damit auf die Treppe einerseits, die die diskreten Eingaben „Motor an“ und „Motor aus“ von der Steuerung erhält und mit einem Impuls reagiert, wenn eine Person die Treppe benutzen will. Andererseits verwendet die Steuerung eine interne Uhr, die mit der Eingabe „Uhr an“ eingeschaltet bzw. zurückgesetzt wird und die nach Ablauf einer vorgegebenen Zeitspanne mit der Ausgabe „Uhr aus“ reagiert.

Aus der Sicht der Steuerung kann die Uhr für Analyse- und Entwurfszwecke mit zur Steuerstrecke gezählt werden, denn Treppe und Uhr gemeinsam erzeugen die Eingaben für den Steuerungsalgorithmus und beide erhalten vom Steuerungsalgorithmus ihre Eingaben. Dies bedeutet eine Verknüpfung beider Teilsysteme im Sinne der Synchronisation.

Welche Eingaben, Zustände und Ausgaben hat die Synchronisation der durch Automaten beschriebenen Treppe und Uhr? Wie sieht der Automatengraph aus?  $\square$

### 11.6.3 Reihenschaltung

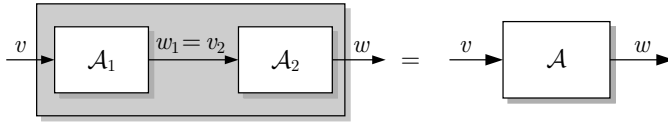
In der in Abb. 11.51 gezeigten Reihenschaltung erhält der Automat  $\mathcal{A}_2$  seine Eingabe vom Automaten  $\mathcal{A}_1$

$$w_1(k) = v_2(k)$$

und für die Eingabe und Ausgabe der Reihenschaltung gilt

$$v(k) = v_1(k)$$

$$w(k) = w_2(k).$$



**Abb. 11.51:** Reihenschaltung zweier Automaten

Damit die Reihenschaltung definiert ist, muss die Beziehung

$$\mathcal{V}_2 \supseteq \mathcal{W}_1$$

für die Wertebereiche der Koppelgrößen gelten.

Die Reihenschaltung wird zwar so interpretiert, dass beide Automaten gleichzeitig in Abhängigkeit von ihren Eingaben schalten. Da die Eingabe des Automaten  $\mathcal{A}_2$  jedoch durch die Ausgabe des Automaten  $\mathcal{A}_1$  bestimmt wird, muss man bei der Berechnung der Zustandsübergänge in Signalrichtung vorgehen und zunächst den neuen Zustand und die neue Ausgabe des linken Automaten ermitteln, bevor man den Zustandsübergang des rechten Automaten bestimmt.

#### Kompositionsregeln für die Reihenschaltung:

$$\mathcal{Z} = \mathcal{Z}_1 \times \mathcal{Z}_2$$

$$\mathcal{V} = \mathcal{V}_1$$

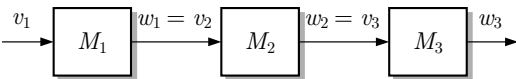
$$\mathcal{W} = \mathcal{W}_2$$

$$G \left( \begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}, v(k) \right) = \begin{pmatrix} G_1(z_1(k), v(k)) \\ G_2(z_2(k), H_1(z_1(k), v(k))) \end{pmatrix} \quad (11.68)$$

$$H \left( \begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}, v(k) \right) = H_2(z_2(k), H_1(z_1(k), v(k))). \quad (11.69)$$

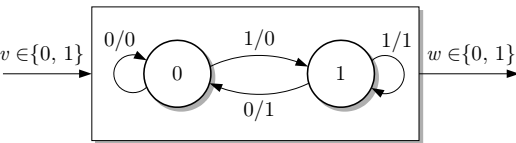
**Beispiel 11.14**    *Kompositionale Modellbildung von drei Werkzeugmaschinen*

Es wird eine Fertigung betrachtet, bei der die Werkstücke drei Werkzeugmaschinen nacheinander durchlaufen und dort bearbeitet werden (Abb. 11.52). Die Werkstücke werden von einem Fließband zur Maschine  $M_1$  befördert, dort bearbeitet und nach Abschluss der Bearbeitung unmittelbar zur Maschine  $M_2$  transportiert. Anschließend werden sie in der Maschine  $M_3$  bearbeitet und schließlich zum Abtransport auf ein zweites Fließband platziert.



**Abb. 11.52:** Reihenschaltung von drei Werkzeugmaschinen

Da zwischen den Maschinen kein Lager angeordnet ist, muss der Werkstückwechsel synchron erfolgen. Der Takt wird also durch die am längsten arbeitende Maschine vorgegeben. Im Folgenden soll nur zwischen der innerhalb des betrachteten Taktes arbeitenden Maschine und der wartenden Maschine unterschieden werden, wobei der Zustand 0 bedeutet, dass sich in der Maschine für einen Takt lang kein Werkstück befindet.



**Abb. 11.53:** Modell der einzelnen Werkzeugmaschine

Das Modell des Gesamtsystems wird im Folgenden in zwei Schritten aufgestellt, wobei zuerst die drei Modelle der einzelnen Maschinen gebildet und dann die Reihenschaltung der drei Modelle berechnet wird. Die drei Maschinen  $M_1$ ,  $M_2$  und  $M_3$  sind durch gleichartige Modelle beschrieben, die folgende Zustände besitzen:

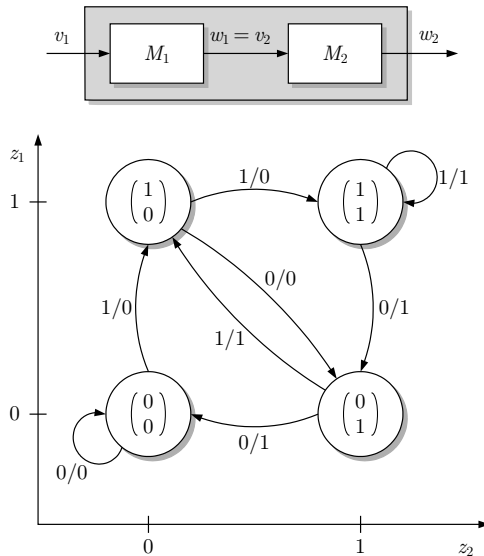
$z$	Bedeutung
0	Die Maschine wartet.
1	Die Maschine arbeitet.

Die Eingaben und Ausgaben haben folgende Wertebereiche:

$v$	Bedeutung
0	Die Maschine erhält kein neues Werkstück.
1	Die Maschine erhält ein neues Werkstück.

$w$	Bedeutung
0	Die Maschine gibt kein Werkstück aus.
1	Die Maschine gibt ein bearbeitetes Werkstück aus.



**Abb. 11.54:** Modell der Reihenschaltung der Maschinen  $M_1$  und  $M_2$

Mit diesen Vereinbarungen erhält man den in Abb. 11.53 gezeigten Automaten.

Für die Bildung des Gesamtmodells werden im ersten Schritt die Modelle der Maschinen  $M_1$  und  $M_2$  kombiniert. Der Zustand  $\mathbf{z}$  der Reihenschaltung ist ein Vektor mit den Zuständen  $z_1$  und  $z_2$  der beiden Maschinen  $\mathbf{z} = (z_1, z_2)^T$ . Zur Veranschaulichung sind die Zustände im Automatengraphen so angeordnet, dass sich der Zustand der Maschine  $M_1$  nach oben und der der Maschine  $M_2$  nach rechts verändert (Abb. 11.54). Der Gesamtautomat hat den Eingang  $v_1$  und den Ausgang  $w_2$ . Seine Zustandsübergangsfunktion  $G$  erhält man durch Anwendung der Kompositionsregel (11.68).

Um das Modell für die drei Maschinen zu erhalten, muss der in Abb. 11.54 gezeigte Automat in einer Reihenschaltung mit dem in Abb. 11.53 gezeigten Automaten kombiniert werden (Abb. 11.55). Der Zustand der Zusammenschaltung wird durch einen dreielementigen Vektor  $\mathbf{z} = (z_1, z_2, z_3)^T$  dargestellt, wobei die oberen beiden Elemente den Zustand des in Abb. 11.54 gezeigten Automaten repräsentieren. Die Eingangsgröße des Gesamtsystems ist  $v = v_1$ , die Ausgangsgröße  $w = w_3$ . Die Zustandsübergangsfunktion des Gesamtautomaten erhält man durch erneute Anwendung der Kompositionsregel für die Reihenschaltung.  $\square$

#### **Aufgabe 11.19\*** Verallgemeinerung der Reihenschaltung

Abbildung 11.56 zeigt eine verallgemeinerte Reihenschaltung, in der drei Automaten miteinander verknüpft sind. Stellen Sie in Verallgemeinerung der Überlegungen aus dem Abschn. 11.6.3 die Kompositionsregeln für den Gesamtautomaten auf.  $\square$

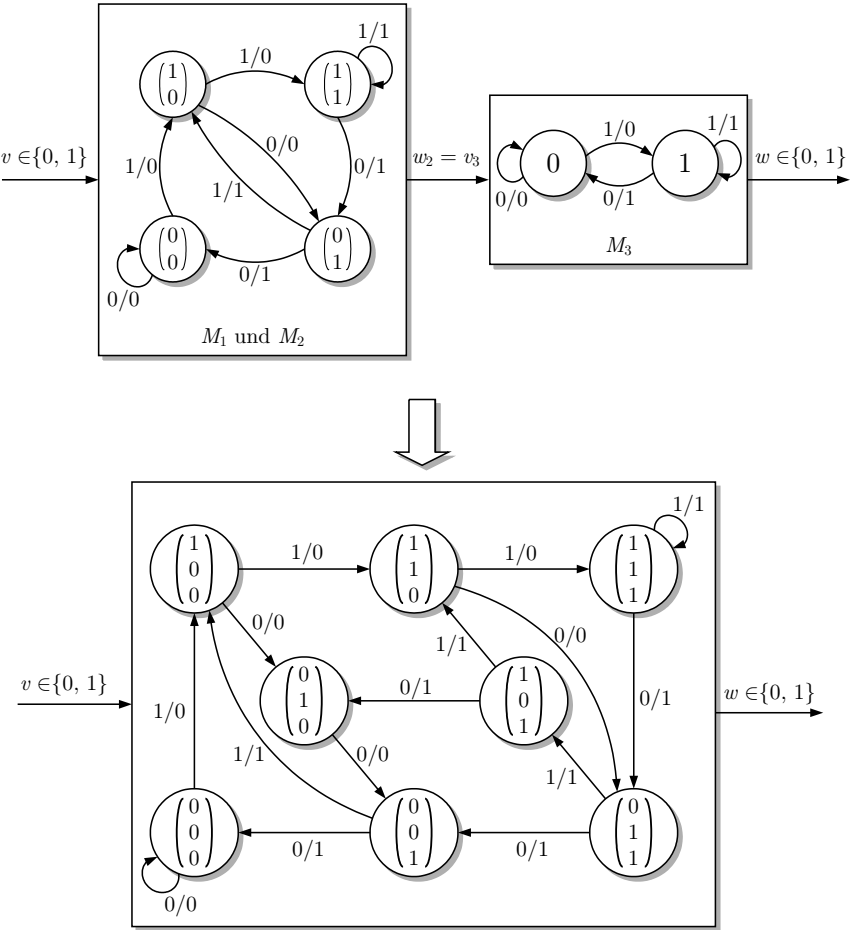


Abb. 11.55: Modell der Reihenschaltung der Maschinen  $M_1$ ,  $M_2$  und  $M_3$

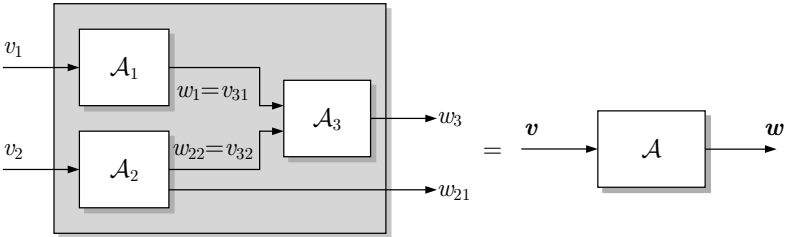


Abb. 11.56: Reihenschaltung von drei Automaten

### 11.6.4 Rückführautomat

Die dritte Kopplungsmöglichkeit führt auf Automaten, deren Eingabe von der eigenen Ausgabe abhängt. Abbildung 11.57 zeigt die einfachste Struktur eines Rückkopplungsautomaten. Wie aus der Abbildung zu erkennen ist, führt die Rückkopplung des Automaten

$$\mathcal{A}_1 = (\mathcal{Z}_1, \mathcal{V}_1, \mathcal{W}_1, G_1, H_1, z_{10})$$

zu einem autonomen Automaten

$$\mathcal{A} = (\mathcal{Z}, G, z_0).$$

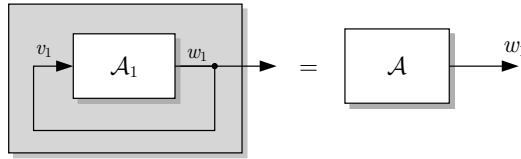
Bezüglich der Zustandsmengen der beiden Automaten gilt erwartungsgemäß

$$\mathcal{Z} = \mathcal{Z}_1,$$

wobei gegebenenfalls Zustände, die im Rückführautomaten nicht erreichbar sind, gestrichen werden können. Aus der Abbildung ist auch zu sehen, dass der rückgeführte Automat nur dann existiert, wenn die Bedingung

$$\mathcal{V}_1 \supseteq \mathcal{W}_1$$

gilt, denn nur unter dieser Bedingung ist der Automat für alle seine Ausgaben, die er als Eingaben erhält, definiert.



**Abb. 11.57:** Automat mit Rückführung

Da der Automat als Eingabe seine eigene Ausgabe erhält ( $v_1(k) = w_1(k)$ ), führt der rückgeführte Automat im Zeitschritt  $k$  einen Zustandsübergang  $z(k) \rightarrow z(k+1)$  aus, bei dem er die Ausgabe

$$w_1(k) = H_1(z_1(k), v_1(k)) \quad (11.70)$$

erzeugt. Diese Ausgabe muss mit  $v_1(k)$  übereinstimmen, weil die Rückführung keine Zeitverzögerung beinhaltet und die Ausgabe zum selben Zeitpunkt als Eingabe des Automaten wirkt. Nur wenn Gl. (11.70) für  $v_1(k) = w_1(k)$  eine Lösung in  $w_1(k)$  besitzt und wenn diese Lösung eindeutig ist, ist der rückgeführte Automat eindeutig definiert. Diese Bedingung muss für alle Zustände erfüllt sein. Ob diese Lösung existiert, hängt von der Funktion  $H_1$ , nicht aber von der Zeit  $k$  ab, so dass Gl. (11.70) einfacher als

$$w_1 = H_1(z_1, w_1) \quad (11.71)$$

geschrieben werden kann.

Der Rückführautomat heißt *wohldefiniert*, wenn es für *alle* Zustände  $z_1 \in \mathcal{Z}_1$  *genau eine* Lösung  $w_1 \in \mathcal{W}_1$  der Gl. (11.71) gibt.

Für wohldefinierte Automaten kann man die Lösung  $w_1$  von Gl. (11.71) in Abhängigkeit von  $z_1$  durch eine Funktion  $H$  ausdrücken:

$$w_1 = H(z_1). \quad (11.72)$$

Damit erhält man die folgenden

**Kompositionsregeln für den wohldefinierten Rückführautomaten:**

$$\begin{aligned} \mathcal{Z} &= \mathcal{Z}_1 \\ G(z(k)) &= G_1(z(k), H(z(k))) \quad \text{mit } H \text{ aus Gl. (11.72).} \end{aligned} \quad (11.73)$$

Die Tatsache, dass die Ausgabe  $w(k)$  über die Ausgabefunktion  $H_1$  von sich selbst abhängt, beschreibt dasselbe Phänomen wie algebraische Schleifen in kontinuierlichen Systemen (vgl. S. 119). In beiden Fällen wirkt der Systemausgang ohne Verzögerung auf den Eingang zurück *und* die Eingangsgröße hat einen unmittelbaren Einfluss auf die Ausgangsgröße. Die zweite Voraussetzung ist bei kontinuierlichen Systemen nur dann erfüllt, wenn das System sprunghaft ist. Bei diskreten Systemen ist dies bei allen Mealy-Automaten der Fall.

Bei linearen kontinuierlichen Systemen konnte für die Lösung von Gl. (3.64), die den zu Gl. (11.71) analogen Sachverhalt beschreibt, eine Existenzbedingung angegeben und die Lösung explizit dargestellt werden. Dies ist bei diskreten Systemen nicht möglich. Hier muss man für jeden Zustand  $z$  einzeln überprüfen, ob es in der Automatentabelle genau eine Zeile mit diesem Zustand gibt, in der  $v$  und  $w$  denselben Wert haben. Wenn dies der Fall ist, gibt der in dieser Zeile stehende Nachfolgezustand  $z'$  den Zustand an, in den der rückgeführte Automat übergeht. Wenn es keine solche Zeile oder mehrere solche Zeilen gibt, ist der rückgeführte Automat nicht wohldefiniert.

Wenn es sich beim Automaten  $\mathcal{A}_1$  um einen Moore-Automaten handelt, so geht die Eingabe  $v$  nicht in die Funktion  $H_1$  ein. Dann ist die Ausgabe dem aktuellen Zustand  $z$  zugeordnet

$$w(k) = H(z(k))$$

und der rückgeführte Automat existiert und führt den zu  $v(k) = w(k)$  gehörenden Zustandsübergang

$$G(z(k)) = G_1(z(k), H_1(z(k)))$$

aus.

#### **Beispiel 11.15** Beschreibung der Rolltreppensteuerung durch einen Rückführautomat

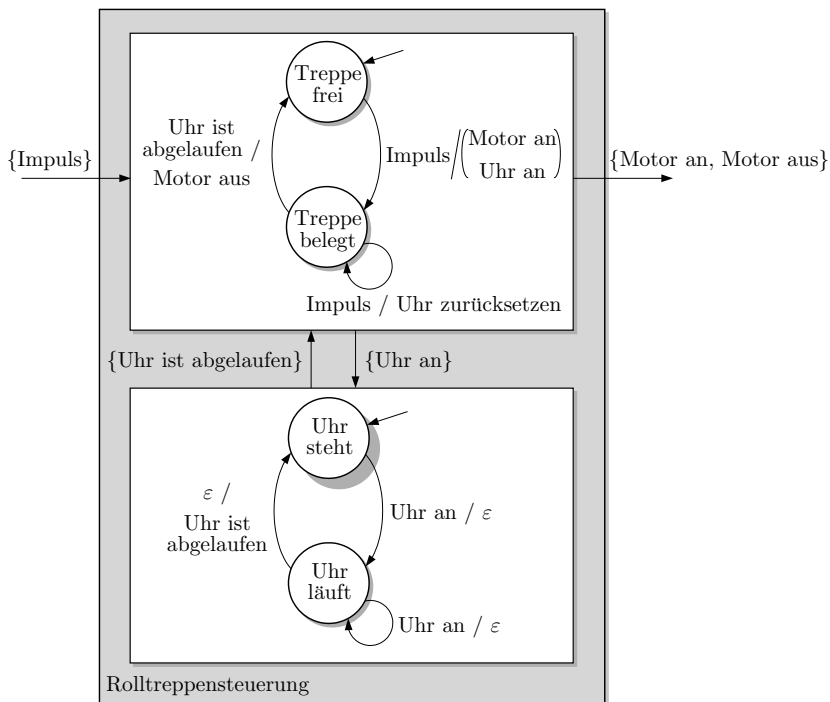
Die Steuerung einer Rolltreppe hat die Aufgabe, die Rolltreppe anzuschalten, wenn der von einer Lichtschranke ausgesendete Impuls anzeigt, dass eine Person die Rolltreppe benutzen will. Die Rolltreppe soll abgeschaltet werden, wenn sie zeitweise nicht benutzt wird. Als Blockschaltbild dargestellt ist die



**Abb. 11.58:** Rolltreppesteuerung

Steuerung ein diskretes System, das die Impulse von der Lichtschranke als Eingaben erhält und mit „Motor an“ oder „Motor aus“ antwortet (Abb. 11.58).

Die Treppensteuerung besteht aus zwei Komponenten. Die erste Komponente, die den eigentlichen Steuerungsalgorithmus enthält, schaltet die ausgeschaltete Treppe ein, wenn die Lichtschranke einen Impuls abgibt. Gleichzeitig schaltet die Steuerung eine Uhr ein. Wenn die Treppe bereits eingeschaltet ist und die Lichtschranke einen erneuten Impuls erzeugt, wird die Uhr auf die Anfangszeit zurückgesetzt, von der aus sie rückwärts läuft. Wenn die Uhr abgelaufen ist, schaltet die Steuerung die Treppe aus.

**Abb. 11.59:** Komposition der Rolltreppesteuerung aus dem Steuerungsalgorithmus und der Uhr

Die Uhr, die in der Steuerung enthalten ist, soll den Zeitpunkt feststellen, ab dem die Treppe frei ist. Bei jedem Impuls der Lichtschranke wird sie auf eine Anfangszeit gesetzt, die so bemessen ist, dass sich nach Ablauf der Uhr niemand mehr auf der Treppe befinden kann. Da es nicht auf die genaue Uhrzeit,

sondern nur darauf ankommt, ob die Uhr abgelaufen ist oder nicht, wird die Uhr als ereignisdiskretes System aufgefasst. Wenn die Uhr abgelaufen ist, erzeugt sie die Ausgabe „Uhr ist abgelaufen“.

**Modellbildung.** Im Folgenden wird das in Abb. 11.59 gezeigte Modell der Rolltreppensteuerung aufgestellt, wobei zunächst der Steuerungsalgorithmus und die Uhr als getrennte Komponenten beschrieben und dann beide Komponenten zum Gesamtmodell zusammengesetzt werden. Der im oberen Teil der Abbildung dargestellte Steuerungsalgorithmus hat zwei Zustände

$$z_1 \in \{\text{Treppe frei, Treppe belegt}\},$$

die anzeigen, ob die Treppe gerade benutzt wird, wobei der Zustand „Treppe frei“ den Anfangszustand  $z_{10}$  darstellt. Er erhält die Eingabe „Impuls“ von der Lichtschranke und die Eingabe „Uhr ist abgelaufen“ von der Uhr. Wenn beide Eingaben gleichzeitig erscheinen, ignoriert der Steuerungsalgorithmus die Eingabe von der Uhr. Beide Eingaben können deshalb zu einer Eingabe zusammengefasst werden:

$$v_1 \in \{\text{Impuls, Uhr ist abgelaufen}\}.$$

Durch seine Ausgabe schaltet der Steuerungsalgorithmus den Motor und die Uhr:

$$w_1 \in \left\{ \begin{pmatrix} \text{Motor an} \\ \text{Uhr an} \end{pmatrix}, \text{Uhr an, Uhr zurücksetzen, Motor aus} \right\}.$$

Die beschriebene Funktion des Steuerungsalgorithmus führt auf den im oberen Teil der Abbildung gezeigten Automatengraphen. Wenn die Treppe frei ist, ist der Treppenmotor abgeschaltet. Erscheint als Eingabe für die Steuerung ein Impuls, so schaltet die Steuerung den Motor an und startet die Uhr. Gleichzeitig speichert die Steuerung, dass die Treppe jetzt belegt ist, indem sie in den Zustand „Treppe belegt“ übergeht. Wenn die Lichtschranke einen Impuls erzeugt während die Steuerung im Zustand „Treppe belegt“ ist, wird die Uhr zurückgesetzt, aber keine den Motor betreffende Ausgabe erzeugt. Wenn die Uhr abgelaufen ist, wird der Motor ausgeschaltet und die Steuerung geht in den Zustand „Treppe frei“ über.

Die Uhr hat zwei Zustände

$$z_2 \in \{\text{Uhr steht, Uhr läuft}\}$$

mit dem Anfangszustand  $z_{20} = \text{„Uhr steht“}$ , die Eingabe

$$v_2 \in \{\text{Uhr an}\}$$

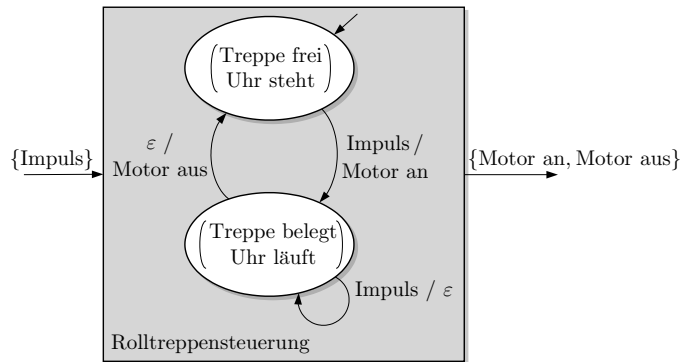
und die Ausgabe

$$w_2 \in \{\text{Uhr ist abgelaufen}\}.$$

Ihr Verhalten wird durch den im unteren Teil der Abbildung gezeigten Automatengraphen dargestellt. Das leere Symbol  $\varepsilon$  zeigt, dass die Uhr beim Anschalten oder Zurücksetzen keine Ausgabe erzeugt bzw. dass die Ausgabe „Uhr ist abgelaufen“ nicht durch eine Eingabe initiiert, sondern nach dem Ablaufen der Uhr spontan erzeugt wird. Deshalb ist dieser Automat ein Moore-Automat.

**Bildung des Rückführautomaten.** Die beiden beschriebenen Automaten sind in einer Rückkopplungsschaltung miteinander verknüpft, wobei die Koppelsignale die Eingabe und die Ausgabe der Uhr betreffen. Der Rückführautomat ist wohldefiniert, weil beide Automaten Moore-Automaten sind. Das Ergebnis ist in Abb. 11.60 zu sehen. Von den vier möglichen Zuständen des kartesischen Produkts der Zustandsmengen der beiden Teilautomaten treten nur die zwei Zustände

$$\begin{pmatrix} \text{Treppe frei} \\ \text{Uhr steht} \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} \text{Treppe belegt} \\ \text{Uhr läuft} \end{pmatrix}$$



**Abb. 11.60:** Rückführautomat, der die Rolltreppesteuerung beschreibt

auf, weil vom Anfangszustand „Treppe frei“ der Steuerung und vom Anfangszustand „Uhr aus“ der Uhr die beiden anderen Zustände

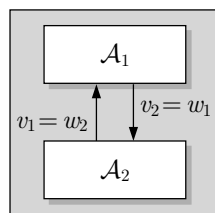
$$\begin{pmatrix} \text{Treppe frei} \\ \text{Uhr läuft} \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} \text{Treppe belegt} \\ \text{Uhr aus} \end{pmatrix}$$

nicht erreichbar sind.

Bei der Anwendung der Kompositionsregel (11.73) muss man beachten, dass hier ein etwas allgemeinerer Rückführautomat betrachtet wird als in Abb. 11.57. Die Eingabe „Impuls“ und die Ausgabe zum Motor bleiben für den Rückführautomat als Eingabe bzw. Ausgabe bestehen. Gleichung (11.73) ist erfüllt, wenn man diese Eingabe bzw. Ausgaben aus dem Steuerungsalgorithmus streicht.  $\square$

**Aufgabe 11.20\*** Rückkopplung mit mehreren Automaten

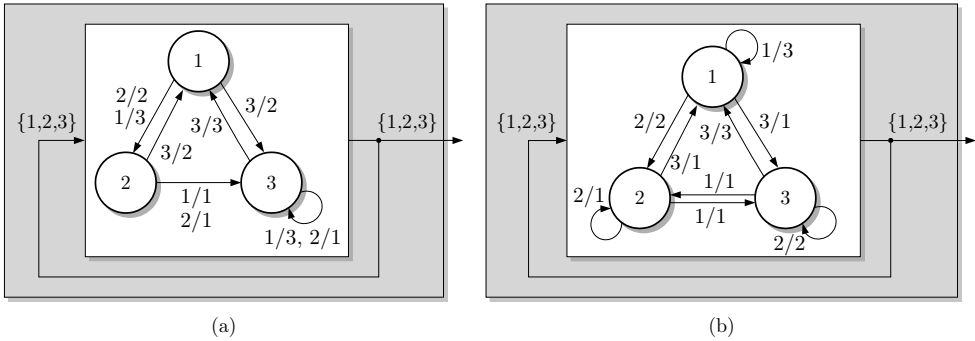
Abbildung 11.61 zeigt ein System, das aus zwei rückgekoppelten Automaten besteht. Da in der angegebenen Schaltung keine Eingaben von außen wirken, entsteht ein autonomer Automat. Unter welchen Bedingungen ist dieser Automat wohldefiniert und welche Zustandsübergangsfunktion hat er?  $\square$



**Abb. 11.61:** Zwei Automaten in Rückführschaltung

**Aufgabe 11.21\***    *Berechnung zweier Rückführautomaten*

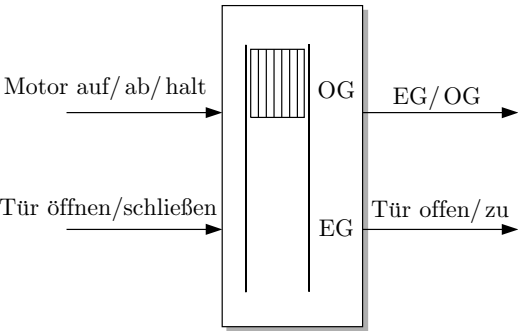
Abbildung 11.62 zeigt zwei rückgekoppelte Automaten. Überprüfen Sie, ob die Automaten wohldefiniert sind und bilden Sie die Rückführautomaten, sofern diese existieren. Geben Sie die Folge der Ausgaben an, die die Rückführautomaten mit dem Anfangszustand  $z_0 = 2$  erzeugen. □



**Abb. 11.62:** Zwei rückgekoppelte Automaten

**Aufgabe 11.22\***    *Beschreibung eines Personenaufzugs durch gekoppelte Automaten*

Der in Abb. 11.63 gezeigte Personenaufzug bewegt sich zwischen dem Erdgeschoss (EG) und dem ersten Obergeschoss (OG) hin und her. Für die Modellbildung sind dabei zwei Prozesse interessant. Einerseits fährt der Fahrkorb in Abhängigkeit von der den Motor betreffenden Eingangsgröße nach oben und unten. Dabei soll zwischen vier Zuständen unterschieden werden, die die Position des Fahrkorbs im Erdgeschoss bzw. Obergeschoss und das Hinauf- bzw. Hinunterfahren beschreiben. Andererseits kann die Tür durch Ansteuerung des entsprechenden Motors geöffnet und geschlossen werden, wobei nur zwischen den Zuständen „geöffnete Tür“ und „geschlossene Tür“ unterschieden wird.

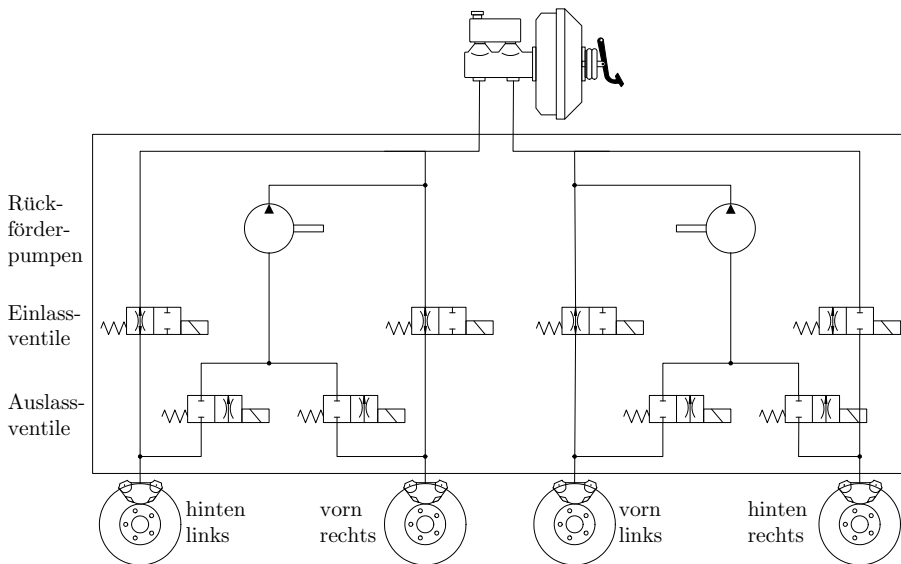


**Abb. 11.63:** Blockschaltbild eines Personenaufzugs

1. Beschreiben Sie das Verhalten des Fahrkorbs und der Tür durch zwei Automaten.
2. Um falsche Steuereingriffe wie z. B. das Bewegen des Fahrkorbs bei geöffneter Tür ausschließen zu können, müssen beide Prozesse gemeinsam betrachtet werden. Bilden Sie deshalb ein Gesamtmodell des Personenaufzugs, markieren die verbotenen Zustände und lesen an den zu diesen Zuständen führenden Kanten die verbotenen Steuereingriffe ab.
3. Beschreiben Sie den Personenaufzug durch ein Petrinetz.
4. Vergleichen Sie die erhaltenen Modelle. □

**Aufgabe 11.23** Diskrete Beschreibung eines Antiblockiersystems

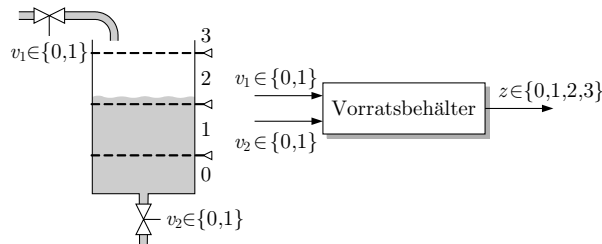
Gegenwärtig in Fahrzeugen eingebaute Antiblockiersysteme (ABS) sind hydraulisch realisiert, wie es in Abb. 11.64 gezeigt ist. Der Schlupf wird nicht kontinuierlich eingestellt, wie es mit der im Beispiel 7.7 beschriebenen Regelung möglich ist, sondern es wird mit diskreten Eingriffen gearbeitet, die die Einlass- und Auslassventile sowie die Rückförderpumpe schalten. Wenn ein Rad blockiert, was mit Hilfe eines Drehzahlsensors erkannt wird, wird das Einlassventil geschlossen, das Auslassventil geöffnet und der Bremsdruck durch die Rückförderpumpe verkleinert, bis sich das Rad wieder dreht. Dann wird der Bremsdruck erneut aufgebaut, indem das Auslassventil geschlossen, das Einlassventil geöffnet und der über das Bremspedal und den Bremskraftverstärker erzeugte Öldruck auf die Bremsen durchgeschaltet wird.


**Abb. 11.64:** Hydraulischer Schaltplan eines Antiblockiersystems

1. Zeichnen Sie ein Blockschaltbild, bei dem ein Block das Fahrzeug und der andere das Steuergerät des ABS darstellen. Welche Kopplungssignale treten auf und welche Wertebereiche haben diese?
2. Bilden Sie zwei Automaten, von denen der eine das Verhalten des Fahrzeugs und der andere das Verhalten der Steuerung beschreibt, wobei Sie zur Vereinfachung nur eines der beiden über eine gemeinsame Rückförderpumpe verbundenen Radpaare betrachten.
3. Bilden Sie das Modell des Gesamtsystems durch Verkopplung der Automaten. □

**Aufgabe 11.24\*** *Compositionale Modellbildung eines geregelten Vorratsbehälters*

Der in Abb. 11.65 gezeigte Vorratsbehälter hat eine diskrete Füllstandmessung, die einen der vier diskreten Werte 0, 1, 2 oder 3 ausgibt. Die Ventile für Zu- und Abfluss können nur vollständig geöffnet oder geschlossen werden. Eine Steuerung soll erreichen, dass der Füllstand unabhängig von der Entnahme  $v_2$  in den diskreten Bereichen 1 und 2 liegt. Es wird dabei angenommen, dass bei geöffneten Ventilen die zufließende Flüssigkeitsmenge größer als die ausfließende Menge ist.



**Abb. 11.65:** Vorratsbehälter mit diskreten Eingangsgrößen

1. Beschreiben Sie den Vorratsbehälter durch einen Automaten.
2. Es wird folgende Regelung verwendet, um den Füllstand im vorgegebenen Bereich zu halten:

$v_1$	$z$
1	0
1	1
0	2
0	3

Bestimmen Sie den Automaten, der den geregelten Vorratsbehälter beschreibt.

3. Überprüfen Sie durch eine Analyse des Automatengraphen, ob das Steuerungsziel erfüllt wird.
4. Welche Zustandsfolge ergibt sich für die Störung

$$V_2 = (1, 0, 1, 1, 0, 0),$$

wenn sich der Behälter zur Zeit  $k = 0$  im Zustand 2 befindet? □

## Literaturhinweise

Automaten werden in mehreren Fachgebieten in sehr unterschiedlicher Weise eingesetzt, so dass sich die in der entsprechenden Literatur herausgestellten Merkmale und Eigenschaften von Automaten wesentlich voneinander unterscheiden. In der mathematischen Literatur werden sie häufig als *abstrakte Automaten* bezeichnet und bezüglich ihres Verhaltens analysiert [37, 98]. Stochastische Automaten mit Anwendungen in der Physik, der Ökonomie oder der Informationstechnik beschreiben zufällige Prozesse, die auf Grund ihrer Hauptanwendungsgebiete weitgehend als autonome Automaten auftreten [16]. In der Informatik spielt die Eigenschaft von Automaten, eine gegebene Symbolfolge zu „akzeptieren“, eine entscheidende Rolle [4, 21, 40, 89]. Die dort verwendeten  $\Sigma$ -Automaten (Standardautomaten) sind autonome Automaten, deren Zustandsübergängen Buchstaben (Ereignisse) eines Alphabets zugeordnet sind. Beim Entwurf digitaler Schaltungen liegt ein Schwerpunkt der Betrachtungen auf der Zerlegung von komplexen, durch Automaten beschriebenen Systemen in kleine, einfach zu realisierende Komponenten, wofür unterschiedliche Verfahren der strukturellen Analyse entwickelt wurden [14].

Demgegenüber gibt es nur wenige Einführungen in die Automatentheorie, die Automaten als Modelle dynamischer Systeme mit Eingang und Ausgang verwenden wie z. B. [56, 71]. In der englischsprachigen Literatur werden die stochastischen Automaten mit Eingang und Ausgang als *Hidden Markov Models* bezeichnet.

Petrinetze wurden von C.A. PETRI in seiner 1962 erschienenen Dissertation [79] eingeführt und ab etwa 1980 für Steuerungsaufgaben eingesetzt. Die ersten ausführlichen Einführungen in diese Thematik sind [1, 54, 99]. In diesen Büchern werden auch Erweiterungen der Petrinetze gegenüber der hier behandelten Grundform beschrieben. Test- und Inhibitorkanten dienen der Darstellung von Kopplungen von Petrinetzen, Zeitbewertungen an den Präkanten bzw. Stellen der Erweiterung der hier betrachteten logischen Beschreibung zu einer zeitbewerteten Beschreibung. Die Nutzung von Petrinetzen für die Modellierung technischer Systeme in unterschiedlichen Gebieten steht im Mittelpunkt des Lehrbuches [92].

Eine umfassende Darstellung der in der Automatisierungstechnik verwendeten Modelle ereignisdiskreter Systeme findet sich in [18]. Die dort eingeführten Modelle bilden die Grundlage vieler in den letzten Jahren entwickelter rechnergestützter Werkzeuge zur Beschreibung und Analyse diskreter Systeme wie z. B. *Stateflow* (Mathworks), *Statemate* (i-logix), *GRAFCET* (IEC 60848) oder *HiGraph* (Siemens). Mit diesen Werkzeugen soll der Anwender sein System in einer dem betrachteten Anwendungsgebiet möglichst nahe stehenden Beschreibungsform darstellen und mit dieser Beschreibung das Verhalten unter vorgegebenen Bedingungen analysieren. Die meisten Werkzeuge basieren auf Automaten oder Petrinetzen, führen jedoch eine Vielzahl weiterer Beschreibungselemente ein, die das Aufstellen eines Modells vereinfachen und z. B. mit Hilfe von Uhren auch eine zeitbewertete Beschreibung ermöglichen. Je größer der Funktionsumfang wird, desto mehr schränkt sich die Nutzung dieser Mittel auf die Simulation ein. Insbesondere bieten die wenigsten der verfügbaren Werkzeuge systematische Analyse- oder Entwurfshilfsmittel für Steuerungen. Übersichten über diese Werkzeuge und die ihnen zu Grunde liegenden Beschreibungsmittel werden z. B. in [2] und [93] gegeben.

Die Eigenschaften nichtdeterministischer Systeme zu verstehen, bereitet mitunter große Schwierigkeiten, weil man an die Eindeutigkeit der Bewegung, mit der ein technischer Prozess auf seine Eingangsgrößen reagiert, gewöhnt ist. Noch schwieriger erweist sich dann das Verständnis wahrscheinlichkeitstheoretischer Modelle wie Markovketten und stochastischer Automaten. Eine gute Vorstellung von derartigen Systemen vermitteln Spiele, bei denen der Zufall den Spielverlauf bestimmt. Eine interessante Betrachtung von Glücksspielen aus der Sicht ihrer wahrscheinlichkeitstheoretischen Beschreibung enthält das Buch [12], dessen Grundideen man gut auf technische Prozesse übertragen kann.





# 12

## Verhalten diskreter Systeme

*Das Verhalten diskreter Systeme ist durch Folgen von Eingaben, Zuständen und Ausgaben beschrieben. In diesem Kapitel werden die charakteristischen Verhaltensformen von Automaten und Petrinetzen behandelt.*

### 12.1 Vorhersage des Systemverhaltens

Die Aufgabe, das Verhalten diskreter Systeme mit Hilfe eines Modells vorherzusagen, ist im Wesentlichen dieselbe wie bei kontinuierlichen Systemen. Für einen gegebenen Anfangszustand  $z_0$  und eine gegebene Eingangsfolge  $V(0 \dots k_e)$  soll unter Nutzung des dynamischen Modells die Ausgangsfolge  $W(0 \dots k_e)$  berechnet werden (vgl. Abb. 4.1 auf S. 123):

#### **Vorhersageaufgabe für diskrete Systeme**

Gegeben: Modell

Anfangszustand  $z_0$

Folge der Eingangssymbole  $V(0 \dots k_e)$

Gesucht: Folge der Ausgangssymbole  $W(0 \dots k_e)$

Die Unterschiede zwischen dem kontinuierlichen und dem diskreten Fall liegen im Detail. Einer betrifft die Länge von Zustands- und Ausgangsfolgen. Während für kontinuierliche Systeme für jedes beliebige Zeitintervall  $[0, t_e]$  eine Trajektorie  $y_{[0, t_e]}$  existiert, muss das bei diskreten

Systemen nicht so sein. Diskrete Systeme können nach einer endlichen Zahl von Ereignissen „stehen bleiben“, also keine weiteren Ereignisse erzeugen, was sich darin ausdrückt, dass die Folge der Ausgangssymbole abbricht. Die Folge  $W(0...k_e)$  muss also gar nicht für einen beliebigen Zeithorizont  $k_e$  existieren.

Als Beispiel denke man an eine Stanze, die durch ein unvollständig ausgeworfenes Werkstück verklemmt. Auch kann eine fehlerhaft entworfene Steuerung dazu führen, dass das System unerwartet stehen bleibt. Ferner ist es möglich, dass die Steuerungsaufgabe verlangt, das System in einen bestimmten Zustand zu überführen und dort zu halten. Dann bricht die Zustandsfolge erwünschtermaßen ab.

Um die Schwierigkeiten zu umgehen, die die Behandlung endlicher Ereignisfolgen mit sich bringt, wird hier i. Allg. vorausgesetzt, dass das System beliebig lange Ereignisfolgen erzeugen kann und der Zeithorizont  $k_e$  nicht durch die Systemdynamik beschränkt ist. Bei Systemen, die nach einer endlichen Anzahl von Ereignissen „stehen bleiben“, kann die Zustandsfolge durch eine Wiederholung des letzten Zustands beliebig verlängert werden, um diese Voraussetzung zu erfüllen.

## 12.2 Verhalten deterministischer Automaten

### 12.2.1 Berechnung der Zustands- und Ausgabefolge

Das Verhalten deterministischer E/A-Automaten ist durch Gl. (11.16)

$$\mathcal{A}: \quad p(k+1) = G(v(k))p(k), \quad p(0) = p_0 \quad (12.1)$$

beschrieben, aus der man bei mehrfacher Anwendung die Beziehung

$$p(k) = G(v(k-1)) \cdot G(v(k-2)) \cdot \dots \cdot G(v(0)) \cdot p_0 \quad (12.2)$$

erhält. Mit dieser Gleichung kann man für jede Eingangsfolge  $V(0...k_e)$  die Zustandsfolge  $Z(0...k_e)$  berechnen und daraus die Ausgangsfolge  $W(0...k_e)$  entsprechend Gl. (11.17)

$$p_w(k) = H(v(k))p(k)$$

bestimmen. Für autonome Automaten vereinfacht sich Gl. (11.16) zu

$$p(k+1) = Gp(k), \quad p(0) = p_0 \quad (12.3)$$

und Gl. (12.2) zu

$$p(k) = G^k p_0.$$

**Interpretation anhand des Automatengraphen.** Die angegebenen Berechnungsvorschriften haben eine direkte Interpretation im Automatengraphen, die zunächst für autonome Automaten erklärt wird, für die die Matrix  $G$  nicht von der Eingabe abhängt. Da die Matrix  $G$  die Adjazenzmatrix des Automatengraphen ist, gelten für sie die aus der Graphentheorie bekannten Ergebnisse.

Im Vektor  $p_0$  ist nur das  $z_0$ -te Element gleich eins, alle anderen gleich null. Man erhält aus der Multiplikation dieses Vektors mit der Matrix  $G$  den Vektor

$$p(1) = Gp_0,$$

der den Nachfolgezustand von  $z_0$  beschreibt. Dieser Multiplikation entspricht im Automatengraphen die Bewegung entlang der vom Zustand  $z_0$  ausgehenden Kante, die zum Nachfolgezustand  $z(1)$  führt, der im Vektor  $p(1)$  durch die Eins markiert ist. Allgemeiner bedeutet die Multiplikation des Vektors  $p(k)$  mit der Adjazenzmatrix  $G$  des Automatengraphen die Bewegung vom Zustand  $z(k)$  zum Nachfolgezustand  $z(k+1)$  entlang der vom Knoten  $z(k)$  ausgehenden Kante.

Bei nichtautonomen Automaten hängt die Bewegung auch von der aktuellen Eingabe ab. Im Automatengraphen folgt die Bewegung der vom Zustand  $z(k)$  ausgehenden Kante, die mit der Eingabe  $v(k)$  bewertet ist (vgl. Gl. (12.1)).

Mit der Bewegung im Automatengraphen ist die Folge von Ausgaben  $w(k)$  verbunden, die an den verwendeten Kanten stehen. Damit beschreibt die Bewegung nicht nur eine Zustandsfolge, sondern gleichzeitig eine Ausgabefolge.

Automaten können also sowohl unter Nutzung ihrer analytischen Darstellung (11.16), (11.17) als auch anhand ihres Automatengraphen analysiert werden (wobei in beiden Fällen dieselben Ergebnisse erhalten werden). Im Folgenden wird diese Parallelität ausgenutzt, um die analytisch erhaltenen Ergebnisse grafisch zu interpretieren.

### 12.2.2 Erreichbarkeitsanalyse des Automatengraphen

In diesem Abschnitt wird untersucht, welche Zustände ein deterministischer Automat ausgehend vom Anfangszustand  $z_0$  annehmen kann. Diese Aufgabestellung entspricht einer Erreichbarkeitsanalyse des Automatengraphen. Sie wird zunächst für autonome Automaten erläutert.

**Erreichbarkeitsanalyse autonomer Automaten.** Ein Automat nimmt einen Zustand  $\bar{z}$  auf seiner in  $z_0$  beginnenden Trajektorie genau dann an, wenn es im Automatengraphen einen Pfad vom Knoten  $z_0$  zum Knoten  $\bar{z}$  gibt. Der Zustand  $\bar{z}$  heißt dann vom Zustand  $z_0$  aus *erreichbar*.

Da die Multiplikation des Vektors  $p_0$  mit der Adjazenzmatrix  $G$  der Bewegung entlang einer Kante entspricht, ist im Vektor

$$p(k) = G^k p_0 \tag{12.4}$$

derjenige Zustand  $z(k)$  mit einer Eins gekennzeichnet, der von  $z_0$  aus nach  $k$  Schritten angenommen wird. Zwischen  $z_0$  und  $z(k)$  gibt es im Automatengraphen also einen Pfad der Länge  $k-1$  (wobei die Länge des Pfades die Anzahl der Kanten des Pfades angibt). Für die Erreichbarkeitsuntersuchung muss man Pfade beliebiger Länge betrachten, allerdings auf Grund des deterministischen Verhaltens des Automaten nur Pfade mit der Länge  $k \leq N-1$ , wobei  $N$  die Anzahl der Zustände und damit die Anzahl der Knoten im Automatengraphen angibt. Wenn ein Zustand nicht auf einem Pfad der Länge  $k \leq N-1$  erreicht werden kann, kann er es auch nicht auf längeren Pfaden.

Statt die entsprechend Gl. (12.4) erzeugten Vektoren  $p(1), p(2), \dots, p(N)$  zu betrachten, kann man deren Summe  $\bar{p}$  untersuchen, die sich aus der Beziehung

$$\bar{p} = \sum_{k=0}^{N-1} G^k p_0$$

ergibt. Von  $z_0$  aus sind alle Zustände  $i$  erreichbar, für die das Element  $\bar{p}_i$  des Vektors  $\bar{p}$  ungleich null ist. Das heißt:

|| Der Zustand  $i \in \mathcal{Z}$  ist genau dann von  $z_0$  erreichbar, wenn  $\bar{p}_i \neq 0$  gilt.

Übrigens erlaubt es die Matrix

$$\bar{G} = \sum_{k=0}^{N-1} G^k,$$

mit der die Beziehung  $\bar{p} = \bar{G} p_0$  gilt, die Erreichbarkeitsanalyse für alle Anfangszustände gleichzeitig durchzuführen. Ein Zustand  $i$  ist von einem Zustand  $j$  genau dann erreichbar, wenn das Element  $\bar{g}_{ij}$  der Matrix  $\bar{G}$  nicht verschwindet. Die Werte des Elements  $\bar{g}_{ij}$  geben an, auf wie vielen Pfaden es möglich ist, vom Zustand  $j$  zum Zustand  $i$  zu gelangen, wobei die Pfade unterschiedliche Länge haben und Knoten mehrfach enthalten können.

**Erreichbarkeitsanalyse nichtautonomer Automaten.** Bei Automaten mit Eingang kann man mit Hilfe einer Erreichbarkeitsanalyse zwei unterschiedliche Fragen beantworten. Erstens kann es interessant sein zu bestimmen, in welche Zustände der Automat bei einer konstanten Eingabe  $v(k) = \bar{v}$  übergeht. Die beschriebene Erreichbarkeitsanalyse muss dann mit  $G(\bar{v})$  an Stelle von  $G$  durchgeführt werden. Bei Steuerungsproblemen will man beispielsweise erreichen, dass das System bei einer geeignet gewählten Eingabe  $\bar{v}$  in einer Teilmenge  $\tilde{\mathcal{Z}} \subset \mathcal{Z}$  der Zustandsmenge verbleibt. Dann dürfen keine außerhalb dieser Teilmenge liegenden Zustände unter Wirkung der Eingabe  $v(k) = \bar{v}$  vom aktuellen Zustand aus erreichbar sein.

Zweitens kann man die Frage stellen, in welche Zustände der Automat vom Anfangszustand aus durch eine geeignet gewählte Eingabefolge  $V$  gesteuert werden kann. Diese Frage stellt sich beispielsweise bei der Steuerbarkeitsanalyse, die im Abschn. 12.2.4 genauer betrachtet wird. Für ihre Beantwortung bildet man die Matrix

$$\tilde{G} = \max_{v \in \mathcal{V}} G(v),$$

deren Element  $\tilde{g}_{\bar{z}z}$  genau dann ungleich null ist, wenn es eine Eingabe  $v \in \mathcal{V}$  gibt, für die der Automat vom Zustand  $z$  direkt in den Zustand  $\bar{z}$  übergeht. Der zu dieser Adjazenzmatrix gehörende Automatengraph entsteht aus dem Automatengraphen des gegebenen Automaten durch Weglassen der Kantenbewertungen. Es entsteht scheinbar ein autonomer nichtdeterministischer Automat. Die Anwendung der Erreichbarkeitsanalyse auf die Matrix  $\tilde{G}$  zeigt, welche Zustände durch eine geeignet gewählte Steuerfolge erreicht werden können.

### 12.2.3 Strukturelle Analyse deterministischer Automaten

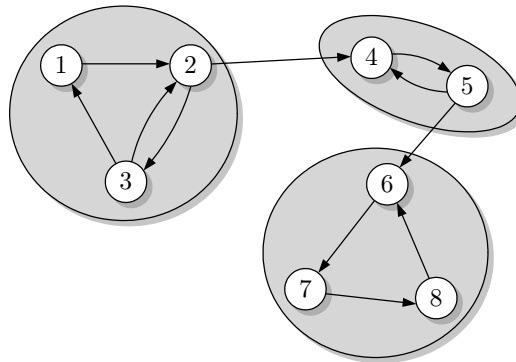
Wichtige Eigenschaften deterministischer Automaten erkennt man, wenn man die Zustände in Abhängigkeit von ihrer gegenseitigen Erreichbarkeit klassifiziert.

|| Zwei Zustände  $\bar{z}$  und  $z$  heißen *stark zusammenhängend*, wenn es im Automatengraphen einen Pfad vom Knoten  $\bar{z}$  zum Knoten  $z$  und einen Pfad vom Knoten  $z$  zum Knoten  $\bar{z}$  gibt.

Man kann die Zustandsmenge  $\mathcal{Z}$  des Automaten in  $q$  disjunkte Mengen  $\mathcal{Z}_i$  stark zusammenhängender Zustände zerlegen:

$$\mathcal{Z} = \mathcal{Z}_1 \cup \mathcal{Z}_2 \cup \dots \cup \mathcal{Z}_q, \quad \mathcal{Z}_i \cap \mathcal{Z}_j = \emptyset \text{ für } i \neq j.$$

Dementsprechend kann man auch den Automatengraphen in  $q$  stark zusammenhängende Teilgraphen zerlegen, die nur in jeweils einer Richtung durch Kanten verbunden sind.



**Abb. 12.1:** Zerlegung der Zustandsmenge in Teilmengen stark zusammenhängender Zustände

Abbildung 12.1 zeigt ein Beispiel, bei dem die Menge der acht Automatenzustände in drei Teilmengen zerlegt wurde. Die zur selben Teilmenge gehörenden Zustände sind untereinander stark zusammenhängend, denn sie sind durch Pfade paarweise verbunden. Demgegenüber gibt es zwischen Zuständen aus unterschiedlichen Teilmengen höchstens einen Pfad in einer Richtung.

Sind alle Zustände der Zustandsmenge  $\mathcal{Z}$  stark zusammenhängend und gilt folglich  $q = 1$ , so heißt der Automat *irreduzibel* (nicht reduzierbar). Bei derartigen Automaten wird jeder Zustand nach einer bestimmten Zahl von Zustandsübergängen erneut angenommen. Die Zustände heißen deshalb *rekurrent* (wiederkehrend). Wenn die Wiederholung eines Zustands periodisch ist, was man an Zyklen im Graphen erkennen kann, heißt der Zustand auch *periodisch*. Da die hier betrachteten Automaten deterministisch sind, sind alle Zustände eines irreduziblen Automaten entweder periodisch mit derselben Periodenlänge oder sie sind aperiodisch. Periodische Zustandsänderungen sind typisch für zyklische Prozesse.

**Beispiel 12.1** Irreduzibler Automat

Der Automat in Abb. 11.6 auf S. 345 ist irreduzibel, denn jeder Knoten ist von jedem anderen Knoten aus erreichbar. Die Zustände sind periodisch, denn sie werden nach jeweils fünf Zustandsübergängen wieder angenommen. Der Automat wäre aperiodisch, wenn eine der gezeigten Kanten fehlt. Dann werden alle Zustände in Abhängigkeit vom Anfangszustand entweder nie oder genau einmal angenommen und dieser Automat erzeugt nur eine endliche Zustandsfolge.  $\square$

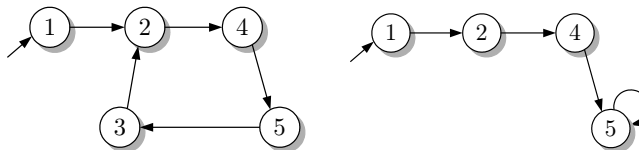
Wenn der Automat reduzierbar ist, so gilt  $q > 1$ , d. h., es gibt mindestens zwei Mengen  $\mathcal{Z}_i$  stark zusammenhängender Zustände. Da zwischen diesen Zustandsmengen Kanten nur in einer Richtung, also z. B. von Knoten der Menge  $\mathcal{Z}_1$  zu Knoten der Menge  $\mathcal{Z}_2$ , aber nicht umgekehrt existieren können, gibt es mindestens einen Teilgraphen, der, sobald er einmal durch den Automaten erreicht wird, nicht wieder verlassen wird. Die Zustandsmenge dieses oder dieser Teilgraphen heißt *ergodisch*. Wenn der betreffende Teilgraph nur aus einem einzelnen Zustand besteht, so wird dieser Zustand *absorbierend* genannt. Andererseits gibt es Zustände, die nur einmal angenommen werden können und als *transiente* Zustände bezeichnet werden.

**Beispiel 12.2** Reduzierbare Automaten

Der in Abb. 12.2 links gezeigte Automat ist reduzierbar mit den Mengen

$$\mathcal{Z}_1 = \{1\}, \quad \mathcal{Z}_2 = \{2, 3, 4, 5\}.$$

Der Automat kann von der Zustandsmenge  $\mathcal{Z}_1$  in die Zustandsmenge  $\mathcal{Z}_2$  übergehen, aber nicht umgekehrt. Sobald der Automat einen Zustand der Menge  $\mathcal{Z}_2$  angenommen hat, bleibt sein Zustand in dieser Menge. Diese Zustandsmenge ist deshalb ergodisch. Der Zustand 1 ist transient. Die Zustände 2, 3, 4 und 5 sind rekurrent.



**Abb. 12.2:** Automat mit periodischer Zustandsmenge (links) und mit absorbierendem Zustand (rechts)

Abbildung 12.2 (rechts) zeigt einen Automaten mit absorbierendem Zustand 5. Der Automat ist reduzierbar, denn jeder Zustand bildet eine eigene stark zusammenhängende Teilmenge. Die Zustände 1, 2 und 4 sind transient.  $\square$

Jeder Automat mit transienten Zuständen muss nach einer bestimmten Anzahl von Zustandsübergängen eine ergodische Zustandsmenge oder einen absorbierenden Zustand erreichen, die bzw. den er nicht wieder verlassen kann. Das System bewegt sich dann innerhalb einer Menge von Endzuständen bzw. hat seinen Endzustand erreicht. Es ist von der Steuerungsaufgabe abhängig, ob man das System in einen solchen Zustand steuern will oder ob man verhindern will,

dass das System in einen solchen Zustand gelangt. Deshalb steht entsprechend der Steuerungsaufgabe entweder die Bewegung des Systems innerhalb des irreduziblen Teils des Automatengraphen oder die Bewegung zu einer ergodischen Zustandsmenge im Mittelpunkt. Im letzten Fall muss durch eine geeignet gewählte Steuerung eine vorgegebene Zustandsmenge ergodisch gemacht werden.

**Erweiterung auf nichtautonome Automaten.** Die bisher eingeführten Begriffe können auf Automaten mit Eingängen erweitert werden, wobei man den Automaten entweder für beliebige Eingänge oder für einen konstanten Eingang  $v(k) = \bar{v}$  untersucht. Im ersten Fall betrachtet man alle Kanten unabhängig von den mit diesen Kanten verbundenen Eingangsgrößen. Dann heißt ein Automatenzustand  $z$  erreichbar vom Zustand  $z_0$ , wenn es eine Eingangsfolge  $V(0 \dots k_e)$  gibt, so dass  $z(k_e) = z$  gilt. Im zweiten Fall streicht man aus dem Automatengraphen diejenigen Kanten heraus, die nicht zur Eingangsgröße  $\bar{v}$  gehören, und analysiert den auf diese Weise entstandenen Automaten.

#### 12.2.4 Steuerbarkeit deterministischer Automaten

Für viele Automatisierungsaufgaben ist es wichtig zu wissen, ob das durch einen deterministischen Automaten mit der Zustandsgleichung

$$p(k+1) = G(v(k)) p(k), \quad k = 0, 1, 2, \dots$$

beschriebene System von jedem seiner Zustände durch eine geeignete Eingangsfolge in jeden anderen Zustand überführt werden kann. Diese Eigenschaft entspricht der vollständigen Steuerbarkeit kontinuierlicher Systeme (vgl. Definition 5.1 auf S. 156). Sie hängt nur von der Zustandsübergangsfunktion  $G$  des Automaten ab, so dass sich die folgenden Untersuchungen auf den Automaten

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, G, z_0)$$

ohne Ausgang und ohne Ausgabefunktion beschränken können.

##### **Definition 12.1 (Steuerbarkeit deterministischer Automaten)**

*Ein deterministischer Automat*

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, G, z_0)$$

*heißt vollständig steuerbar, wenn er von jedem beliebigen Anfangszustand  $z_0$  in endlicher Zeit  $k_e$  durch eine geeignet gewählte Eingangsfolge  $V(0 \dots k_e)$  in einen beliebigen Endzustand  $z(k_e) = z_e$  überführt werden kann.*

Bei nicht vollständig steuerbaren Automaten kann man gegebenenfalls Zustandspaare  $(z_0, z_e)$  finden, für die eine Eingangsfolge  $V$  existiert, durch die der Automat vom Zustand  $z_0$  in den Zustand  $z_e$  überführt wird.  $z_e$  heißt dann von  $z_0$  aus erreichbar.

Die Steuerbarkeit eines Automaten kann anhand des Automatengraphen überprüft werden. Ein Zustand  $z_e$  ist genau dann vom Anfangszustand  $z_0$  aus erreichbar, wenn es im Automatengraphen einen Pfad von  $z_0$  nach  $z_e$  gibt. Die an den Kanten des Graphen stehenden Eingaben

ergeben die Eingabefolge  $V(0...k_e)$ , für die der Automat den gewünschten Zustandsübergang vollzieht.  $k_e$  bezeichnet die Pfadlänge.

Für die Steuerbarkeitsuntersuchung sind die Kantenbewertungen ohne Belang. Wenn man die Kantenbewertungen weglässt, erhält man einen autonomen nichtdeterministischen Automaten  $\mathcal{N} = (\mathcal{Z}, \tilde{G}, z_0)$ , für dessen Zustandsübergangsrelation  $\tilde{G}$  die Beziehung

$$\tilde{G}(z', z) = 1 \iff \exists v : z' = G(z, v) \quad (12.5)$$

gilt. Hier wird die Zustandsübergangsrelation des autonomen Automaten zur Unterscheidung von der Zustandsübergangsfunktion  $G$  des deterministischen Automaten mit Eingang als  $\tilde{G}$  bezeichnet.

**Satz 12.1 (Steuerbarkeit deterministischer Automaten)**

*Ein deterministischer Automat  $\mathcal{A} = (\mathcal{Z}, \mathcal{V}, G, z_0)$  ist genau dann vollständig steuerbar, wenn der aus ihm entsprechend Gl. (12.5) gebildete nichtdeterministische Automat  $\mathcal{N} = (\mathcal{Z}, \tilde{G}, z_0)$  irreduzibel ist.*

Wie bei kontinuierlichen Systemen bedeutet die Steuerbarkeit die Möglichkeit, das System in einen gewünschten Zustand zu überführen. Sie bedeutet aber nicht, dass es auch möglich ist, das System durch einen geeignet gewählten Eingang in dem gewünschten Zustand zu halten.

## 12.3 Verhalten nichtdeterministischer Automaten

### 12.3.1 Berechnung der Zustands- und Ausgabefolgen

Das Verhalten nichtdeterministischer Automaten wird durch die Beziehung (11.32)

$$\mathcal{N} : (z(k+1), w(k), z(k), v(k)) \in \mathcal{L}, \quad z(0) = z_0 \quad (12.6)$$

dargestellt, die die aus einem möglichen Nachfolgezustand  $z(k+1)$ , der Ausgabe  $w(k)$ , dem aktuellen Zustand  $z(k)$  und der Eingabe  $v(k)$  bestehenden Tupel beschreibt. Wendet man diese Beziehung auf einen Anfangszustand  $z_0$  und eine Eingangsfolge

$$V(0...k_e) = (v(0), v(1), v(2), \dots, v(k_e))$$

an, so erhält man eine Menge von möglichen Zustandsfolgen

$$Z(0...k_e) = (z(0), z(1), z(2), \dots, z(k_e))$$

und Ausgabefolgen

$$W(0...k_e) = (w(0), w(1), w(2), \dots, w(k_e)).$$

Jeweils vier zusammengehörende Elemente dieser drei Folgen erfüllen die Verhaltensrelation  $\mathcal{L}$ , wobei allerdings nicht wie beim deterministischen Automaten eine Gleichung diese Größen eindeutig bestimmt, sondern die Verhaltensrelation möglicherweise mehrere Werte zulässt.



**Beispiel 12.3** Verhalten eines nichtdeterministischen Automaten

Der in Abb. 11.19 auf S. 367 gezeigte Automat ist durch

$$\mathcal{N} = (\{1, 2, 3, 4, 5, 6\}, \{1, 2\}, \{1, 2, 3\}, L, \{1, 4\})$$

mit

$$L =$$

$z'$	$w$	$z$	$v$
1	1	1	1
5	2	1	1
4	3	1	2
2	2	2	1
1	1	2	2
2	1	3	1
2	2	3	1
3	1	3	1
2	3	3	2
3	3	3	2
5	3	4	1
5	1	4	2
3	3	5	1
6	1	5	1
4	3	5	2
5	2	5	2
6	2	5	2
3	3	6	1
3	3	6	2

beschrieben. Für die Eingangsfolge  $V(0\dots 2) = (2, 1, 2)$  erhält man entsprechend Gl. (12.6) die folgenden Zustands- und Ausgangsfolgen.

- Mit dem Anfangszustand  $z(0) = 1$  und der Eingabe  $v(0) = 2$  ist Gl. (11.33) für genau ein Tupel erfüllt

$$L(4, 3, 1, 2) = 1.$$

Der Automat geht in den Zustand  $z(1) = 4$  über und erzeugt dabei die Ausgabe  $w(0) = 3$ . Vom Anfangszustand  $z(0) = 4$  liest man aus der Tabelle für  $L$  die Gleichung

$$L(5, 1, 4, 2) = 1$$

ab, d. h., der Automat geht in den Zustand  $z(1) = 5$  über und erzeugt dabei die Ausgabe  $w(0) = 1$ .

- Für den Zeitschritt  $k = 1$  müssen alle Tupel  $(z(2), w(1), z(1), v(1))$  mit  $z(1) = 4$  oder  $z(1) = 5$  sowie  $v(1) = 1$  gesucht werden:

$$L(5, 3, 4, 1) = 1$$

$$L(3, 3, 5, 1) = 1$$

$$L(6, 1, 5, 1) = 1.$$

Sie besagen, dass der Automat in die Zustände  $z(2) = 3$  oder  $z(2) = 5$  oder  $z(2) = 6$  übergeht und die Ausgabe  $w(1) = 1$  oder  $w(1) = 3$  erzeugt.

- Für  $k = 2$  müssen deshalb drei Fälle untersucht werden. Für  $z(2) = 3$  gibt es bei der Eingabe  $v(2) = 2$  die Tupel

$$L(2, 3, 3, 2) = 1$$

$$L(3, 3, 3, 2) = 1,$$

für die der Automat in einen der Nachfolgezustände  $z(3) = 2$  oder  $z(2) = 3$  übergeht, wobei er in jedem Fall die Ausgabe  $w(2) = 3$  erzeugt. Für den Zustand  $z(2) = 5$  sind folgende Tupel der Verhaltensrelation wichtig:

$$L(4, 3, 5, 2) = 1$$

$$L(5, 2, 5, 2) = 1$$

$$L(6, 2, 5, 2) = 1.$$

Der Automat geht in die Zustände 4, 5 oder 6 über und erzeugt die Ausgaben 2 oder 3. Für den Zustand  $z(2) = 6$  führt der Eintrag

$$L(3, 3, 6, 2) = 1$$

der Verhaltensrelation auf den Nachfolgezustand  $z(3) = 3$ , wobei die Ausgabe  $w(2) = 3$  generiert wird.

Damit erhält man folgende Zustands- und Ausgangsfolgen:

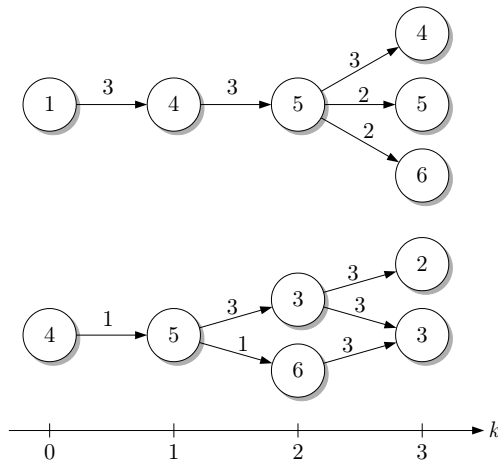
$$\begin{aligned} Z(0\dots 3) &= (1, 4, 5, 4) \text{ oder } (1, 4, 5, 5) \text{ oder } (1, 4, 5, 6) \text{ oder } (1, 5, 3, 2) \\ &\quad \text{oder } (1, 5, 3, 3) \text{ oder } (1, 5, 6, 3) \end{aligned}$$

$$W(0\dots 2) = (1, 1, 3) \text{ oder } (1, 3, 3) \text{ oder } (3, 3, 2) \text{ oder } (3, 3, 3). \quad (12.7)$$

**Diskussion.** Das Ergebnis zeigt zwei für das Verhalten nichtdeterministischer Automaten charakteristische Eigenschaften. Erstens gibt es sowohl für die Zustandsfolge als auch für die Ausgabefolge mehrere Möglichkeiten, die als „Lösung“ der Gl. (11.33) entstehen. Diese Lösungen werden häufig zu Mengen möglicher Trajektorien zusammengefasst. Zweitens muss die Anzahl der möglichen Zustandsfolgen nicht mit der Anzahl der Ausgabefolgen übereinstimmen. Bei dem Beispiel erzeugt der Automat bei den Zustandsfolgen  $(4, 5, 3, 2)$  und  $(4, 5, 3, 3)$  dieselbe Ausgabefolge  $(1, 3, 3)$ . Andererseits kann auch der Fall eintreten, dass auf einer Zustandsfolge eine von mehreren Ausgabefolgen erzeugt wird. Dieser Fall tritt ein, wenn der hier betrachtete Automat im Zustand 3 die Eingabe 1 erhält. Wenn man den Automaten genauer analysieren will, ist es deshalb besser, die Zustandsfolgen zusammen mit der zugehörigen Ausgabefolge zu notieren, beispielsweise in der in Abb. 12.3 dargestellten Baumstruktur, die die Zustandsfolgen zusammen mit den bei den Zustandsübergängen erzeugten Ausgaben zeigt. Dieses Diagramm, bei dem die Knoten bezüglich der Zeit sortiert sind und jeder Knoten mit mindestens einem Vorgänger und einem Nachfolger verbunden ist, wird als *Trellis-Diagramm* bezeichnet.

Die Abbildung zeigt, dass bei nichtdeterministischen Automaten mit vielen nichtdeterministischen Zustandsübergängen die Menge der zu einem Zeitpunkt  $k$  möglichen Zustände sehr groß sein kann.

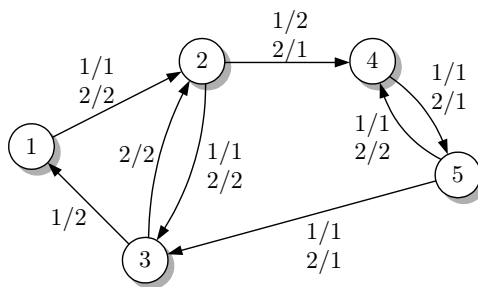
Die hier angegebene Berechnung der Zustands- und Ausgabefolgen nutzt die Darstellung der Verhaltensrelation  $L$  als Tabelle und zeigt, wie man die ungewohnte Beschreibung der Bewegung nichtdeterministischer Automaten durch Gl. (11.33) anwenden kann. Für kleine Automaten, deren Graphen man übersichtlich aufzeichnen kann, lassen sich die gesuchten Folgen anschaulicher als Pfade im Automatengraphen ablesen. Die in Abb. 12.3 oben angegebene Zustandsfolge erkennt man an den entsprechenden Kanten  $1 \rightarrow 4 \rightarrow 5 \rightarrow 4$  des Automatengraphen, die gültige Zustandsübergänge beschreiben, weil dieser Pfad in einem der beiden möglichen Anfangszustände beginnt und an den Kanten die Eingabefolge  $V(0\dots 2) = (2, 1, 2)$  steht. Dasselbe gilt für die anderen Zustandsfolgen.  $\square$



**Abb. 12.3:** Grafische Darstellung des Verhaltens eines nichtdeterministischen Automaten

### 12.3.2 Strukturelle Analyse nichtdeterministischer Automaten

Die für deterministische Automaten eingeführten Begriffe und Eigenschaften gelten in etwas verallgemeinerter Form auch für nichtdeterministische Automaten. Der Unterschied liegt vor allem in der Tatsache, dass die Zustandsübergänge nicht mehr eindeutig durch den aktuellen Zustand und den aktuellen Eingang bestimmt werden. Wenn die Zustände stark zusammenhängend sind, so ist nicht gesichert, dass alle diese Zustände bei der Bewegung des Automaten tatsächlich angenommen werden, denn der Automat kann auf Grund nichtdeterministischer Übergänge bestimmte Zustände „auslassen“, obwohl sie vom Anfangszustand aus erreichbar sind.



**Abb. 12.4:** Irreduzibler Automat

**Beispiel 12.4** Strukturelle Analyse eines nichtdeterministischen Automaten

Abbildung 12.4 zeigt den Graphen eines irreduziblen nichtdeterministischen Automaten. Im Unterschied zu irreduziblen deterministischen Automaten müssen die Zustände nicht periodisch angenommen werden, denn auf Grund der nichtdeterministischen Zustandsübergänge ist für die konstante Eingabe  $v(k) = 1$  z. B. die folgende Zustandsfolge möglich:

$$Z = (2, 3, 1, 2, 4, 5, 3, 1, \dots).$$

Der Automat kann für diese Eingabe sogar periodisch zwischen den Zuständen 4 und 5 hin- und herschalten. □

**Aufgabe 12.1** Analyse des Modells der vier Werkzeugmaschinen

Analysieren Sie den in Abb. 11.18 auf S. 365 gezeigten Automaten, der die Arbeitsweise von vier Werkzeugmaschinen beschreibt. Welche Konsequenzen hat Ihr Analyseergebnis für die Auslastung der Werkzeugmaschinen? Was passiert, wenn die Maschine  $M_2$  die Bearbeitung eines Werkstücks aufgrund einer Störung nicht beendet? □

**12.3.3 Steuerbarkeit nichtdeterministischer Automaten**

Die Steuerbarkeitsanalyse nichtdeterministischer Automaten stößt auf eine durch das nichtdeterministische Verhalten hervorgerufene Schwierigkeit. Da bei nichtdeterministischen Zustandsübergängen nicht bekannt ist, welcher der möglichen Nachfolgezustände angenommen wird, kann man nur unter sehr einschränkenden Bedingungen sicher sein, dass man durch eine Steuerfolge  $V(0 \dots k_e)$  den Automaten von seinem Anfangszustand  $z_0$  in einen gewünschten Endzustand  $z_e$  überführt. Deshalb ist es zweckmäßig, die Steuerbarkeit in Bezug zu einer Menge  $\tilde{Z} \subset Z$  von Zuständen zu definieren, wobei es gleichgültig ist, in welchen Zustand  $z \in \tilde{Z}$  das System übergeht.

**Definition 12.2 (Steuerbarkeit nichtdeterministischer Automaten)**

*Ein nichtdeterministischer Automat  $\mathcal{N} = (Z, \mathcal{V}, L, Z_0)$  heißt steuerbar in Bezug zur Menge  $Z_0$  von Anfangszuständen und einer Zielzustandsmenge  $\tilde{Z} \subseteq Z$ , wenn er von jedem Anfangszustand  $z_0 \in Z_0$  in endlicher Zeit  $k_e$  durch eine geeignet gewählte Eingangsfolge  $V(0 \dots k_e)$  in einen Endzustand  $z(k_e) \in \tilde{Z}$  überführt wird.*

Man sagt dann auch, dass die Menge  $\tilde{Z}$  von  $Z_0$  aus steuerbar ist.

Bei nichtdeterministischen Automaten ist die Erreichbarkeit eines Zustands im Automatengraphen nur eine notwendige Bedingung dafür, dass der Automat durch eine geeignet gewählte Eingabefolge in diesen Zustand überführt wird. Deshalb gilt:

|| Wenn ein nichtdeterministischer Automat bezüglich der Mengen  $Z_0$  und  $\tilde{Z}$  steuerbar ist, so ist von jedem  $z_0 \in Z_0$  mindestens ein Zustand  $z \in \tilde{Z}$  im Automatengraphen erreichbar.

Um die Steuerbarkeit zu überprüfen, geht man nach folgendem Algorithmus vor:

---

**Algorithmus 12.1** Steuerbarkeitsanalyse nichtdeterministischer Automaten
 

---

**Gegeben:** Automat  $\mathcal{N}$   
 Menge  $Z_0$  der Anfangszustände  
 Zielzustandsmenge  $\bar{Z}$

**Schleife:** Für jeden Anfangszustand  $z_0 \in Z_0$  wird folgende Analyse durchgeführt:

1. Bestimmen Sie einen Pfad vom Anfangszustand  $z_0$  zu einem Zustand  $z \in \bar{Z}$ .
2. Erzeugen Sie eine Eingabefolge  $V(0 \dots k_e)$ , unter der der Automat diese Zustandsfolge erzeugen kann.
3. Bestimmen Sie die Menge aller Pfade, die der Automat unter dieser Eingabefolge  $V(0 \dots k_e)$  durchlaufen kann.
4. Wenn alle diese Pfade zu oder durch einen Zustand  $z \in \bar{Z}$  führen, ist der Automat bezüglich  $z_0$  und  $\bar{Z}$  steuerbar. Andernfalls bestimmen Sie entweder eine andere Eingangsfolge, die zu dem im Schritt 1 ermittelten Pfad gehört, und setzen den Algorithmus mit Schritt 3 fort, oder Sie bestimmen einen anderen Pfad von  $z_0$  zu einem Zustand  $z \in \bar{Z}$  und setzen mit Schritt 2 fort. Wenn dies nicht möglich ist, ist der Automat bezüglich  $z_0$  und  $\bar{Z}$  nicht steuerbar.

**Ergebnis:** Aussage über die Steuerbarkeit des Automaten bezüglich  $Z_0$  und  $\bar{Z}$ .

---

**Beispiel 12.5** Verhalten eines Batchprozesses

Betrachtet werden die zwei Behälter in Abb. 12.5, die durch entsprechende Ansteuerung der Pumpe  $P$  und der beiden Ventile  $V_1$  und  $V_2$  gefüllt und geleert werden. Dabei ist es für die Prozessführung nur wichtig, ob die Behälter leer sind (also der Füllstand unterhalb der unteren Schranke liegt), ob die Behälter einen Füllstand bis zur Höhe der Verbindungsleitung haben oder ob der Füllstand noch höher ist. Die Ventile und die Pumpe können nur ein- oder ausgeschaltet werden.

Da jeder Behälter drei qualitativ verschiedene Füllstände hat, hat das Behältersystem neun diskrete Zustände, die entsprechend ihrer Bedeutung in Abb. 12.6 angeordnet sind. Je weiter rechts der Zustand liegt, umso höher ist der Füllstand im Tank 1, je weiter oben er liegt, umso höher ist Tank 2 gefüllt.

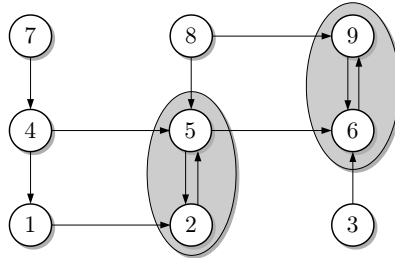
An den Kanten stehen die diskreten Werte der Eingangsgrößen, für die das Behältersystem den entsprechenden diskreten Zustandsübergang ausführt. Die drei Größen für die Pumpe, Ventil 1 und Ventil 2 sind zu einem Vektor zusammengefasst, wobei eine 1 eine laufende Pumpe bzw. ein geöffnetes Ventil und eine 0 eine ausgeschaltete Pumpe bzw. ein geschlossenes Ventil bedeutet. Eine Stern kann durch eine Null oder eine Eins ersetzt werden.

Der Automat ist nichtdeterministisch, denn beispielsweise kann es vom Zustand 5 aus bei der Eingangsgröße  $(0, 1, 1)^T$  sowohl einen Übergang zum Knoten 2 als auch zum Knoten 4 geben. Das hängt damit zusammen, dass bei geöffneten Ventilen entweder im Tank 1 oder im Tank 2 zuerst der tiefe Füllstand erreicht wird. In beiden Fällen wird nach den Zwischenzuständen 2 bzw. 4 bei gleichbleibender Eingangsgröße schließlich der Endzustand 1 erreicht.



$$V(0\dots 5) = \left( \begin{pmatrix} 1 \\ 0 \\ * \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ * \end{pmatrix} \right).$$

3. Es gibt keinen weiteren Pfad, den das System bei der angegebenen Eingangsfolge durchlaufen könnte.
4. Folglich führen alle Pfade unter der angegebenen Eingangsfolge zum Zielzustand 8. Das System ist deshalb bezüglich der Zustände  $z_0 = 1$  und  $z_e = 8$  steuerbar.



**Abb. 12.7:** Reduzierter Automat für blockiertes Ventil 1

Interessant ist es zu untersuchen, wie sich der Automatengraph verändert, wenn das Ventil 1 im geschlossenen Zustand blockiert ist. Dafür müssen alle Kanten gestrichen werden, deren zweite Komponente eine Eins ist. Der reduzierte Automatengraph ist in Abb. 12.7 zu sehen. Jetzt sind nur noch die Zustände 2, 5 sowie 6 und 9 stark zusammenhängend. Die Menge  $\{6, 9\}$  bildet eine ergodische Menge, denn von den anderen Zuständen gibt es nur Übergänge zu dieser Menge stark zusammenhängender Zustände, aber nicht in umgekehrter Richtung. Das System kann jetzt nur noch in stark eingeschränkter Weise im diskreten Zustandsraum bewegt werden.  $\square$

## 12.4 Verhalten stochastischer Automaten

### 12.4.1 Berechnung der Zustandsfolgen autonomer stochastischer Automaten

Das Verhalten autonomer stochastischer Automaten ist durch die Chapman-Kolmogorov-Gleichung (11.46) dargestellt, die ausführlich geschrieben folgende Form hat:

$$\mathcal{S} : \text{Prob}(Z(k+1)=z') = \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k+1)=z' | Z(k)=z) \cdot \text{Prob}(Z(k)=z). \quad (12.8)$$

Um die Wahrscheinlichkeit dafür zu bestimmen, dass der Zustand  $z'$  zum Zeitpunkt  $k+1$  angenommen wird, wird die Wahrscheinlichkeit  $\text{Prob}(Z(k)=z)$  dafür, dass  $z$  der aktuelle Zustand ist, mit der Wahrscheinlichkeit  $\text{Prob}(Z(k+1)=z' | Z(k)=z)$  für den Übergang vom Zustand  $z$  in den Zustand  $z'$  multipliziert. Die Summation erfolgt über alle Zustände  $z \in \mathcal{Z}$ . Zur Bestimmung der Aufenthaltswahrscheinlichkeit des Automaten in allen Zuständen muss Gl. (12.8) für alle  $z' \in \mathcal{Z}$  angewendet werden.

Fasst man, wie im Abschn. 11.4.2, die Wahrscheinlichkeiten der einzelnen Zustände zu einem  $N$ -dimensionalen Vektor  $\mathbf{p}$  zusammen, so erhält Gl. (12.8) die Matrixform (11.48)

$$\mathcal{S} : \mathbf{p}(k+1) = \mathbf{G} \mathbf{p}(k), \quad \mathbf{p}(0) = \mathbf{p}_0, \quad (12.9)$$

die auf eine Folge von Wahrscheinlichkeitsvektoren

$$\mathbf{P} = (\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots, \mathbf{p}(k_e))$$

führt, die die vom Automaten erzeugbaren Zustandsfolgen in kompakter Form beschreibt. Demzufolge kann der Automat zum Zeitpunkt  $k$  jeden Zustand  $z$  annehmen, für den das Element  $p_z(k)$  des Vektors  $\mathbf{p}(k)$  positiv ist. Der Wert des Elements gibt außerdem die Wahrscheinlichkeit an, dass sich der Automat im betreffenden Zustand befindet. Verschwindende Elemente in diesem Vektor zeigen an, in welchem Zustand sich der Automat nicht befinden kann.

Wenn man die Wahrscheinlichkeiten  $p_z(k)$  den Zustandsknoten  $z$  des Automatengraphen zuordnet, so erhält man die für den Zeitpunkt  $k+1$  geltende Größe  $p_{z'}(k+1)$  für den Zustand  $z'$ , indem man alle zum Knoten  $z'$  führenden Kanten betrachtet, die Kantengewichte

$$g_{z'z} = \text{Prob}(Z(k+1) = z' \mid Z(k) = z)$$

mit den Wahrscheinlichkeiten  $p_z(k)$  der Anfangsknoten dieser Kanten multipliziert und die Ergebnisse addiert.<sup>1</sup> Bildhaft ausgedrückt, fließt in jedem Zeitschritt Wahrscheinlichkeit von den Anfangsknoten zu den Endknoten jeder Kante, wobei die Stärke der Flüsse durch die Anfangsknoten und die an den Kanten stehenden Übergangswahrscheinlichkeiten bestimmt wird.

### Beispiel 12.6 Verhaltensanalyse der Stanze

Zur Erläuterung der Chapman-Kolmogorov-Gleichung wird das Modell der Stanze aus Beispiel 11.7 vereinfacht, indem die Zustände 2 bis 6 nicht mehr einzeln aufgeführt werden. In dem in Abb. 12.8 gezeigten stochastischen Automaten beinhaltet der Zustandsübergang  $1 \rightarrow 7$  den gesamten Stanzvorgang, während die verbleibenden Übergänge das nichtdeterministische Verhalten auf Grund von Fehlern darstellen. Die Kanten ohne Bewertung haben eine Übergangswahrscheinlichkeit von eins.

Die Zustandsmenge des Automaten ist durch  $\mathcal{Z} = \{1, 7, 8, 9\}$  gegeben. Über diese Zustände muss die Summation in Gl. (12.8) erfolgen. Der Anfangszustand sei  $z(0) = 1$ .

Die Anwendung von Gl. (12.8) für  $k = 0$  ergibt für  $z' = 1$  ausführlich geschrieben die Aufenthaltswahrscheinlichkeit

$$\begin{aligned} \text{Prob}(Z(1) = 1) &= \text{Prob}(Z(k) = 1 \mid Z(0) = 1) \cdot \text{Prob}(Z(0) = 1) \\ &\quad + \text{Prob}(Z(k) = 1 \mid Z(0) = 7) \cdot \text{Prob}(Z(0) = 7) \\ &\quad + \text{Prob}(Z(k) = 1 \mid Z(0) = 8) \cdot \text{Prob}(Z(0) = 8) \\ &\quad + \text{Prob}(Z(k) = 1 \mid Z(0) = 9) \cdot \text{Prob}(Z(0) = 9) \\ &= 0 \cdot 1 + 0,9985 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 = 0. \end{aligned}$$

Für die anderen Zustände erhält man mit den im Abschn. 11.4.2 eingeführten Abkürzungen die folgenden Wahrscheinlichkeiten:

<sup>1</sup> Hier gelten also dieselben Regeln wie bei dem im Abschn. 2.2.3 eingeführten Signalflussgraphen.



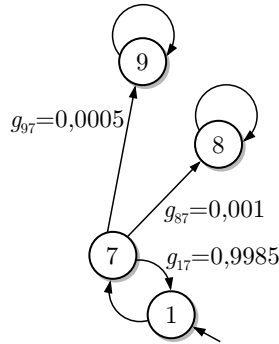


Abb. 12.8: Vereinfachtes Modell der Stanze

$$\begin{aligned} \text{für } z' = 7 : \quad p_7(1) &= g_{71}p_1(0) + g_{77}p_7(0) + g_{78}p_8(0) + g_{79}p_9(0) \\ &= 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 = 1 \end{aligned}$$

$$\begin{aligned} \text{für } z' = 8 : \quad p_8(1) &= g_{81}p_1(0) + g_{87}p_7(0) + g_{88}p_8(0) + g_{89}p_9(0) \\ &= 0 \cdot 1 + 0,001 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 = 0 \end{aligned}$$

$$\begin{aligned} \text{für } z' = 9 : \quad p_9(1) &= g_{91}p_1(0) + g_{97}p_7(0) + g_{98}p_8(0) + g_{99}p_9(0) \\ &= 0 \cdot 1 + 0,0005 \cdot 0 + 0 \cdot 0 + 1 \cdot 0 = 0. \end{aligned}$$

Der Automat geht also mit Wahrscheinlichkeit eins in den Zustand 7 über:

$$\mathbf{p}(1) = \begin{pmatrix} p_1(1) \\ p_7(1) \\ p_8(1) \\ p_9(1) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

Im nächsten Zeitschritt erhält man auf dieselbe Weise

$$\begin{aligned} \text{für } z' = 1 : \quad p_1(2) &= g_{11}p_1(1) + g_{17}p_7(1) + g_{18}p_8(1) + g_{19}p_9(1) \\ &= 0 \cdot 0 + 0,9985 \cdot 1 + 0 \cdot 0 + 0 \cdot 0 = 0,9985 \end{aligned}$$

$$\begin{aligned} \text{für } z' = 7 : \quad p_7(2) &= g_{71}p_1(1) + g_{77}p_7(1) + g_{78}p_8(1) + g_{79}p_9(1) \\ &= 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 = 0 \end{aligned}$$

$$\begin{aligned} \text{für } z' = 8 : \quad p_8(2) &= g_{81}p_1(1) + g_{87}p_7(1) + g_{88}p_8(1) + g_{89}p_9(1) \\ &= 0 \cdot 0 + 0,001 \cdot 1 + 1 \cdot 0 + 0 \cdot 0 = 0,001 \end{aligned}$$

$$\begin{aligned} \text{für } z' = 9 : \quad p_9(2) &= g_{91}p_1(1) + g_{97}p_7(1) + g_{98}p_8(1) + g_{99}p_9(1) \\ &= 0 \cdot 0 + 0,0005 \cdot 1 + 0 \cdot 0 + 1 \cdot 0 = 0,0005, \end{aligned}$$

d. h., der Zustand 1 wird mit der Wahrscheinlichkeit 0,9985 angenommen und die Zustände 8 und 9 mit den Wahrscheinlichkeiten 0,001 bzw. 0,0005.  $\square$

### 12.4.2 Strukturelle Analyse stochastischer Automaten

Die für deterministische Automaten im Abschn. 12.2.4 eingeführte und auf nichtdeterministische Automaten erweiterte Klassifikation der Automatenzustände kann ohne Veränderungen auf stochastische Automaten übernommen werden. Wenn der Automatengraph reduzierbar ist, gibt es eine Teilmenge  $\mathcal{Z}_i$  der Zustandsmenge  $\mathcal{Z}$ , in die der Automat für hinreichend lange Zeit vollständig übergeht. Dies sieht man am einfachsten an der Matrixdarstellung des stochastischen Automaten. Wenn der Automatengraph reduzierbar ist, so ist auch die Matrix  $G$  in Gl. (12.9) reduzierbar, d. h., sie kann durch Vertauschen von Zeilen und Spalten (was einem Ummummern der Zustände entspricht) in eine Blockdreiecksform gebracht werden, die für einen in zwei Teilgraphen zerlegbaren Automatengraphen der Umformung in

$$G = \begin{pmatrix} G_{11} & O \\ G_{21} & G_{22} \end{pmatrix}$$

entspricht. Von dem zur Teilmatrix  $G_{11}$  gehörenden Teilgraphen gehen Kanten in den zu  $G_{22}$  gehörenden Teilgraphen, aber nicht umgekehrt. Deshalb kann nur „Wahrscheinlichkeitsmasse“ vom ersten in den zweiten Teilgraphen „fließen“, so dass sich der Automatenzustand nach hinreichend langer Zeit mit Wahrscheinlichkeit eins in dem zu  $G_{22}$  gehörenden Teilgraphen befindet. Die Zustandsmenge dieses Teilgraphen ist ergodisch.

Gibt es einen absorbierenden Zustand des stochastischen Automaten, so befindet sich der Automat für hinreichend große Zeit  $k$  mit Wahrscheinlichkeit eins in diesem Zustand. Transiente Zustände werden zeitweise mit einer positiven Wahrscheinlichkeit angenommen, aber für hinreichend große Zeit ist die Aufenthaltswahrscheinlichkeit in diesen Zuständen gleich null.

### 12.4.3 Erweiterung auf Automaten mit Eingang und Ausgang

Wenn man sich bei einem stochastischen Automaten nur für die Zustandsfolge interessiert, so kann man die Chapman-Kolmogorov-Gleichung unter Verwendung der Zustandsübergangswahrscheinlichkeit  $G(z' | z, v)$  direkt erweitern. Für eine gegebene Eingabefolge

$$V(0...k) = (v_0, v_1, \dots, v_k) \quad (12.10)$$

bestimmt man dann die bedingte Wahrscheinlichkeit

$$\text{Prob}(Z(k+1) = z' | V(0) = v_0, V(1) = v_1, \dots, V(k) = v_k), \quad (12.11)$$

mit der der Automat zum Zeitpunkt  $k + 1$  den Zustand  $z'$  annimmt. Die Wahrscheinlichkeit (12.11) wird kürzer als  $\text{Prob}(z' | V(0...k))$  geschrieben. Es gilt

$$\begin{aligned} \text{Prob}(z' | V(0...k)) &= \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k+1) = z' | Z(k) = z, V(k) = v_k) \cdot \text{Prob}(z | V(0...k-1)) \\ &= \sum_{z \in \mathcal{Z}} G(z' | z, v_k) \cdot \text{Prob}(z | V(0...k-1)), \end{aligned} \quad (12.12)$$

wobei hier und in den folgenden Gleichungen  $z'$  stets den Zustand zum Zeitpunkt  $k + 1$  und  $z$  den Zustand zur Zeit  $k$  bezeichnet. Gleichung (12.12) ist eine rekursive Darstellung, bei der die Aufenthaltswahrscheinlichkeit im Zustand  $z'$  zur Zeit  $k + 1$  aus der Aufenthaltswahrscheinlichkeit aller Zustände  $z \in \mathcal{Z}$  zum Zeitpunkt  $k$  bestimmt wird. Da die Zustände  $Z(k)$  nur von den Eingaben bis zum Zeitpunkt  $k - 1$  abhängen, steht im Bedingungsteil auf der rechten Seite die kürzere Eingabefolge  $V(0 \dots k - 1)$ .

Wie beim autonomen stochastischen Automaten können die auf diese Weise erhaltenen Wahrscheinlichkeiten in Vektoren  $\mathbf{p}(k)$  und diese zur Folge

$$\mathbf{P} = (\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots, \mathbf{p}(k_e))$$

zusammengefasst werden. Das Element  $p_z(k)$  des Vektors  $\mathbf{p}(k)$  ist dabei gleich der Wahrscheinlichkeit  $\text{Prob}(z \mid V(0 \dots k - 1))$ . Gleichung (12.12) wird dann in der Form

$$\mathcal{S} : \mathbf{p}(k + 1) = \mathbf{G}(v_k) \cdot \mathbf{p}(k) \quad (12.13)$$

geschrieben, in der  $\mathbf{G}(v)$  die Matrix der Zustandsübergangswahrscheinlichkeit

$$G(z' \mid z, v) = \text{Prob}(Z(1) = z' \mid Z(0) = z, V(0) = v)$$

für die Eingabe  $v$  darstellt. Bis auf die Tatsache, dass sich diese Matrix in Abhängigkeit von der aktuellen Eingabe  $v$  ändert, sind die Berechnungsvorschriften (12.12) und (12.13) und folglich die Eigenschaften stochastischer Automaten mit Eingängen dieselben wie die autonomer stochastischer Automaten (vgl. mit Gln. (12.8) und (12.9)).

Die Berechnung des Verhaltens wird etwas komplizierter, wenn mit den Zustandsübergängen die erzeugten Ausgaben bestimmt werden sollen. Man muss dann mit der Verhaltensrelation  $L$  arbeiten und die Verbundwahrscheinlichkeit

$$\text{Prob}(Z(k + 1) = z', W(k) = w \mid V(0) = v_0, V(1) = v_1, \dots, V(k) = v_k) \quad (12.14)$$

für das gemeinsame Auftreten des Nachfolgezustands  $z'$  und der Ausgabe  $w$  für den Fall berechnen, dass der Automat die Eingabefolge (12.10) erhält. Diese Wahrscheinlichkeit wird durch  $\text{Prob}(z', w \mid V(0 \dots k))$  abgekürzt. Es gilt

$$\begin{aligned} & \text{Prob}(z', w \mid V(0 \dots k)) \\ &= \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k + 1) = z', W(k) = w \mid Z(k) = z, V(k) = v_k) \cdot \text{Prob}(z \mid V(0 \dots k - 1)) \\ &= \sum_{z \in \mathcal{Z}} L(z', w \mid z, v_k) \cdot \text{Prob}(z \mid V(0 \dots k - 1)). \end{aligned}$$

Dies ist eine rekursive Darstellung, wenn man beachtet, dass man die Zustandswahrscheinlichkeit  $\text{Prob}(z \mid V(0 \dots k - 1))$  für den Zustand  $Z(k) = z$  als Randwahrscheinlichkeit

$$\begin{aligned} \text{Prob}(z \mid V(0 \dots k - 1)) &= \sum_{w \in \mathcal{W}} \text{Prob}(Z(k) = z, W(k - 1) = w \mid V(0 \dots k - 1)) \\ &= \sum_{w \in \mathcal{W}} \text{Prob}(z, w \mid V(0 \dots k - 1)) \end{aligned}$$

aus der im vorherigen Berechnungsschritt ermittelten Wahrscheinlichkeitsverteilung

$$\text{Prob}(z, w \mid V(0 \dots k-1))$$

erhält. Das Ergebnis sind Folgen von Zuständen und den dazugehörigen Ausgaben, für die in Erweiterung derartiger Folgen nichtdeterministischer Automaten in Gl. (12.7) jetzt auch die Auftretenswahrscheinlichkeit ermittelt wurde.

<b>Aufgabe 12.2*</b> <i>Verhalten eines Regensensors</i>
--

Bestimmen Sie den Zustand des Regensensors aus Aufg. 11.12 auf S. 385 für den Zeitpunkt  $k = 1$  unter folgenden Bedingungen:

- (a)  $\text{Prob}(Z(0) = 1) = 1, \quad v(0) = 2$
- (b)  $\text{Prob}(Z(0) = 1) = 0,5, \quad v(0) = 2$

Wird der Regen richtig erkannt?

□

## 12.5 Verhalten von Petrinetzen

### 12.5.1 Berechnung der Markierungsfolgen

Das Verhalten von Petrinetzen wird durch Markierungsfolgen beschrieben, die ausgehend von einer Anfangsmarkierung durch Schalten aktivierter Transitionen entstehen. In diesen Folgen stehen die Markierungsvektoren  $\mathbf{p}(k)$  für  $k = 0, 1, \dots, k_e$ :

$$\mathbf{P}(0 \dots k_e) = (\mathbf{p}(0), \mathbf{p}(1), \mathbf{p}(2), \dots, \mathbf{p}(k_e)).$$

Dabei ist es für Petrinetze typisch, dass mehrere Markierungsfolgen auftreten, weil Petrinetze vor allem zur Beschreibung nichtdeterministischer Systeme eingesetzt werden.

Für kleine Beispiele ermittelt man die Markierungsfolgen häufig aus der grafischen Darstellung des Petrinetzes, weil dies anschaulich ist und gleichzeitig die Interpretation der erhaltenen Folgen erleichtert. Für größere Netze verwendet man die analytische Netzbeschreibung (11.64)

$$\mathcal{PN}: \begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{N}\mathbf{t}(k) & \text{mit } \mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k), \mathbf{v}(k)) \\ \mathbf{p}(0) = \mathbf{p}_0. \end{cases}$$

Die bei nichtdeterministischen Automaten angewendete Methode, alle möglichen Nachfolgezustände gleichzeitig zu markieren, ist bei Petrinetzen nicht anwendbar, weil der Zustand des Netzes durch den gesamten Markierungsvektor und nicht durch eine einzelne markierte Stelle gekennzeichnet ist. Beim nichtdeterministischen Automaten bedeutet ein Vektor  $\mathbf{p}(k)$ , bei dem mehrere Elemente mit einer Eins „markiert“ sind, dass der Automat zum Zeitpunkt  $k$  in einem der betreffenden Zustände ist. Beim Petrinetz ist jedoch nicht erkennbar, wie sich eine Menge markierter Stellen aus Teilmengen der möglichen Markierungen zusammensetzt. Man muss die möglichen Schaltfolgen deshalb nacheinander betrachten, um die entsprechenden Markierungsfolgen zu erzeugen.

### 12.5.2 Strukturelle Analyse von Petrinetzen

Im Erreichbarkeitsgraphen  $\mathcal{E}(\mathcal{M}, \mathcal{T})$  wird dargestellt, welche Markierungsfolgen im Petrinetz möglich sind. Die Knoten  $M \in \mathcal{M}$  des Erreichbarkeitsgraphen beschreiben die Markierungen  $M$ , die entweder durch Aufzählung der markierten Stellen oder durch den Markierungsvektor  $\mathbf{p}$  notiert werden. Die gerichteten Kanten  $(M', M) \in \mathcal{T}$  des Graphen sind mit derjenigen Transition  $t$  bewertet, durch deren Schalten die Markierung  $M$  des Anfangsknotens der Kante in die Markierung  $M'$  am Ende der Kante übergeht:

$$M \xrightarrow{t} M'.$$

Bei Petrinetzen mit Eingängen wird an die Kanten auch die Eingabe geschrieben, unter der die entsprechende Transition aktiviert ist.

Der Erreichbarkeitsgraph zeigt, welche unterschiedlichen Zustände das Petrinetz ausgehend von der Anfangsmarkierung annehmen kann. Auf Grund der Möglichkeit, in einem Petrinetz parallel ablaufende Prozesse in sehr kompakter Weise zu beschreiben, ist der Erreichbarkeitsgraph i. Allg. viel größer als das Petrinetz selbst.

Der Erreichbarkeitsgraph kann als nichtdeterministischer Automat interpretiert werden, der dasselbe Verhalten hat wie das Petrinetz. Anstatt den Markenfluss im Petrinetz zu betrachten, sieht man sich die im Erreichbarkeitsgraphen möglichen Zustandsübergänge an.

Auf Grund dieser Äquivalenz von Erreichbarkeitsgraph und Automat kann man die für Automaten im Abschn. 12.2.4 eingeführte strukturelle Analyse auf den Erreichbarkeitsgraphen anwenden. Dementsprechend nennt man eine Markierung  $\bar{M}$  von einer Anfangsmarkierung  $M_0$  aus erreichbar, wenn es im Erreichbarkeitsgraphen einen Pfad von  $M_0$  nach  $\bar{M}$  gibt. Alle derartigen Markierungen einschließlich der Anfangsmarkierung  $M_0$  selbst fasst man zur *Erreichbarkeitsmenge*  $\mathcal{R}(M_0)$  zusammen.

Bestimmt man die Menge der mit der Anfangsmarkierung stark zusammenhängenden Markierungen, so erkennt man, ob der Erreichbarkeitsgraph reduzierbar ist. Ist dies nicht der Fall, so verhält sich das System zyklisch und alle Zustände sind rekurrent, gegebenenfalls sogar periodisch. Andernfalls kann man den Erreichbarkeitsgraphen in stark zusammenhängende Teilgraphen zerlegen, von denen mindestens einer ergodisch ist. Das Petrinetz bewegt sich in diesen Teilgraphen hinein und bleibt dort. Man nennt diesen Teilgraphen auch *Senke*.

**Lebendigkeit von Petrinetzen.** Analysiert man den Erreichbarkeitsgraphen bezüglich der an den Kanten vorkommenden Transitionen, so kann man *tote Transitionen* erkennen. Dies sind Transitionen, die niemals schalten und deshalb an keiner Kante des Erreichbarkeitsgraphen vorkommen. Derartige Transitionen kann man nicht unmittelbar im Petrinetz identifizieren, weil das Schalten der Transitionen entsprechend der Schaltregel davon abhängt, ob das Netz von der Anfangsmarkierung aus zu einer Markierung kommt, bei der die Transitionen aktiviert sind. Im Erreichbarkeitsgraphen sind die toten Transitionen jedoch erkennbar.

*Tote Markierungen* bzw. *Verklümmungen* sind Markierungen, von denen im Erreichbarkeitsgraphen keine Kanten abgehen. Bei diesen Markierungen gibt es also keine aktivierte Transition. Wenn das Petrinetz diese Zustände annimmt, bleibt es „stehen“ und erzeugt keine weiteren Ereignisse. Besitzt der Erreichbarkeitsgraph des Petrinetzes keine derartigen Knoten, so heißt das Petrinetz *lebendig*. Die Lebendigkeit des Netzes bedeutet, dass es zu jeder Situation eine Nachfolgesituation gibt, die vom Netz erzeugte Zustandsfolge also niemals abreißt.

Die Lebendigkeit des Netzes ist nicht gleichbedeutend damit, dass es keine toten Transitionen gibt, weil das Weiterschalten des Netzes auch mit einer Teilmenge von Transitionen möglich ist. Allerdings gibt es auch strengere Definitionen der Lebendigkeit als die hier verwendete, die das Vorhandensein von toten Transitionen verbieten.

### Beispiel 12.7 Erreichbarkeitsgraph eines Petrinetzes

Das in Abb. 11.30 auf S. 390 gezeigte Petrinetz soll die Anfangsmarkierung besitzen, in der nur die Stelle  $p_1$  markiert ist. Bei der Bestimmung des Erreichbarkeitsgraphen wird nun untersucht, welche Transitionen nacheinander schalten können. Durch die Anfangsmarkierung ist nur die Transition  $t_1$  aktiviert, durch deren Schalten die Stelle  $p_2$  markiert wird. Dann erhalten durch Schalten der Transition  $t_2$  die Stellen  $p_3$  und  $p_4$  je eine Marke.

Das weitere Schalten ist nichtdeterministisch, weil sowohl die Transition  $t_3$  als auch die Transition  $t_4$  aktiviert ist. In der Reihenfolge des Schaltens der Transitionen wandern die Marken von der Stelle  $p_3$  zur Stelle  $p_5$  bzw. von  $p_4$  nach  $p_6$ . Man kann auch den Fall in die Betrachtungen einbeziehen, dass beiden Transitionen gleichzeitig schalten, was einem Ereignis entspricht, das durch das gleichzeitige Schalten zweier Transitionen repräsentiert wird. Das Ergebnis sind drei verschiedene Markierungen, die in Abb. 12.9 in den drei vom Knoten mit den Markierungen  $p_3, p_4$  ausgehenden Kanten dargestellt sind. Durch weiteres Schalten erhält man den rechten Teil des gezeigten Erreichbarkeitsgraphen. Der Graph ist zyklisch, weil nach dem Schalten der Transition  $t_6$  wieder die Anfangsmarkierung entsteht.

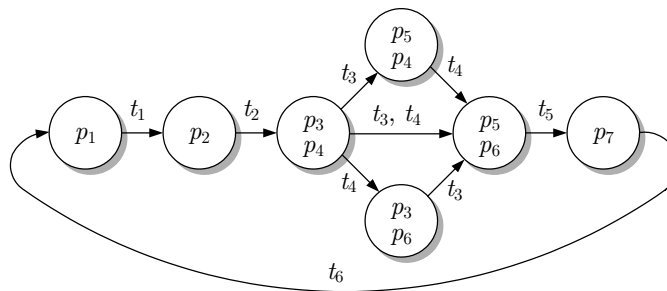


Abb. 12.9: Erreichbarkeitsgraph des Petrinetzes

Der in Abb. 12.9 gezeigte Erreichbarkeitsgraph kann als Automatengraph des zum Petrinetz äquivalenten Automaten interpretiert werden. Der Graph ist irreduzibel. Er enthält folglich keine toten Markierungen, so dass das betrachtete Petrinetz lebendig ist.  $\square$

### 12.5.3 Invarianten

Die bisherigen Eigenschaften von Petrinetzen wurden anhand des Erreichbarkeitsgraphen definiert und analysiert. Erreichbarkeitsgraphen sind aber i. Allg. sehr umfangreich und ihre Analyse deshalb aufwändig. Als alternativer Weg der Petrinetzanalyse werden im Folgenden Netzinvarianten definiert, die sich auf die analytische Darstellung des Petrinetzes beziehen. Sie werden nur bei autonomen Petrinetzen verwendet und sind Eigenschaften der Netzmatrix  $N$ .

**T-Invarianten.** *Transitionsinvarianten* (T-Invarianten) sind Mengen von Transitionen, durch deren Schalten eine Markierung reproduziert wird. In Gl. (11.63)

$$\mathcal{PN} : \begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{N} \mathbf{t}(k) & \text{mit } \mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k)) \\ \mathbf{p}(0) = \mathbf{p}_0 \end{cases} \quad (12.15)$$

soll also gelten

$$\mathbf{p}(\bar{k}) = \mathbf{p}(0),$$

wobei  $\bar{k}$  die Anzahl der Zustandsübergänge bezeichnet, nach der die Markierung  $\mathbf{p}(0)$  wieder auftritt. Aus Gl. (12.15) erhält man

$$\mathbf{p}(\bar{k}) = \mathbf{p}(0) + \mathbf{N} \sum_{k=0}^{\bar{k}-1} \mathbf{t}(k).$$

Für das Petrinetz muss folglich die Beziehung

T-Invariante  $\mathbf{t}$ :  $\mathbf{N} \mathbf{t} = \mathbf{0}$

(12.16)

gelten, wobei sich der Vektor  $\mathbf{t}$  entsprechend

$$\mathbf{t} = \sum_{k=0}^{\bar{k}-1} \mathbf{t}(k) \quad (12.17)$$

aus den  $\bar{k}$  Schaltvektoren  $\mathbf{t}(k)$  zusammensetzt. Vektoren  $\mathbf{t}$ , die die Bedingung (12.16) erfüllen, heißen T-Invarianten.

Man interessiert sich für die ganzzahligen Lösungen dieser Gleichung, die nicht als Linearkombination anderer ganzzahliger Lösungen dargestellt werden können, und spricht von den elementaren Lösungen der Gl. (12.16). Mit zwei oder mehreren T-Invarianten erfüllt nämlich auch deren Linearkombination (mit ganzzahligen Wichtungsfaktoren) die Bedingung (12.16). Das heißt, es können sich im Petrinetz mehrere Zyklen überlagern.

Entsprechend Gl. (12.17) zeigt das  $i$ -te Element der T-Invarianten  $\mathbf{t}$ , wie oft die Transition  $t_i$  schalten muss, damit die Markierung  $\mathbf{p}(0)$  reproduziert wird. T-Invarianten beschreiben also Schleifen im Erreichbarkeitsgraphen des Petrinetzes. Wenn das Petrinetz einen Regelkreis beschreibt, der zyklisch bestimmte Prozessstufen durchlaufen soll (wie z. B. eine Stanze), dann kann anhand der T-Invarianten untersucht werden, ob die Steuerung richtig entworfen wurde und den gewünschten Zyklus erzeugt.

Gleichung (12.16) wurde abgeleitet, ohne auf die Bedingung

$$\mathbf{t}(k) \in \mathcal{T}_{\text{akt}}(\mathbf{p}(k))$$

zu achten. Für jede berechnete T-Invariante muss deshalb noch überprüft werden, ob es eine Schaltfolge  $\mathbf{t}(0), \mathbf{t}(1), \dots, \mathbf{t}(\bar{k})$  gibt, die die Schaltregel erfüllt und mit der die T-Invariante in der Form (12.17) dargestellt werden kann. Die Existenz einer oder mehrerer T-Invarianten ist

also nur eine notwendige Bedingung dafür, dass eine Anfangsmarkierung  $p(0)$  nach  $\bar{k}$  Schritten reproduziert wird.

Deshalb bedeutet die Existenz einer T-Invarianten auch nicht, dass beliebige Anfangsmarkierungen reproduziert werden, sondern man muss mit der zu einer T-Invarianten gehörenden Schaltfolge auch die Anfangsmarkierungen bestimmen, für die dies der Fall ist.

Zwischen der Existenz von T-Invarianten und der Lebendigkeit des Petrinetzes besteht ein direkter Zusammenhang. Damit das Netz lebendig ist, also sich unendlich lange bewegt, muss es mindestens eine Transitionenmenge geben, nach deren Schalten eine vorher bereits erhaltene Markierung wiederhergestellt ist.

|| Jedes lebendige Netz besitzt mindestens eine T-Invariante.

Andererseits gilt: wenn man feststellt, dass das Netz keine T-Invariante besitzt, so weiß man, dass es nicht lebendig ist. Der Vollständigkeit halber sei hinzugefügt, dass diese Aussage streng genommen nur gilt, wenn das Netz beschränkt ist, d. h., wenn sich die Markierungen nach einer *endlichen* Zahl von Ereignissen wiederholen.

**S-Invarianten.** *Stelleninvarianten* (S-Invarianten) sind Mengen von Stellen mit einer bei jeder Markierung konstanten Markenzahl. Es wird nicht gefordert, dass jede Stelle einzeln eine konstante Markenanzahl aufweist, sondern dass die Gesamtzahl der Marken in der betrachteten Stellenmenge beim Schalten konstant bleibt. Beim Zusammenzählen der Stellen können die Stellenmarkierungen auch mit einem ganzzahligen Gewicht multipliziert werden. Folglich muss es einen Zeilenvektor  $s^T$  geben, mit dem die Beziehung

$$s^T p(k+1) = s^T p(k), \quad k = 0, 1, 2, \dots$$

gilt, was entsprechend Gl. (12.15) der Fall ist, wenn die Bedingung

S-Invariante  $s$ :  $s^T N = 0^T$

(12.18)

erfüllt ist. Der Vektor  $s$  heißt dann S-Invariante.

Das Element  $s_i$  des Vektors  $s$  gibt das Gewicht an, mit dem die Markierung der Stelle  $p_i$  in die Summenbildung eingeht. Mit der Gewichtung kann dem Umstand Rechnung getragen werden, dass sich beim Schalten von Transitionen mit unterschiedlicher Anzahl von Prä- und Poststellen die Anzahl der Marken im Netz ändert. So liegt nach dem Schalten einer Transition mit zwei Prästellen und einer Poststelle an Stelle der beiden Marken der Prästellen nur noch eine Marke in der Poststelle. Wenn man aber die Poststelle mit zwei gewichtet, bleibt die gewichtete Markensumme bei diesem Schalten konstant.

Die Menge aller Stellen, deren Markierung mit einem positiven Gewicht in die Summenbildung eingeht, wird als Trägermenge bezeichnet. Sie ist durch die von null verschiedenen Elemente des Vektors  $s$  festgelegt.

S-Invarianten kann man verwenden, um den ordnungsgemäßen Ablauf eines Prozesses zu überwachen. Ändert sich die gewichtete Markensumme  $s^T p(k)$ , so lässt dies auf einen Fehlerfall schließen. Im Kap. 14 werden S-Invarianten eingesetzt, um Steuerungen zu entwerfen, die vorgegebenen Güteforderungen genügen.



**Beispiel 12.8** *Invarianten des Petrinetzes zur Beschreibung der Werkzeugmaschinen*

Das in Abb. 11.36 auf S. 397 gezeigte Petrinetz stellt die Arbeitsweise von vier Werkzeugmaschinen dar. Es hat die Netzmatrix

$$N = \begin{matrix} & \begin{matrix} t_0 & t_1 & t_2 & t_3 & t_4 & t_5 \end{matrix} \\ \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{matrix} & \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{pmatrix} \end{matrix},$$

deren T-Invarianten aus der Beziehung

$$N\mathbf{t} = \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{pmatrix} = \mathbf{0}$$

bestimmt werden, wobei die Elemente des Vektors  $\mathbf{t}$  den Bezeichnungen der Transitionen im Petrinetz entsprechen. Durch Ausmultiplizieren erhält man die folgenden vier Gleichungen

$$t_0 = t_1 + t_2 \quad (12.19)$$

$$t_1 = t_3 \quad (12.20)$$

$$t_2 = t_4 \quad (12.21)$$

$$t_5 = t_3 + t_4, \quad (12.22)$$

aus denen die sechs Unbekannten  $t_0, \dots, t_5$  zu bestimmen sind. Man erhält zwei Lösungen:

$$t_0 = 1, \quad t_1 = 1, \quad t_2 = 0, \quad t_3 = 1, \quad t_4 = 0, \quad t_5 = 1 \quad (12.23)$$

$$t_0 = 1, \quad t_1 = 0, \quad t_2 = 1, \quad t_3 = 0, \quad t_4 = 1, \quad t_5 = 1. \quad (12.24)$$

Die Gln. (12.19) – (12.22) sowie die erhaltenen Invarianten haben folgende Interpretationen. Die Gln. (12.19) und (12.22) besagen, dass zur Reproduktion einer Markierung die Transitionen  $t_0$  und  $t_5$  genauso oft schalten müssen wie die Transitionen  $t_1$  und  $t_2$  bzw.  $t_3$  und  $t_4$  zusammen, was man anhand des Markenflusses durch die Stellen  $p_1$  und  $p_4$  leicht sieht. Andererseits müssen zur Reproduktion einer Markierung die im selben Teilprozess liegenden Transitionen  $t_1$  und  $t_3$  bzw.  $t_2$  und  $t_4$  gleich oft schalten.

Die erhaltenen T-Invarianten geben zwei Möglichkeiten der Reproduktion der Markierung an. Die T-Invariante (12.23) beschreibt einen Markenfluss über  $p_1, p_2$  und  $p_4$ , die andere Invariante einen Markenfluss über  $p_1, p_3$  und  $p_4$ . Dass man die Anfangsmarkierung auch dadurch reproduzieren kann, dass man beide Pfade des Petrinetzes nutzt, wird durch die Summe beider T-Invarianten

$$t_0 = 2, \quad t_1 = 1, \quad t_2 = 1, \quad t_3 = 1, \quad t_4 = 1, \quad t_5 = 2$$

beschrieben, die ebenfalls die Gl. (12.16) erfüllt. Wie man aus dem Petrinetz sehen kann, wird die Schaltregel für beide T-Invarianten erfüllt, wenn bei der Anfangsmarkierung nur die Stelle  $p_1$  eine Marke besitzt. Andererseits reproduziert die T-Invariante (12.23) auch die Anfangsmarkierung, bei der nur die Stelle  $p_2$  eine Marke besitzt. Diese Markierung wird nicht durch die zweite T-Invariante reproduziert, weil für sie die Schaltregel nicht erfüllt ist.

Zur Berechnung der S-Invarianten des Netzes muss man die Gleichung

$$(s_1 \ s_2 \ s_3 \ s_4) \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{pmatrix} = \mathbf{0}^T$$

lösen. Da die Netzmatrix  $N$  den Rang 4 hat, gibt es nur die triviale Lösung

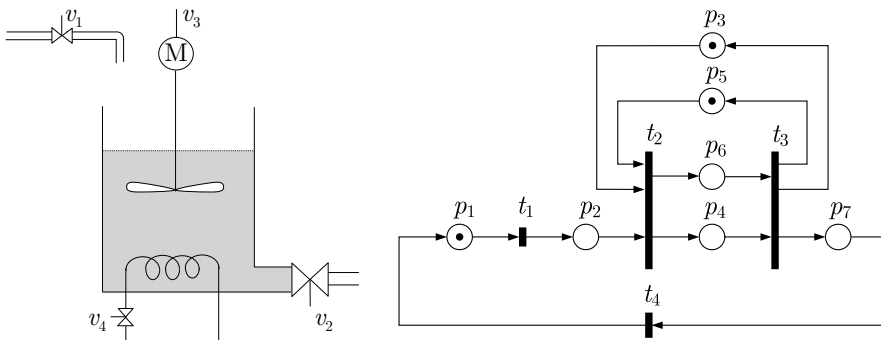
$$s_1 = s_2 = s_3 = s_4 = 0,$$

also keine S-Invariante. Das heißt, es gibt keine Menge von Stellen, deren Markenzahl beim Schalten des Netzes konstant bleibt. Das ist dadurch zu erklären, dass bei diesem Netz der Markenfluss den Fluss der Werkstücke symbolisiert, die die Werkzeugmaschinen nach der Bearbeitung verlassen. Die Marken werden nicht in einer Menge von Stellen „gehalten“, sondern kommen durch die Transition  $t_0$  ins Netz hinein und verlassen das Netz über die Transition  $t_5$  wieder.

Dieses Beispiel verdeutlicht, dass die Verwendung von Invarianten zur Beschreibung von Zyklen des Erreichbarkeitsgraphen wesentlich einfacher ist, als die Berechnung und die Analyse des Erreichbarkeitsgraphen selbst, der dem in Abb. 11.18 auf S. 365 gezeigten Automatengraphen zur Beschreibung der Arbeitsweise der Werkzeugmaschinen entspricht.  $\square$

### Beispiel 12.9 Invarianten eines Petrinetzes zur Beschreibung eines Batchprozesses

Der in Abb. 12.10 links gezeigte Reaktor soll in einem Batchprozess verwendet werden, um eine Flüssigkeit zu erhitzen. Der Behälter wird gefüllt, der Rührer angeschaltet, die Flüssigkeit mit der Heizung auf eine Solltemperatur gebracht, der Rührer abgeschaltet und der Behälter wieder entleert. Dafür werden die diskreten Eingangsgrößen  $v_1$ ,  $v_2$ ,  $v_3$  und  $v_4$  entsprechend geschaltet.



**Abb. 12.10:** Batchprozess (links) und Petrinetz zur Beschreibung des gesteuerten Batchprozesses (rechts)

Für die Beschreibung des Batchprozesses durch ein Petrinetz werden folgende Situationen definiert:

Stelle	Bedeutung
$p_1$	Der Behälter ist leer.
$p_2$	Der Behälter ist mit kalter Flüssigkeit gefüllt.
$p_3$	Der Rührer ist ausgeschaltet.
$p_4$	Der Rührer ist eingeschaltet.
$p_5$	Die Heizung ist ausgeschaltet.
$p_6$	Die Heizung ist eingeschaltet.
$p_7$	Der Behälter ist mit erhitzter Flüssigkeit gefüllt.

Die Transitionen haben folgende Bedeutung:

Transition	Bedeutung
$t_1$	Das Füllen des Behälters ist abgeschlossen.
$t_2$	Die Heizung und der Rührer werden eingeschaltet.
$t_3$	Die Heizung und der Rührer werden ausgeschaltet.
$t_4$	Das Entleeren des Behälters ist abgeschlossen.

Die angegebene Prozessfolge wird durch das im rechten Teil von Abb. 12.10 gezeigte Petrinetz beschrieben. Es ist ein Synchronisationsgraph mit der Netzmatrix

$$N = \begin{matrix} & \begin{matrix} t_1 & t_2 & t_3 & t_4 \end{matrix} \\ \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \end{matrix} & \begin{pmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} \end{matrix}.$$

Zur Bestimmung der T-Invarianten ist die Gleichung

$$\begin{pmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{pmatrix} = \mathbf{0}$$

zu lösen, aus der man die Beziehung

$$t_1 = t_2 = t_3 = t_4$$

erhält. Die einzige T-Invariante heißt

$$t_1 = 1, \quad t_2 = 1, \quad t_3 = 1, \quad t_4 = 1.$$

Der Ausgangszustand des Prozesses, der durch die in Abb. 12.10 (rechts) markierten Stellen beschrieben ist, wird nach einem einmaligen Schalten aller vier Transitionen reproduziert. Da der Prozess zyklisch abläuft, ist dies das erwartete Ergebnis.

Die Anfangsmarkierung, bei der nur die Stelle  $p_1$  eine Marke besitzt, kann nicht reproduziert werden. Dies zeigt, dass das Vorhandensein von T-Invarianten nicht auf die Reproduzierbarkeit beliebiger Markierungen schließen lässt.

Zur Bestimmung der S-Invarianten muss man die Gleichung

$$(s_1 \ s_2 \ s_3 \ s_4 \ s_5 \ s_6 \ s_7) \begin{pmatrix} -1 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} = \mathbf{0}$$

lösen, die auf folgende Beziehungen führt:

$$\begin{aligned} s_1 &= s_2 \\ s_4 + s_6 &= s_2 + s_3 + s_5 \\ s_4 + s_6 &= s_3 + s_5 + s_7 \\ s_1 &= s_7. \end{aligned}$$

Eine Lösung ist durch

$$s_1 = s_2 = s_4 = s_7 = 1 \quad \text{und} \quad s_3 = s_5 = s_6 = 0,$$

also durch

$$\mathbf{s}^T = (1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 1)$$

gegeben. Sie besagt, dass die Stellen  $p_1, p_2, p_4$  und  $p_7$  zur Trägermenge dieser S-Invarianten gehören und die Summe der Markierungen dieser Stellen eine konstante Anzahl von Marken ergibt. Dies ist daran ersichtlich, dass die im Anfangszustand in der Stelle  $p_1$  liegende Marke über die genannten Stellen wandert. Der ordnungsgemäße Ablauf des Batchprozesses ist also dadurch gekennzeichnet, dass zu jedem Zeitpunkt genau eine der Stellen  $p_1, p_2, p_4$  und  $p_7$  markiert ist.

Eine andere Lösung ist

$$s_1 = s_2 = s_5 = s_6 = s_7 = 0 \quad \text{und} \quad s_3 = s_4 = 1$$

bzw.

$$\mathbf{s}^T = (0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0).$$

Sie besagt, dass die Stellen  $p_3$  und  $p_4$  zusammen stets dieselbe Markenanzahl besitzen. Diese Lösung betrifft den Zyklus

$$p_3 \xrightarrow{t_2} p_4 \xrightarrow{t_3} p_3,$$

der das Ein- und Ausschalten des Rührers beschreibt. Es gibt eine weitere S-Invariante mit einer ähnlichen Interpretation für die Heizung. Darüber hinaus erfüllen alle Linearkombinationen der beschriebenen S-Invarianten die Beziehung (12.18), d. h., sie beschreiben Zyklen bzw. überlagerte Zyklen.  $\square$

### Aufgabe 12.3 S- und T-Invarianten eines Petrinetzes

Berechnen Sie die S- und T-Invarianten des in Abb. 11.35 auf S. 395 gezeigten Petrinetzes. Wie verändert sich das Ergebnis, wenn anstelle der beiden Transitionen  $t_2$  und  $t_3$  nur eine Transition zwischen der Stelle  $p_2$  einerseits und den Stellen  $p_3$  und  $p_4$  andererseits liegt? Begründen Sie Ihre Ergebnisse anhand der Bedeutung, die S- und T-Invarianten für das Verhalten eines Petrinetzes haben.  $\square$

**Aufgabe 12.4\*** *Invarianten von Synchronisationsgraphen und Zustandsmaschinen*

1. Bestimmen Sie die Invarianten des Synchronisationsgraphen in Abb. 11.37 auf S. 399 und interpretieren Sie das Ergebnis.
2. Zeichnen Sie ein Petrinetz, das die Eigenschaften einer Zustandsmaschine besitzt und berechnen Sie dessen Invarianten. Interpretieren Sie Ihr Ergebnis. Was können Sie aus Ihrem Beispiel für Zustandsmaschinen im Allgemeinen ableiten?  $\square$

**Literaturhinweise**

Für automatisierungstechnische Anwendungen ist vor allem die Literatur zu ereignisdiskreten Systemen interessant, in der das Verhalten von Automaten und Petrinetzen nicht nur für den autonomen Fall, sondern auch unter dem Einfluss von Eingängen untersucht wird. Unter diesem Gesichtspunkt werden die Lehrbücher [18, 71] empfohlen. Das Verhalten von Markovketten und stochastischen Automaten wird ausführlich in [16] behandelt.



# 13

## Steuerung diskreter Systeme

*Dieses Kapitel behandelt die grundlegenden Eigenschaften gesteuerter diskreter Systeme, die dabei verwendeten Steuerungsstrukturen und die technische Realisierung von Steuereinrichtungen durch speicherprogrammierbare Steuerungen.*

### 13.1 Steuerungsaufgaben für diskrete Systeme

Bei der Steuerung diskreter Systeme wird die folgende Aufgabe betrachtet: Gegeben sind ein diskretes System mit dem Eingangssignal  $v(k)$  und dem Ausgangssignal  $w(k)$  sowie ein Steuerungsziel. Gesucht ist eine Steuereinrichtung (Steuerung), die die Eingabefolge  $V(0 \dots k_e)$  in Abhängigkeit von der gemessenen Ausgabefolge  $W(0 \dots k_e)$  so festlegt, dass das Steuerungsziel erreicht wird. Diese Aufgabe stimmt im Prinzip mit der auf S. 211 gestellten Entwurfsaufgabe überein, nur dass jetzt Güteforderungen erhoben werden, die dem diskreten Charakter der Strecke entsprechen.

Das Steuerungsziel kann in Abhängigkeit von der praktischen Aufgabenstellung sehr unterschiedlich formuliert sein. Häufig beinhaltet es ein oder mehrere Ziele der folgenden Art:

- **Vorgegebener Endzustand:** Das diskrete System soll in den Endzustand  $z_e$  überführt werden bzw. die Ausgabefolge soll mit dem Symbol  $w_e$  enden.
- **Vorgegebener Ablauf:** Das diskrete System soll eine vorgegebene Zustandsfolge  $Z(0 \dots k_e)$  durchlaufen oder eine vorgegebene Folge  $W(0 \dots k_e)$  von Ausgabesymbolen erzeugen.
- **Verbotene Zustände:** Das diskrete System darf die Zustände der vorgegebenen Menge  $\bar{Z} \subset Z$  nicht annehmen.

- **Verbotene Zustandsübergänge:** Das diskrete System darf die Zustandsübergänge (Ereignisse) einer vorgegebenen Menge  $\tilde{\mathcal{E}}$  nicht ausführen.

Wenn man diese Steuerungsziele in eine vorgegebene (formalisierte) Form bringt, bezeichnet man sie als *Spezifikation* der Steuerung (obwohl diese Festlegungen nicht die Steuerung allein, sondern das Verhalten der Strecke unter der Wirkung der Steuerung betreffen). Offenbar entstehen die ersten beiden Forderungen, dass ein System einen vorgegebenen Endzustand erreichen oder einen gegebenen Ablauf erzeugen soll, aus der Funktion des Systems, bestimmte Arbeitsschritte zu durchlaufen, während durch das Verbot von Zuständen oder Zustandsübergängen Sicherheitsanforderungen repräsentiert werden. Man unterscheidet deshalb bei den Spezifikationen auch allgemeiner zwischen Produktivitätsanforderungen und Sicherheitsanforderungen.

Die Aufgabe des Steuerungsentwurfs besteht darin, ein Steuerungsgesetz  $\Sigma_R$  bzw. Reglergesetz zu finden, mit dem das Steuerungsziel erreicht wird.

#### Steuerungsentwurf für diskrete Systeme

Gegeben:	Modell der Steuerstrecke $\Sigma_S$ Spezifikation der Steuerung
Gesucht:	Steuerungsgesetz $\Sigma_R$

**Verknüpfungssteuerungen.** Im einfachsten Fall besteht das Entwurfsergebnis aus einem Steuerungsgesetz, mit dem die Eingabe  $v(k)$  der Steuerstrecke aus der zur selben Zeit auftretenden Ausgabe  $w(k)$  ermittelt wird:

$$\Sigma_R : v(k) = H_s(w(k)). \quad (13.1)$$

Wenn unterschiedliche Endzustände  $z_e$  vorgeschrieben sind, so ist die Funktion  $H_s$  auch vom aktuell geforderten Endzustand abhängig:

$$v(k) = H_s(w(k), z_e).$$

Allgemeiner können neben einem Endzustand auch bestimmte Ausgaben oder Zustandsfolgen als Steuerungsziel auftreten, so dass man das Steuerungsziel allgemeiner durch eine Funktion  $s(k)$  repräsentiert:

$$v(k) = H_s(w(k), s(k)).$$

Steuerungen nach Gl. (13.1) bezeichnet man als Verknüpfungssteuerungen. Sie sind das Analogon zu statischen Reglern wie z. B. P-Regler (7.3).

Die drei Gleichungen zeigen, dass die Steuerung im geschlossenen Wirkungskreis arbeitet (Abb. 13.1). Es hat sich deshalb in den letzten Jahren eingebürgert, auch für diskrete Steuerungen, die im geschlossenen Wirkungskreis arbeiten, den Begriff Regelung zu verwenden. Dies wird im Folgenden immer dann getan, wenn die Arbeit im geschlossenen Wirkungskreis hervorgehoben werden soll. Viele Steuerungen für diskrete Systeme arbeiten jedoch teilweise im geschlossenen Kreis und teilweise in der offenen Kette, was unter dem allgemeineren Begriff Steuerung zusammengefasst wird (vgl. Abschn. 1.4.6).



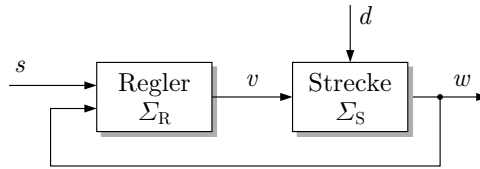


Abb. 13.1: Diskreter Regelkreis

Die Wirkung der in Abb. 13.1 eingetragenen Störung  $d$ , die wie bei kontinuierlichen Systemen die Übertragungseigenschaften der Strecke von der Eingangsgröße  $v$  auf die Ausgangsgröße  $w$  beeinflusst, wird bei ereignisdiskreten Systemen typischerweise nicht durch eine zusätzliche Eingangsgröße im Streckenmodell, sondern durch das daraus resultierende nichtdeterministische Verhalten berücksichtigt, das durch nichtdeterministische Automaten oder Petrinetze dargestellt werden kann. Die Störung führt hier zur Unkenntnis, in welchen Zustand aus einer Menge von möglichen Nachfolgezustände das System übergeht.

**Ablaufsteuerungen.** Häufig müssen Steuerungen dynamische Elemente enthalten, um das Steuerungsziel zu verwirklichen. Dann hängt die aktuelle Eingabe  $v(k)$  nicht nur von der letzten, sondern auch von vorherigen Ausgaben ab

$$\Sigma_R : v(k) = H_s(w(k), w(k-1), w(k-2), \dots, w(k-N_s)) \quad (13.2)$$

und man spricht von einer Ablaufsteuerung. An Stelle der im Steuerungsalgorithmus gespeicherten Folge

$$W(k-1 \dots k-N_s) = (w(k-1), w(k-2), \dots, w(k-N_s))$$

der letzten  $N_s$ -Ausgaben führt man den Zustand  $z_s(k)$  der Steuerung ein, mit dem man Gl. (13.2) kürzer als

$$v(k) = H_s(z_s(k), w(k))$$

schreiben kann. Um das Steuerungsziel  $s(k)$  erweitert, hat die Steuerung die Form

$$\Sigma_R : v(k) = H_s(z_s(k), w(k), s(k)).$$

Auch in den hier angegebenen Beziehungen wird die Wirkung der Steuerung im geschlossenen Kreis offensichtlich.

Wenn der aktuelle Zustand  $z(k)$  der Strecke messbar ist, erhält die Steuerung sehr viele Informationen und benötigt meist keine eigene Dynamik. Die Steuerungsaufgabe kann dann mit einer Verknüpfungssteuerung gelöst werden. Wenn jedoch die Messgröße  $w(k)$  den Prozesszustand nicht vollständig wiedergibt, muss sich die Steuerung den Verlauf  $w(k)$ ,  $w(k-1)$ ,  $w(k-2)$ , ...,  $w(k-N_s)$  vergangener Ausgaben bzw. davon abgeleitete Informationen im Zustand  $z_s(k)$  merken, wodurch eine Ablaufsteuerung entsteht.

**Vergleich von kontinuierlichen und diskreten Regelkreisen.** Die diskrete Steuereinrichtung bildet mit der Steuerstrecke einen *geschlossenen Wirkungskreis*, der als Regelkreis (oder in der älteren Literatur als Steuerkreis) bezeichnet wird (Abb. 13.1). Der Unterschied zum kontinuierlichen Regelkreis (Abb. 7.1 auf S. 211) besteht darin, dass die Signale  $v$  und  $w$ , die den

Regler mit der Regelstrecke verkoppeln, wertdiskret sind. Das hat zur Folge, dass der Kreis nur zu diskreten Zeitpunkten geschlossen wird, während der kontinuierliche Regelkreis durch einen ununterbrochenen Einfluss des Reglers auf die Strecke und umgekehrt gekennzeichnet ist. Das Steuerungsziel  $s(k)$ , das der Führungsgröße des kontinuierlichen Regelkreises entspricht, bezieht sich im diskreten Regelkreis auf diskrete Zustands- oder Ausgabewerte.

Der Vergleich der Abbildungen 13.1 und 7.1 zeigt, dass sich die kontinuierliche und diskrete Regelung nicht in Bezug auf die Wirkungsstruktur, sondern nur in der Art der für die Prozessbeobachtung und Prozessbeeinflussung dienenden Signale unterscheiden. Beide Kreise können um Steuerungskomponenten erweitert werden, die in der offenen Wirkungskette arbeiten.

## 13.2 Verknüpfungssteuerungen

Verknüpfungssteuerungen berechnen den aktuellen Wert  $v(k)$  der Stellgröße aus dem aktuellen Wert  $w(k)$  der Messgröße unter Beachtung des Steuerungsziels  $s(k)$ :

$$\text{Verknüpfungssteuerung } \Sigma_R: \quad v(k) = H_s(w(k), s(k)). \quad (13.3)$$

Für  $s(k)$  kann eine der auf S. 457 genannten Spezifikationen eingesetzt werden, beispielsweise  $s(k) = z_e$ , wenn das Steuerungsziel im Erreichen des Endzustands  $z_e$  besteht.

Verknüpfungssteuerungen sind statische Systeme, die keine eigenen Speicher besitzen und deshalb die Ausgabe  $w$  und das Steuerungsziel  $z_e$  zu jedem Zeitpunkt  $k$  in derselben Weise verarbeiten. Da sich  $w$  häufig aus mehreren Signalwerten zusammensetzt, muss die Steuerung die aktuellen Werte dieser Signale zum Wert  $v$  der Eingabe verknüpfen, was sich im Namen dieser Art der Steuerung widerspiegelt. Das Reglergesetz  $H_s$  kann als eine Tabelle geschrieben werden, die jeder Ausgabe  $w \in \mathcal{W}$  der Strecke eine Eingabe  $v \in \mathcal{V}$  zuordnet. Es kann durch ein Schaltnetz (kombinatorische Schaltung) realisiert werden.

Verknüpfungssteuerungen werden verwendet, um Sicherheitsfunktionen zu realisieren oder einfache Steuerungsaufgaben zu lösen. Beispiele sind

- die Steuerung einer Ampel an einer Garageneinfahrt, die die Ampel auf grün schaltet, wenn in der Garage mindestens ein Platz frei ist und die Garagenausfahrt nicht durch Gegenverkehr benutzt wird (Beispiel 13.1),
- die Ansteuerung einer siebenelementigen Anzeige zur Darstellung einer Zahl,
- eine Steuerung, die einen Behälter füllt, sobald der Füllstand unter einen bestimmten Wert gesunken ist.

Wenn die Signalwerte binär sind, kann man die Funktion  $H_s$  als einen booleschen Ausdruck schreiben (vgl. S. 346). Man kann die Tabelle auch als boolesche Funktion darstellen und diese Funktion entweder in eine Normalform (z. B. in die disjunktive Normalform) überführen oder mit Vereinfachungsverfahren so umformen, dass sie möglichst wenige Glieder enthält. Bis vor etwa zehn Jahren hat man sich sehr stark für diese Vereinfachungsverfahren interessiert, um ein Reglergesetz zu erhalten, das möglichst wenige Verknüpfungen enthält und deshalb in der

technischen Realisierung als Schaltnetzwerk mit möglichst wenigen Bauelementen auskommt. Derartige Vereinfachungsmethoden sind bis heute ein wichtiger Bestandteil von Lehrbüchern über Steuerungstechnik. Da man die Steuerung jetzt aber durch Rechner realisiert und das Reglergesetz programmiert, kommt es nicht mehr in erster Linie auf eine möglichst kleine Zahl von Verknüpfungen, sondern stärker auf eine gut strukturierte Darstellung an, so dass die erwähnten Vereinfachungsverfahren ihre Bedeutung verloren haben und hier nicht behandelt werden.

### Beispiel 13.1 Steuerung der Ampel an einer Garageneinfahrt

Eine Ampel an der Garageneinfahrt soll verhindern, dass Fahrzeuge in die Garage hineinfahren, wenn es entweder keinen freien Platz gibt oder wenn die Einfahrt gerade durch ein herausfahrendes Fahrzeug benutzt wird. Als Informationen erhält die Steuerung zwei binäre Größen:

$$w_1 = \begin{cases} 1 & \text{Garage ist frei} \\ 0 & \text{Garage ist belegt,} \end{cases} \quad \text{und} \quad w_2 = \begin{cases} 1 & \text{Ausfahrt ist frei} \\ 0 & \text{Ausfahrt ist belegt.} \end{cases}$$

Die Ampelsteuerung gibt eine der folgenden Eingaben vor:

$$v = \begin{cases} 1 & \text{Ampel auf grün schalten} \\ 0 & \text{Ampel auf rot schalten.} \end{cases}$$

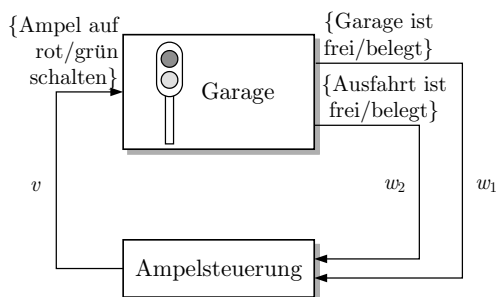


Abb. 13.2: Beispiel für eine Verknüpfungssteuerung

Die Steuereinrichtung soll die Ampel auf grün schalten, wenn die Garage und die Ausfahrt frei sind (Abb. 13.2). Dafür muss sie wie folgt funktionieren:

$$H_s = \begin{array}{|c|c|c|} \hline & v & w_1 & w_2 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ \hline \end{array}$$

Die Tabelle gibt die Funktion  $H_s$  aus Gl. (13.3) wieder. Sie kann für dieses Beispiel auch als boolesche Funktion geschrieben werden:

$$v(k) = w_1(k) \wedge w_2(k). \quad (13.4)$$

Die Steuerung ist eine Verknüpfungssteuerung, denn der Stelleingriff wird nur in Abhängigkeit von der aktuellen Ausgabe der Strecke bestimmt. Da das Steuerungsziel für alle Zeiten unverändert bleibt, erscheint es nicht explizit im Reglergesetz (13.4).

Die Steuerung (13.4) kann durch eine einfache Schaltung realisiert werden, beispielsweise durch zwei in Reihe geschaltete Relais, die bei  $w_1 = 1$  bzw.  $w_2 = 1$  ihre Kontakte schließen und damit die Ampel auf grün schalten. □

13.3 Ablaufsteuerungen

13.3.1 Reglergesetz

Ablaufsteuerungen sind Steuerungen mit einer eigenen Dynamik. Man kann sie deshalb in der Form

Ablaufsteuerung  $\Sigma_R$  :

$$\begin{cases} z_s(k+1) = G_s(z_s(k), w(k), s(k)), & z_s(0) = z_{s0} \\ v(k) = H_s(z_s(k), w(k), s(k)) \end{cases}$$

(13.5)

notieren, wobei  $z_s(k)$  ihren Zustand und  $s(k)$  das Steuerungsziel bezeichnet. Da sie ein deterministisches Verhalten aufweisen sollen, können Zustands- und Ausgabegleichung getrennt geschrieben werden, wie es bei deterministischen Automaten immer möglich ist.

Dass diskrete Regler häufig interne Zustände benötigen, wird durch das folgende Beispiel illustriert.

Beispiel 13.2 Steuerung eines Getränkeautomaten

Ein Getränkeautomat soll Tee, Kaffee oder Kakao zubereiten, nachdem eine Münze eingeworfen und das Getränk ausgewählt wurde. Bevor ein Getränk ausgewählt ist, soll es für den Kunden möglich sein, durch Betätigung eines Geldrückgabeknopfes die eingeworfene Münze zurückzufordern. Die Steuerung muss sich also „merken“, ob das Getränk bereits bezahlt ist und ob die Zubereitung des gewählten Getränks bereits begonnen hat. Dafür sind Speicher notwendig.

Zur Vereinfachung der Betrachtungen wird davon ausgegangen, dass die Überprüfung der Echtheit und des Wertes der eingeworfenen Münze durch einen geeigneten Mechanismus ohne Eingriff der hier betrachteten Steuerung erfolgt und dass die Zubereitung der Getränke durch Auswurf des Bechers, Einfüllen einer bestimmten Menge von Getränekpulver, Steuerung der Wasserzufuhr usw. von unterlagerten Steuerungen übernommen wird.

Der Getränkeautomat besteht aus den Komponenten, die das Geld kassieren und die Getränke herstellen („Getränkeautomat“) und der Ablaufsteuerung, die die Bezahl- und Herstellungsprozesse auslöst. Beide Teile agieren in der in Abb. 13.3 gezeigten Rückkopplungsstruktur.

Die Ablaufsteuerung erhält folgende Informationen:

$w$	Bedeutung
$w_1$	Münze ist eingeworfen.
$w_2$	Geldrückgabeknopf ist gedrückt.
$w_{31}$	Tee ist gewählt.
$w_{32}$	Kaffee ist gewählt.
$w_{33}$	Kakao ist gewählt.
$w_4$	Getränkezubereitung ist abgeschlossen.

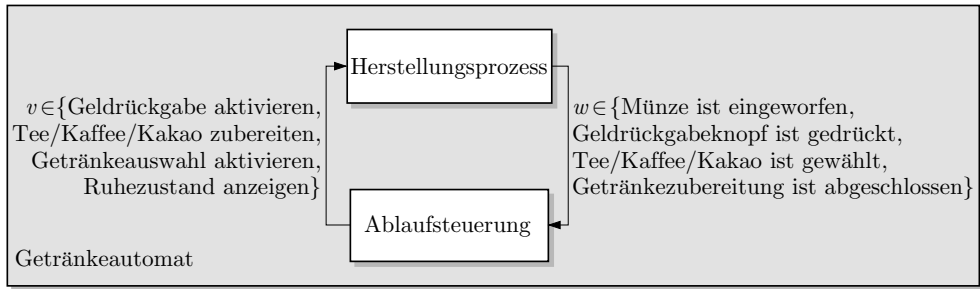


Abb. 13.3: Beispiel für eine Ablaufsteuerung

Sie löst daraufhin einen der folgenden Prozesse aus:

$v$	Bedeutung
$v_1$	Geldrückgabe aktivieren
$v_{21}$	Tee zubereiten
$v_{22}$	Kaffee zubereiten
$v_{23}$	Kakao zubereiten
$v_3$	Getränkeauswahl aktivieren
$v_4$	Ruhezustand anzeigen

Damit sich der Regler merkt, in welchem Schritt sich der Zubereitungsprozess befindet, werden drei Zustände eingeführt:

$z_s$	Bedeutung
1	Ruhezustand.
2	Getränk ist bezahlt.
3	Getränk wird zubereitet.

Die Ablaufsteuerung ist durch einen Automaten

$$\Sigma_R = (\mathcal{Z}, \mathcal{W}, \mathcal{V}, G_s, H_s, z_1)$$

mit den bereits definierten Zuständen, Eingaben und Ausgaben

$$\mathcal{Z} = \{1, 2, 3\}$$

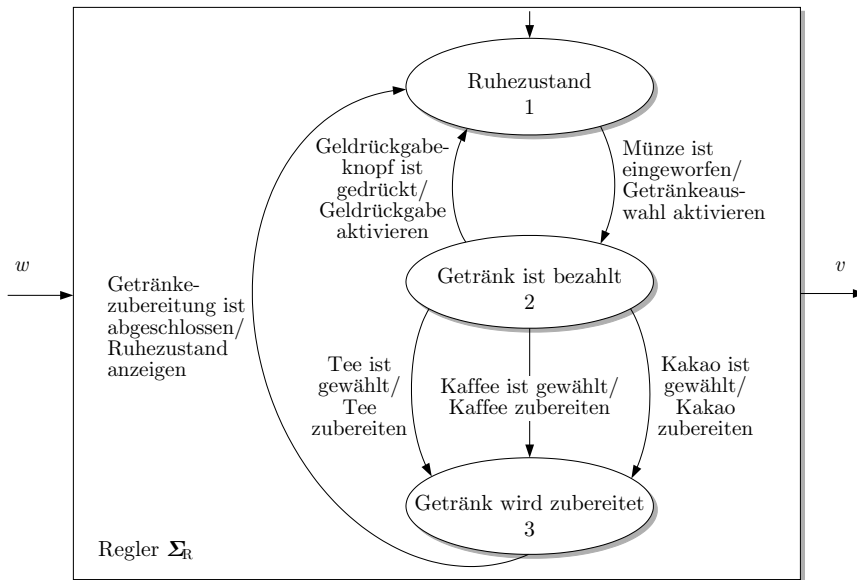
$$\mathcal{W} = \{w_1, w_2, w_{31}, w_{32}, w_{33}, w_4\}$$

$$\mathcal{V} = \{v_1, v_{21}, v_{22}, v_{23}, v_3, v_4\}$$

und folgenden Funktionen beschrieben:

Nachfolgezustand	Ausgabe	Zustand	Eingabe
$z_s(k+1)$	$v(k)$	$z_s(k)$	$w(k)$
2	$v_3$	1	$w_1$
1	$v_1$	2	$w_2$
3	$v_{21}$	2	$w_{31}$
3	$v_{22}$	2	$w_{32}$
3	$v_{23}$	2	$w_{33}$
1	$v_4$	3	$w_4$

$G_s, H_s =$



**Abb. 13.4:** Ablaufsteuerung  $\Sigma_R$  des Getränkeautomaten

Die Begriffe Eingabe und Ausgabe werden vom Standpunkt des Reglers verwendet. Für alle in der Tabelle nicht aufgeführten Paare  $(z_s(k), w(k))$  führt der Regler keine Zustandsänderung aus und erzeugt keine Ausgabe. Abbildung 13.4 zeigt den Automatengraphen des Reglers  $\Sigma_R$ .

**Realisierung der Steuerung.** Bei alten Getränkeautomaten ist die Steuerung durch eine Mechanik realisiert, die die notwendige Folge von Aktionen auslöst, indem sie den Münzeinwurf mechanisch kontrolliert und einen Hebel bewegt, der die Knöpfe für die Wahl des Getränks freigibt. In Abhängigkeit vom gewählten Getränk werden dann weitere Aktionen für die Getränkezubereitung eingeleitet. Wie man sich leicht vorstellen kann, ist dafür eine ausgeklügelte Mechanik erforderlich, weshalb die auf diese Weise gesteuerten Automaten meist nur wenige unterschiedliche Getränke herstellen. Moderne Geräte werden durch eine elektronisch realisierte Steuerung automatisiert, so dass sie ein größeres Getränkeangebot haben und weiterer Komfort wie z. B. Geldwechsel möglich ist.

Unabhängig von der Implementierung muss die Steuerung die durch den Automaten  $\Sigma_R$  beschriebene Funktion realisieren. Die Steuerung arbeitet im geschlossenen Wirkungskreis, denn sie reagiert auf Ausgaben  $w(k) \in \mathcal{W}$  der Regelstrecke mit Eingaben  $v(k) \in \mathcal{V}$ . Dabei besteht bei der hier verwendeten Betrachtungsweise die Strecke aus dem Getränkeautomaten und dem Nutzer, denn die Nutzereingaben wurden als Ausgaben  $w(k)$  der Strecke interpretiert. Man kann das Modell  $\Sigma_R$  der Ablaufsteuerung auch in die Form (13.5) bringen, bei der die Wahl des Getränks als Spezifikation  $s(k)$  des Steuerungsziels interpretiert wird.  $\square$

Wie das Beispiel zeigt, werden Steuerungen verwendet, um gewünschte Arbeitsabläufe zu realisieren, bei denen eine Reihe von Arbeitsschritten in einer auf das Steuerungsziel abgestimmten Reihenfolge auszuführen sind. Beginn und Ende der Arbeitsschritte werden durch Ereignisse, also durch Veränderungen von Signalwerten am Ausgang der Strecke, angezeigt. Die einzelnen Schritte werden durch die Strecke ausgeführt, während der Regler vorgibt, welcher Schritt

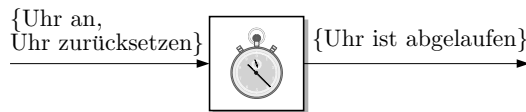
in einer gegebenen Situation als nächstes begonnen wird. Der nächste Schritt soll dem vorausgehenden immer dann folgen, wenn die Strecke eine Rückmeldung gibt und damit anzeigt, dass der vorherige Schritt erfolgreich abgeschlossen wurde. Diese Rückmeldung wird auch als *Weiterschaltbedingung* bezeichnet.

Wie beim Getränkeautomaten kann die durch den Regler erzeugte Aktionsfolge von Zielvorgaben abhängen, die einen von mehreren möglichen Zweigen im Steuerungsalgorithmus aktivieren. Um diese Funktion erfüllen zu können, muss der Regler speichern, in welchem aktuellen Zustand sich die Strecke befindet. Sein Zustand  $z_s$  entspricht deshalb häufig dem Zustand  $z$  der Strecke.

### 13.3.2 Zeitplansteuerungen

Bei den bisher betrachteten Steuerungen wurden die Aktivitäten der Steuereinrichtung durch Ereignisse ausgelöst, die der zu steuernde Prozess erzeugt. In zahlreichen Anwendungsfällen soll die Steuerung darüber hinaus in der Lage sein, einen bestimmten Zeitplan einzuhalten, wobei die Zeit entweder absolut oder in Relation zu Prozessereignissen gemessen wird.

Für diese Erweiterung muss die Steuereinrichtung eine oder mehrere Uhren besitzen, die nach dem Ablauf einer bestimmten Zeitspanne Ereignisse erzeugen. So braucht man beispielsweise eine Uhr, um einen geeigneten Abschaltzeitpunkt für eine nicht mehr benutzte Rolltreppe bestimmen zu können (Beispiel 13.3).



**Abb. 13.5:** Uhr als ereignisdiskretes System

Abbildung 13.5 zeigt, wie eine Uhr als diskretes dynamisches System behandelt wird. Die Dynamik der eingeschalteten Uhr ist kontinuierlich, denn sie kann durch das Zustandsraummodell

$$\text{Uhr : } \begin{cases} \dot{x}(t) = 1, & x(t_0) = 0 \\ y(t) = x(t) \end{cases}$$

beschrieben werden, dessen Ausgangsgröße zur Zeit  $t$  die Zeitdauer  $t - t_0$  anzeigt, die seit dem Einschalten der Uhr zur Zeit  $t_0$  vergangen ist:

$$y(t) = \int_{t_0}^t 1 \, d\tau = t - t_0.$$

Hier wird die Uhr jedoch mit diskreten Eingangs- und Ausgangsgrößen betrachtet. Die Eingangsgröße schaltet die Uhr ein bzw. setzt den Zustand  $x$  auf null zurück. Die diskrete Ausgabe „Uhr ist abgelaufen“ wird zu der Zeit  $t$  erzeugt, bei der die Uhr eine vorgegebene Zeit  $\bar{t}$  erreicht hat ( $y(t) = \bar{t}$ ). Dieselben Betrachtungen lassen sich natürlich für Uhren durchführen, die durch

die Eingabe auf die Zeit  $\bar{t}$  gesetzt werden und rückwärts laufen, um bei  $y(t) = 0$  zur Zeit  $t$  das Ereignis „Uhr ist abgelaufen“ zu generieren.

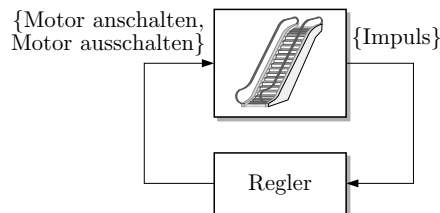
**Zeitgeführte Ablaufsteuerung.** Wenn die von der Steuerung erzeugten Stelleingriffe  $v(k)$  ausschließlich von der Zeit abhängen, die seit der letzten Eingabe vergangen ist, spricht man von einer zeitgeführten Ablaufsteuerung. Dies ist eine Steuerung in der offenen Wirkungskette, denn der aktuelle Zustand der Strecke hat keinen Einfluss auf den nächsten Steuereingriff (vgl. Beispiel 2.11 auf S. 75).

Ein bekanntes Beispiel für eine zeitgeführte Ablaufsteuerung ist eine Ampelsteuerung, bei der die Umschaltung der Ampel nur von der seit der letzten Umschaltung vergangenen Zeit abhängig ist. Welchen Nachteil eine solche Steuerung in der offenen Wirkungskette hat, weiß jeder Verkehrsteilnehmer. Die Ampel kann nicht auf das aktuelle Verkehrsaufkommen reagieren und schaltet eine Fahrtrichtung auch dann auf grün, wenn sie zum gegenwärtigen Zeitpunkt niemand benutzen will.

**Steuerungsentwurf.** Um Ablaufsteuerungen, die im geschlossenen Wirkungskreis arbeiten, und Zeitplansteuerungen, die zusammen mit ihren internen Uhren in einer offenen Wirkungskette die Steuereingaben erzeugen, beim Entwurf gleichartig behandeln zu können, werden Uhren während des Entwurfs als Teil des zu steuernden Prozesses aufgefasst, so dass die Signale „Uhr anschalten“ oder „Uhr zurücksetzen“ Ausgaben der Steuerung an die um die Uhr erweiterte Strecke sind. Das Ereignis „Uhr ist abgelaufen“ wirkt wie eine Ausgabe des Prozesses und ist folglich eine Eingabe der Steuerung. Bei der Realisierung der Steuerung muss die Uhr natürlich als Bestandteil der Steuereinrichtung durch analoge Uhren oder Programmteile realisiert werden. Das folgende Beispiel zeigt den dargestellten Entwurfsweg.

### Beispiel 13.3 Regelung einer Rolltreppe

Eine Regelung soll eine Rolltreppe anschalten, wenn ein Kontakt anzeigt, dass jemand die Treppe benutzen will. Wenn der Kontakt über eine bestimmte Zeitspanne  $\bar{t}$  hinweg nicht erneut betätigt wurde, soll der Regler die Treppe wieder abschalten. Die Zeitspanne ist so bemessen, dass der letzte Benutzer am Ende der Rolltreppe angekommen ist, bevor die Treppe abgeschaltet wird.



**Abb. 13.6:** Blockschaltbild der geregelten Rolltreppe

Die geregelte Rolltreppe hat die in Abb. 13.6 gezeigte Struktur. Wenn sich ein Benutzer der Rolltreppe nähert, erzeugt eine Lichtschranke einen Impuls, der die einzige Ausgabe der Treppe für die hier betrachtete Steuerung ist:

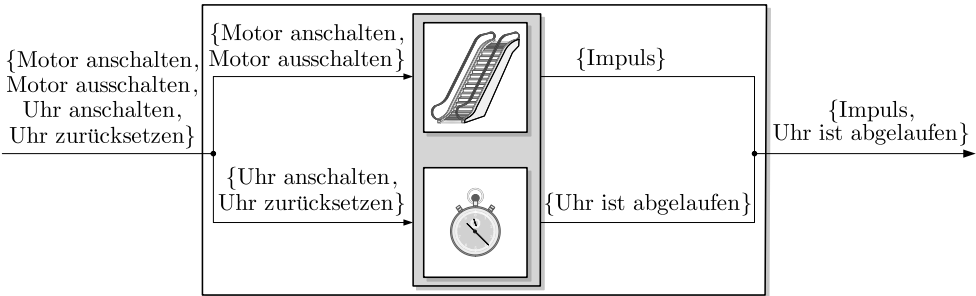


$w_1$  = „Impuls“: Ein Benutzer betritt die Treppe.

Der Regler schaltet den Rolltreppenmotor ein bzw. aus

- $v_{11}$  = Motor anschalten
- $v_{12}$  = Motor ausschalten,

so dass sich die Rolltreppe in einem der beiden Zustände „Treppe ist ausgeschaltet“ oder „Treppe ist angeschaltet“ befindet.



**Abb. 13.7:** Erweiterte Steuerstrecke, die aus der Rolltreppe und einer Uhr besteht

Um die gestellte Regelungsaufgabe zu lösen, muss die Rolltreppensteuerung eine Uhr besitzen, die bei jeder Betätigung des Kontakts gestartet bzw. auf null zurückgesetzt wird. Für die Festlegung des Steuerungsalgorithmus werden Rolltreppe und Uhr gemeinsam als erweiterter Prozess aufgefasst (Abb. 13.7), dessen vier Eingaben

$v$	Bedeutung
$v_{11}$	Motor anschalten
$v_{12}$	Motor ausschalten
$v_{21}$	Uhr anschalten
$v_{22}$	Uhr zurücksetzen

vom Regler in den Kombinationen

$$\begin{aligned} v_1 &= \begin{pmatrix} \text{Motor anschalten} \\ \text{Uhr anschalten} \end{pmatrix} = \begin{pmatrix} v_{11} \\ v_{21} \end{pmatrix} \\ v_2 &= \begin{pmatrix} \varepsilon \\ \text{Uhr zurücksetzen} \end{pmatrix} = \begin{pmatrix} \varepsilon \\ v_{22} \end{pmatrix} \\ v_3 &= \begin{pmatrix} \text{Motor ausschalten} \\ \varepsilon \end{pmatrix} = \begin{pmatrix} v_{12} \\ \varepsilon \end{pmatrix} \end{aligned}$$

verwendet werden. Die Ausgaben von Rolltreppe und Uhr

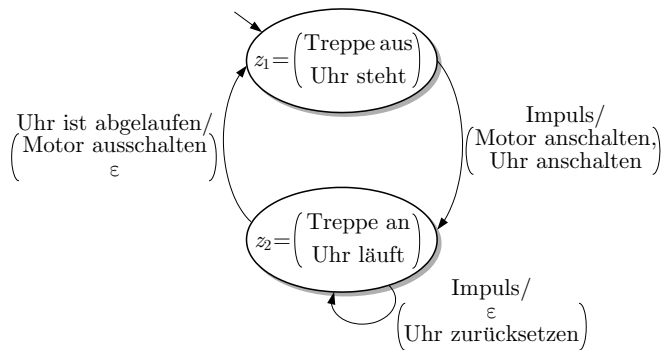
$w$	Bedeutung
$w_1$	Impuls (Ein Benutzer betritt die Treppe.)
$w_2$	Uhr ist abgelaufen.

sind unabhängig voneinander und es wird vereinfachend angenommen, dass sie nie zum selben Zeitpunkt eintreten. Die Vorstellung, dass die Uhr ein Bestandteil der Regelstrecke sei, hat den Vorteil, dass das Reglergesetz durch ein logisches (nicht zeitbewertetes) Modell dargestellt werden kann, weil alle maßgebenden Zeitverzögerungen in der erweiterten Regelstrecke liegen.

Die erweiterte Regelstrecke (Abb. 13.7) kann sich in vier möglichen Zuständen befinden, von denen nur die beiden Zustände

$$z_1 = \begin{pmatrix} \text{Treppe aus} \\ \text{Uhr steht} \end{pmatrix} \quad \text{und} \quad z_2 = \begin{pmatrix} \text{Treppe an} \\ \text{Uhr läuft} \end{pmatrix}$$

von Bedeutung sind.



**Abb. 13.8:** Automat, der den Regler  $\Sigma_R$  für die Rolltreppe beschreibt

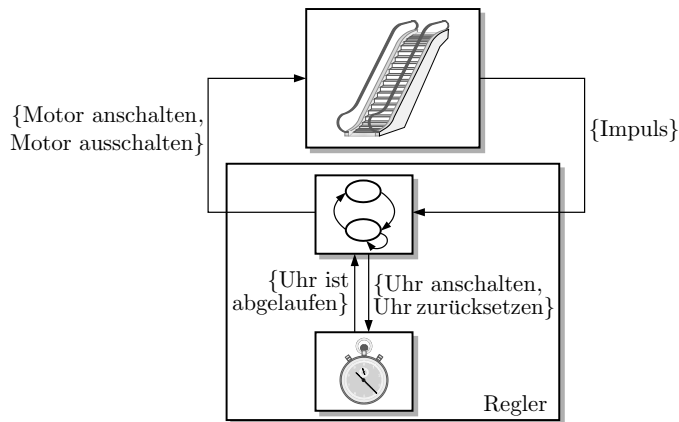
Der Regler  $\Sigma_R$  für die Rolltreppe ist in Abb. 13.8 als Automatengraph zu sehen. Er beschreibt das Reglergesetz (13.5), dessen Funktionen  $G_s$  und  $H_s$  äquivalent durch die folgenden Tabellen wiedergegeben werden:

$G_s =$	$z_s(k+1)$	$z_s(k)$	$w(k)$
	$z_2$	$z_1$	$w_1$
	$z_2$	$z_2$	$w_1$
	$z_1$	$z_2$	$w_2$

$H_s =$	$v(k)$	$z_s(k)$	$w(k)$
	$v_1$	$z_1$	$w_1$
	$v_2$	$z_2$	$w_1$
	$v_3$	$z_2$	$w_2$

Um die Regelung anzuwenden, muss die Uhr als Komponente des Reglers realisiert werden. Das durch den Automaten repräsentierte Reglergesetz wirkt deshalb einerseits auf die interne Uhr und andererseits auf den Rolltreppenmotor (Abb. 13.9).

**Diskussion.** Dieses Beispiel – wie auch das Beispiel der Rolltreppensteuerung – zeigt, dass das Reglergesetz dadurch aufgestellt wird, dass man in einem Modell beschreibt, wie sich der Regelkreis verhalten soll und dass man dann dieses Modell als Reglergesetz interpretiert. Wenn man in Abb. 13.8 die Kantenbeschriftungen streicht, beschreibt das Modell die zwei Zustände, in denen sich die geregelte Rolltreppe befinden kann. Das Modell stellt dann die Rolltreppe einschließlich ihrer Steuerung dar, also so, wie man das Verhalten der Rolltreppe als externer Beobachter sieht. Um dieses Modell als Reglergesetz einzusetzen, muss man an die Kanten schreiben, welche Ausgabe die Regelstrecke (Rolltreppe, Uhr) erzeugt und mit welcher Eingabe der Regler reagiert, wenn die betreffenden Zustandsübergänge auftreten. Dadurch entsteht aus dem Modell des Regelkreises ein Modell, das als Reglergesetz verwendet werden kann.  $\square$



**Abb. 13.9:** Blockschaltbild mit Kennzeichnung des strukturellen Aufbaus des Reglers

**Steuerungen mit Zählern.** Bei einer Reihe von Anwendungen soll die Steuerung dafür sorgen, dass bestimmte Prozessschritte wiederholt werden. So müssen Reinigungsvorgänge mit einer bestimmten Häufigkeit ausgeführt werden (z. B. in einer Waschmaschine oder einem Geschirrspüler). Die Steuerung muss sich dann merken, wie oft der betreffende Prozessschritt bereits ausgeführt wurde. Dafür werden Zähler verwendet, deren interner Zustand die Zahl der bereits durchgeführten Arbeitszyklen beschreibt. Diese Zähler können ähnlich wie Uhren behandelt werden.

#### Aufgabe 13.1 *Komfortverbesserung in Kraftfahrzeugen*

Moderne Personenkraftwagen besitzen Steuerungen, mit denen die folgenden Spezifikationen erfüllt werden:

- Wenn der Rückwärtsgang eingelegt wird, wird der rechte Spiegel nach unten gedreht, damit der Fahrer die Bordsteinkante besser sehen kann. Außerdem wird bei Regenwetter der Scheibenwischer an der Heckscheibe aktiviert, allerdings nur, wenn die Außentemperatur über  $4^{\circ}\text{C}$  liegt. Weiterhin wird die Lautstärke der Stereoanlage gemindert. Diese Vorgänge werden rückgängig gemacht, wenn der Rückwärtsgang wieder herausgenommen wird.
- Seitlich angebrachte Zusatzscheinwerfer sollen Kurven ausleuchten. Sie werden eingeschaltet, wenn die Geschwindigkeit des Fahrzeugs unter  $40 \frac{\text{km}}{\text{h}}$  liegt und wenn entweder der rechte bzw. linke Blinker benutzt oder das Lenkrad um mehr als  $10^{\circ}$  verdreht wurde. Wenn die angegebenen Bedingungen nicht mehr erfüllt sind, werden die Scheinwerfer nach Ablauf einer Zeitdauer  $\bar{T}$  abgeschaltet.

Braucht man für die Erfüllung dieser Spezifikationen Verknüpfungssteuerungen oder Ablaufsteuerungen? Beschreiben Sie die Steuerungen in der Form (13.3) bzw. (13.5). □

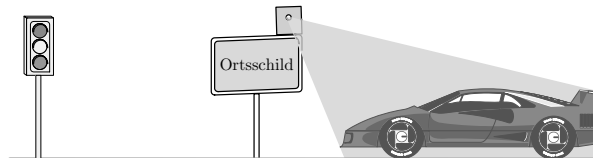
**Aufgabe 13.2\*** Steuerung einer Treppenhausbeleuchtung

Das Licht im Treppenhaus von Mehrfamilienhäusern wird häufig dadurch ein- und ausgeschaltet, dass man beim Betreten des dunklen Treppenhauses durch einen Taster das Licht einschaltet und die Steuerung das Licht nach Ablauf einer bestimmten Zeit ausschaltet.

1. Was sind der Eingang bzw. der Ausgang der Treppenhausbeleuchtung als der zu steuernde Prozess, was beschreibt den Zustand?
2. Stellen Sie den Steuerungsalgorithmus durch einen Automatengraphen und eine Automatentabelle dar.
3. Die bisher betrachtete Steuerung der Treppenhausbeleuchtung hat den Nachteil, dass die in der Steuerung enthaltene Uhr nicht zurückgesetzt werden kann, wenn man den Lichttaster bei eingeschalteter Beleuchtung bedient. Deshalb kann es passieren, dass sich die Beleuchtung auf „halber Treppe“ abschaltet und Personen im dunklen Treppenhaus stehen. Modifizieren Sie die Steuerung so, dass man diese Situation verhindern kann, indem man beim Betreten des Treppenhauses den Lichttaster selbst dann betätigt, wenn die Beleuchtung bereits eingeschaltet ist. ☐

**Aufgabe 13.3\*** Steuerung einer „Bremsampel“

In Spanien und Italien versucht man in kleinen Ortschaften mit Hilfe von Ampeln, die Geschwindigkeit des Durchgangsverkehrs auf die erlaubte Höchstgeschwindigkeit von  $50 \frac{\text{km}}{\text{h}}$  zu begrenzen. Ein Sensor am Ortsschild erkennt, wann ein Fahrzeug kommt. Die etwa 100 Meter davon entfernte Ampel steht auf rot und wird erst nach Ablauf derjenigen Zeit auf gelb und dann auf grün umgestellt, die ein mit der erlaubten Höchstgeschwindigkeit fahrendes Fahrzeug vom Ortsschild bis zur Ampel benötigt (bei 100 m also nach etwa 7 Sekunden). Zu schnell fahrende Fahrzeuge müssen vor der roten Ampel halten.



**Abb. 13.10:** Technische Realisierung einer „Bremsampel“

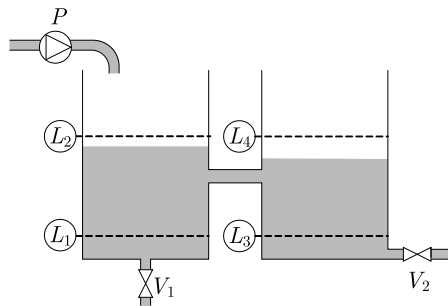
1. Wie sieht das Blockschaltbild des Regelkreises aus? Aus welchen Komponenten besteht der zu steuernde Prozess?
2. Betrachten Sie zunächst nur ein einzelnes Fahrzeug und beschreiben Sie durch einen Automaten, wie sich das gesteuerte System verhalten soll. Beachten Sie dabei, dass beim Rückschalten der Ampel auf rot die Ampel zunächst auf gelb stehen muss.
3. Inwiefern kann man das Modell des Regelkreises als Steuerungsalgorithmus verwenden?
4. Erweitern Sie Ihr Modell so, dass die Ampel nur dann auf rot zurückgeschaltet wird, wenn man auf Grund des Abstandes zu den nachfolgenden Fahrzeugen sicher sein kann, dass ordnungsgemäß fahrende Fahrzeuge nicht versehentlich durch die Ampel gestoppt werden. ☐

**Aufgabe 13.4\*** Steuerung eines Batchprozesses

Mit dem in Abb. 13.11 gezeigten Behältersystem soll durch eine geeignete Steuerung der folgende zyklische Prozess realisiert werden:

1. Der Prozess beginnt mit leeren Behältern. Beide Behälter werden mit Hilfe der Pumpe  $P$  bei geschlossenen Ventilen  $V_1$  und  $V_2$  gefüllt, bis die Füllstände  $h_1$  und  $h_2$  oberhalb der Positionen  $l_2$  bzw.  $l_4$  der Sensoren  $L_2$  und  $L_4$  liegen.
2. Es wird das Ventil  $V_2$  geöffnet, bis der Füllstand des rechten Behälters unterhalb des Sensors  $L_3$  liegt. Dann wird das Ventil  $V_2$  wieder geschlossen.
3. Der linke Behälter wird durch das Ventil  $V_1$  entleert. Dann wiederholt sich der angegebene Prozess.

Die Füllstandssensoren  $L_i$  arbeiten diskret, wobei sie den Wert eins ausgeben, wenn der Füllstand im Behälter über der Sensorposition liegt. Die Ventile und die Pumpe können nur vollständig geöffnet und geschlossen bzw. ein- oder ausgeschaltet werden.



**Abb. 13.11:** Behältersystem mit Füllstandssensoren

Lösen Sie hierzu die folgenden Aufgaben:

1. Zeichnen Sie das Blockschaltbild des Regelkreises.
2. Welche diskreten Prozesszustände muss der Regler erkennen, um die Spezifikationen erfüllen zu können? Definieren Sie diese Prozesszustände unter Angabe der Sensorwerte, die in diesen Zuständen auftreten, und der entsprechenden Intervalle für die Füllstände in beiden Behältern.
3. Muss die Aufgabe mit einer Verknüpfungssteuerung oder einer Ablaufsteuerung gelöst werden? Beschreiben Sie das Steuerungsgesetz.
4. Zeichnen Sie qualitativ das zeitliche Verhalten der beiden Behälter sowie der Steuereinrichtung, wobei Sie für die Behälter sowohl das kontinuierliche Verhalten  $h_1(t)$ ,  $h_2(t)$  der Füllstände als auch das ereignisdiskrete Verhalten  $z(t)$  der Sensoren angeben.  $\square$

**Aufgabe 13.5** Steuerung der vollautomatischen U-Bahn in Lille

In Lille fährt die U-Bahn seit vielen Jahren automatisch ohne Fahrzeugführer. Die Bahnsteige der Haltestellen sind durch Glastüren versperrt. Die Fahrzeuge der U-Bahn halten so, dass die Wagentüren und die Glastüren der Bahnsteige genau hintereinander stehen und sich gleichzeitig öffnen und schließen können. In den Wagen sind Geschwindigkeitsregelungen installiert, die die Wagen auf eine vorgeschriebene Geschwindigkeit beschleunigen, mit konstanter Geschwindigkeit führen bzw. mit einer

vorgegebenen Bremsbeschleunigung zum Stillstand bringen. Auf der hier betrachteten Steuerungsebene dienen die Kommandos „beschleunigen“, „mit konstanter Geschwindigkeit fahren“, „abbremsen“ als diskrete Eingangsgrößen für diese Regelungen und das Fahrzeug gibt die diskreten Ausgangsgrößen „Sollgeschwindigkeit erreicht“ und „Fahrzeug steht“ aus.

Im Folgenden wird die Steuerung betrachtet, die dafür sorgt, dass die Wagen nach dem Schließen der Türen beschleunigt werden, bis zur nächsten Station mit konstanter Geschwindigkeit fahren, rechtzeitig abgebremst werden und sich die Türen der Wagen und der Bahnsteige öffnen und schließen.

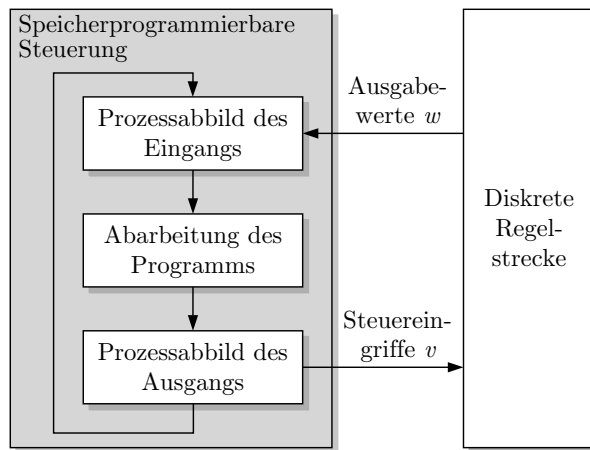
1. Zeichnen Sie ein Blockschaltbild des gesteuerten U-Bahn-Wagens. Welche Elemente der U-Bahn gehören zur Steuerstrecke?
2. Welche Informationen benötigen Sie zur Realisierung der Steuerung. Tragen Sie diese Informationen als Ausgaben und Eingaben der Steuerstrecke in das Blockschaltbild ein. Wo müssen Mess- und Stellglieder im U-Bahn-Wagen, an der Fahrstrecke bzw. am Bahnsteig angebracht werden, damit die Steuereinrichtung und die Steuerstrecke in der gewünschten Weise miteinander kommunizieren können?
3. Beschreiben Sie durch einen Automaten, wie das Steuerungsgesetz aussehen muss, damit ein einzelner Wagen automatisch fahren kann. Handelt es sich dabei um eine Steuerung in der offenen Wirkungskette oder eine Steuerung im geschlossenen Wirkungskreis (Regelung)?
4. Erweitern Sie die Steuerung auf mehrere gleichzeitig fahrende Züge, wobei verhindert werden muss, dass die Züge zu dicht aufeinander auffahren. Sind dafür mehr Informationen als beim Betrieb eines einzelnen Wagens notwendig?
5. Im U-Bahn-Verkehr wird wie im Eisenbahnverkehr das Streckennetz in „Blöcke“ unterteilt, also in Streckenabschnitte, die durch Ampeln freigegeben oder blockiert werden. Dabei wird die Ampel eines vor einem Block mit roter Ampel liegenden Blocks auf gelb geschaltet, um anzuzeigen, dass der nachfolgende Block gesperrt ist. Wie muss die Steuerung des U-Bahn-Wagens erweitert werden, um auf derartige Signale selbstständig reagieren zu können? Wie müssen die Ampeln an den Blöcken gesteuert werden? □

## 13.4 Speicherprogrammierbare Steuerungen

Für die Realisierung diskreter Steuerungen müssen die Funktion  $H_s$  der Verknüpfungssteuerung (13.3) bzw. die Funktionen  $G_s$  und  $H_s$  der Ablaufsteuerung (13.5) technisch umgesetzt werden. Als erster Schritt müssen dafür die zwischen der Steuereinrichtung und dem zu steuernden Prozess ausgetauschten Signale kodiert werden, wobei häufig eine binäre Kodierung verwendet wird. Die weitere Verarbeitung der Signale hängt von der verwendeten Hardware ab.

Früher entstand das Steuerungsprogramm durch eine Verdrahtung von Schaltern, die durch Relais oder Schütze realisiert wurden. Man spricht bei dieser Realisierung von einer *verbindungsprogrammierten Steuerung*. So wurde die einfache Steuerung (13.4) dadurch realisiert, dass die Drähte mit den binären Signalen  $w_1$  und  $w_2$  an zwei in Reihe geschaltete Relais geführt wurden, die nur dann einen Stromfluss zulassen ( $v = 1$ ), wenn beide Signale den Wert 1 haben und folglich beide Relais geschlossen sind. Bei derartigen Steuerungen war eine Änderung des Steuerungsgesetzes nur durch Änderung der Hardware möglich. Die Steuerung war durch die Verdrahtung von Schaltern zwar sehr sicher, aber unflexibel und in der Realisierung aufwändig.

Deshalb werden heute diskrete Steuerungen durch Programme beschrieben, mit denen ein Rechner die Ausgaben der Steuerstrecke verarbeitet und die dabei erhaltenen Stellgrößen an die Strecke überträgt. Derartige Steuerungen heißen *speicherprogrammierbare Steuerungen* (SPS). Sie können vielfältigere und kompliziertere Steuerungsalgorithmen realisieren als verbindungsprogrammierte Steuerungen. Änderungen des Steuerungsalgorithmus sind durch Programmänderungen und ohne Veränderung der Hardware möglich, zumindest solange keine neuen Mess- oder Stellglieder genutzt werden sollen. Außerdem können moderne speicherprogrammierbare Steuerungen für die Realisierung kontinuierlicher Regelungen eingesetzt werden.



**Abb. 13.12:** Struktur einer speicherprogrammierbaren Steuerung

Abbildung 13.12 zeigt den Aufbau einer SPS. Die Ausgabewerte  $w$  der Steuerstrecke werden in einen Speicherbereich der SPS geschrieben, der Prozessabbild des Eingangs heißt. Auf diese Werte greift das Programm zu, das den Steuerungsalgorithmus realisiert. Die Ergebnisse werden in einen anderen Speicherbereich der SPS geschrieben, der Prozessabbild des Ausgangs genannt wird. Aus diesem Bereich werden die Steuereingriffe  $v$  ausgelesen und an den zu steuernden Prozess übermittelt.

Die SPS bearbeitet die Schritte

1. Einlesen der Ausgabewerte
2. Abarbeiten des Programms
3. Auslesen der Steuereingriffe

zyklisch. Eine Veränderung der Ausgabewerte am Prozess wird vom Steuerungsalgorithmus nur zu Beginn jedes Zyklus wahrgenommen und der Steuereingriff nur nach Abarbeitung des Programms verändert. Die SPS „wartet“ also nicht darauf, dass das nächste Ereignis am Prozessausgang sichtbar wird, wie es bei der Modellbildung der Strecke und der Steuerung stets angenommen wird. Ist die Zykluszeit, die typischerweise im Millisekundenbereich liegt, im Vergleich zum zeitlichen Verhalten der Strecke klein, wirkt sich dieser Unterschied nicht auf die Güte der Steuerung aus. Die Steuerung arbeitet quasikontinuierlich.

Speicherprogrammierbare Steuerungen verfügen heute über umfangreiche Hilfen für die Programmierung (Entwicklungsumgebungen), so dass die Programme weitgehend durch grafisches Programmieren erstellt werden können, wobei die einzelnen Prozessschritte bzw. Modelle der Steuerung als Blöcke dargestellt sind und mit einem grafischen Editor verbunden werden. Die am weitesten verbreiteten Darstellungsformen von Steuerungsalgorithmen sind *Anweisungslisten*, *Funktionspläne* und *Kontaktpläne*. Sie enthalten Blöcke, mit denen die gemessenen Signale zu den Stellgrößen verknüpft werden. Wenn die Messgrößen binär sind oder binär kodiert wurden, können die Verarbeitungsschritte im Wesentlichen auf Operationen der booleschen Algebra zurückgeführt werden. Integrierte Prozessdatenbanken erleichtern einen Zugriff auf alle über den Prozess verfügbaren Informationen. International wird der Programmierstandard IEC 61131-3 eingeführt, um speicherprogrammierbare Steuerungen unterschiedlicher Hersteller kompatibel zu machen.

## Literaturhinweise

Eine Einführung in die Norm IEC 61131-3 wird in [104] gegeben, wo auch ausführlich auf den Aufbau und die Programmierung von speicherprogrammierbaren Steuerungen eingegangen wird. Als Beispiel für eine Programmieranleitung wird hier das Buch [10] angeführt, das die Sprache STEP 7 beschreibt.



# 14

## Entwurf diskreter Steuerungen

*Nach einer Erläuterung des allgemeinen Entwurfsalgorithmus für diskrete Steuerungen werden Entwurfsverfahren für Automaten und Petrinetze beschrieben und an Beispielen demonstriert.*

### 14.1 Entwurfsschritte

Im Mittelpunkt dieses Kapitels stehen systematische Entwurfsverfahren, bei denen entsprechend dem im Abschn. 14.1.1 angegebenen Entwurfsalgorithmus aus einem diskreten Modell des zu steuernden Systems und einer formalen Spezifikation des Steuerungsziels das Steuerungsgesetz abgeleitet wird. Derartige Verfahren werden im Abschn. 14.2 für Automaten und im Abschn. 14.3 für Petrinetze behandelt. Dabei wird gezeigt, dass man bei diesen Verfahren garantieren kann, dass das erhaltene Steuerungs- bzw. Reglergesetz die Spezifikationen erfüllt. Allerdings setzt dieses Entwurfsvorgehen eine umfangreiche Modellierung voraus, weshalb es nur für „kleine“ Aufgaben eingesetzt werden kann.

Die Komplexität des systematischen Entwurfs verbietet heute noch den breiten Einsatz der systematischen Entwurfsverfahren, weshalb in der Praxis der im Abschn. 14.1.2 erläuterte heuristische Steuerungsentwurf dominiert. Bei den dabei erhaltenen Steuerungsgesetzen ist nicht garantiert, dass die Spezifikation eingehalten wird, was man deshalb durch eine Verifikation überprüfen muss. Das grundlegende Vorgehen der Verifikation ist im Abschn. 14.4 dargestellt.

### 14.1.1 Systematischer Steuerungsentwurf

Im Folgenden wird davon ausgegangen, dass die Struktur des Regelkreises vorgegeben ist. Es ist also bekannt, welche physikalischen Größen die Signale  $v(k)$  und  $w(k)$  darstellen und welche Werte diese Größen annehmen können. Um die im Abschn. 13.1 gestellte Entwurfsaufgabe zu lösen, geht man in folgenden Schritten vor, die man in der Literatur auch als Steuerungsentwurf oder Steuerungssynthese bezeichnet. Da im entstehenden Steuerungsgesetz die Eingabe  $v(k)$  für die Strecke i. Allg. in Abhängigkeit von der aktuellen Ausgabe  $w(k)$  gebildet wird, ist das Ergebnis des folgenden Vorgehens eine rückgeführte Steuerung, so dass man auch in diesem Kapitel den Begriff „Steuerung“ meist durch den Begriff „Regelung“ ersetzen kann.

---

#### Entwurfsverfahren 14.1 Steuerungsentwurf für diskrete Systeme

---

**Gegeben:** Steuerstrecke  $\Sigma_S$   
Spezifikation des Verhaltens des gesteuerten Systems  $\bar{\Sigma}$

1. Stellen Sie das Modell der Steuerstrecke auf.
2. Legen Sie das Steuerungsgesetz  $\Sigma_R$  fest.
3. Analysieren Sie das Verhalten des gesteuerten Systems  $\bar{\Sigma}$ .
4. Bewerten Sie die Güte des gesteuerten Systems: Sind die Spezifikationen erfüllt, so ist das Entwurfsproblem gelöst. Andernfalls müssen Sie den Steuerungsentwurf ab Schritt 2 wiederholen.

**Ergebnis:** Steuerungsgesetz  $\Sigma_R$ .

---

Das im ersten Schritt aufgestellte Modell der Steuerstrecke muss alle möglichen Kombinationen von Zuständen und Eingaben berücksichtigen, denn das Modell soll das Verhalten des zu steuernden Prozesses für alle technisch realisierbaren Stelleingriffe darstellen. Welche Eingabe in welchem Streckenzustand ausgewählt werden muss, um das Steuerungsziel zu erreichen, wird im zweiten Schritt entschieden. Methoden dafür werden in den Abschnitten 14.2 und 14.3 erläutert. Für die anschließende Analyse werden die Modelle der Steuerstrecke und der Steuerung zu einem Modell des gesteuerten Systems zusammengefasst. Die Analyse soll zeigen, ob die Spezifikationen erfüllt sind. Ist dies der Fall, kann das Steuerungsgesetz angewendet werden. Andernfalls muss der im zweiten Schritt ausgewählte Steuerungsalgorithmus modifiziert werden.

Dieses Entwurfsvorgehen ist dem im Abschn. 8.1 für kontinuierliche Systeme beschriebenen sehr ähnlich. Bezüglich der Anwendung auf praktische Aufgabenstellungen unterscheiden sich beide Entwurfsverfahren jedoch wesentlich. Während für kontinuierliche Regelstrecken eine Vielzahl von Reglerentwurfverfahren entwickelt worden sind und rechnergestützte Werkzeuge ein schnelles Durchlaufen des Entwurfszyklus ermöglichen, gibt es für den Schritt 2 des o. a. Entwurfsalgorithmus für diskrete Systeme erst wenige ausgearbeitete Entwurfswege. Die in den nachfolgenden Abschnitten behandelten Entwurfsverfahren sind auf bestimmte Modellformen und Entwurfsforderungen zugeschnitten.

Der zweite Unterschied zwischen dem Reglerentwurf für kontinuierliche und diskrete Systeme resultiert aus der Komplexität diskreter Modelle und ist für die praktische Entwurfsdurchführung noch wichtiger. Bei einer kontinuierlichen Regelung kann man sich i. Allg. auf wenige Stell- und Regelgrößen konzentrieren, wobei es häufig sogar ausreicht, die Regelstrecke als ein System mit einer Eingangsgröße  $u(t)$  und einer Ausgangsgröße  $y(t)$  zu betrachten, wie es in diesem Buch praktiziert wird. Der Zusammenhang zwischen diesen Größen wird durch ein Zustandsraummodell (3.26) beschrieben, wobei die in diesem Modell vorkommenden Funktionen  $g$  und  $h$  meist als analytische Ausdrücke notiert werden können. Der Reglerentwurf stützt sich also auf ein kompaktes, gut handhabbares Streckenmodell.

Im Vergleich dazu ist die Modellbildung diskreter Systeme wesentlich aufwändiger und führt auf wesentlich komplexere Modelle. Es wurde im Abschn. 11.1.2 bereits darauf hingewiesen – und die behandelten Beispiele haben es deutlich gemacht – dass man bei der Aufstellung des Modells diskreter Systeme i. Allg. keine analytischen Ausdrücke für die Zustandsübergangsfunktion  $G$  und die Ausgabefunktion  $H$  aufschreiben kann, sondern diese Funktionen in Tabellenform o. ä. notieren muss. Für praktisch relevante Systeme sind diese Tabellen sehr groß. Außerdem ist die Anzahl der zu berücksichtigenden Eingangsgrößen und Ausgangsgrößen in den meisten Anwendungsfällen sehr groß. Obwohl jede dieser Größen nur endlich viele diskrete Signalwerte annehmen kann, steigen die Kombinationsmöglichkeiten dieser Werte exponentiell mit der Anzahl der Signale und der Anzahl der möglichen Signalwerte. Der Aufwand für die Modellbildung diskreter Systeme ist also ungleich höher als bei kontinuierlichen Systemen.

Die Komplexität der als ersten Schritt einer Steuerungssynthese zu lösenden Modellbildungsaufgabe soll folgende Überschlagsrechnung verdeutlichen. Bei der Modellbildung muss die Zustandsübergangsfunktion  $G : \mathcal{Z} \times \mathcal{V} \rightarrow \mathcal{Z}$  für alle  $N \cdot M$  Elemente des kartesischen Produkts  $\mathcal{Z} \times \mathcal{V}$  (mit großen Zahlen  $N = |\mathcal{Z}|$  und  $M = |\mathcal{V}|$ ) festgelegt werden. Wenn man eine Strecke mit je 100 binären Mess- und Stellgrößen hat und sich der Prozess in 1000 diskreten Zuständen befinden kann, so muss das Modell aus der Menge der  $1000 \cdot 1000 = 1\,000\,000$  möglichen Zustandsübergänge diejenigen repräsentieren, die bei den  $2 \cdot 100 = 200$  möglichen Steuereingriffen auftreten können. Wenn nur 1% der Zustandsübergänge auf Grund der physikalischen Wirkprinzipien des zu steuernden Prozesses möglich sind, so muss das Modell immer noch 10000 Zustandsübergänge beschreiben. Nicht nur die Komplexität eines derartigen Modells, sondern auch der Aufwand für seine Aufstellung ist so groß, dass die Anwendung von Syntheseverfahren auf sehr kleine Steuerungsprobleme beschränkt ist.

### 14.1.2 Heuristische Festlegung des Steuerungsgesetzes

Anstatt das Verhalten der Strecke in allen Zuständen für alle möglichen Eingaben zu beschreiben, stellt man beim heuristischen Steuerungsentwurf ein Modell auf, das das Verhalten der Strecke nur unter den für das Erreichen des Steuerungsziels zweckmäßigen Eingaben erfasst. Der Vorteil dieses Weges ist offensichtlich. Da man das Steuerungsziel bei der Modellbildung stets vor Augen hat, beschreibt man den zu steuernden Prozess nicht für alle möglichen Eingaben, sondern nur für diejenigen, die von der Steuereinrichtung in den betrachteten Streckenzuständen ausgewählt werden. Damit reduziert sich die Anzahl der zu betrachtenden Argumente der Zustandsübergangsfunktion  $G$  im Extremfall auf  $N$ , wenn jedem Zustand genau eine Eingabe zugeordnet ist. Die Funktion  $G$  ist dann nicht mehr für ihren gesamten Definitionsbereich

festgelegt, sondern nur für diejenigen Elemente, die im geschlossenen Kreis tatsächlich auftreten können. Wenn man ein Modell für das gesteuerte System durch die Kombination des nur teilweise definierten Streckenmodells mit dem Steuerungsgesetz aufstellt und das Verhalten des Gesamtsystems simuliert, treten für die Strecke gar keine anderen Eingaben als die bei der Modellbildung berücksichtigten auf und das Modell ist für die Analyse des Gesamtsystems ausreichend.

#### Beispiel 14.1 Rezeptsteuerung

Die beschriebene Reduktion der Modellbildungsaufgabe wird für den in Abb. 12.10 auf S. 452 links gezeigten Reaktor erläutert, der in einem Batchprozess zum Erhitzen einer Flüssigkeit verwendet wird. Dafür soll das folgende Rezept realisiert werden:

1. Behälter füllen.
2. Rührer anschalten.
3. Heizung anschalten, bis die Flüssigkeit eine Solltemperatur erreicht hat.
4. Rührer ausschalten.
5. Behälter entleeren.

Um diese Spezifikation zu erfüllen, muss die Steuerung die diskreten Eingangsgrößen  $v_1$ ,  $v_2$ ,  $v_3$  und  $v_4$  geeignet festlegen, also jedem Zustand des Batchprozesses den nächsten Stelleingriff zuordnen. So müssen nach dem Befüllen des Behälters der Rührer und die Heizung angeschaltet werden, aber es ist beispielsweise nicht sinnvoll, in dieser Situation den Behälter durch Öffnen des Ventils  $V_2$  zu entleeren. Während ein Modell, das die Steuerstrecke bezüglich des gewünschten Verhaltens beschreibt, nur die Steuereingriffe berücksichtigt, die zum Anschalten des Rührers und der Heizung führen, müsste ein Modell, das die Strecke für alle möglichen Eingaben darstellt, das Verhalten des Behälters auch beim gleichzeitigen Öffnen aller Ventile oder beim Öffnen des Ablassventils und gleichzeitigem Einschalten des Rührers usw. erfassen. Bei den vier binären Eingangsgrößen  $v_1$ ,  $v_2$ ,  $v_3$  und  $v_4$  des Batchprozesses muss die Zustandsübergangsfunktion  $G$  also in jedem Zustand für  $2^4 = 16$  mögliche Eingaben aufgestellt werden, während bei Beachtung des zu realisierenden Rezepts für jeden Zustand nur eine Eingabe berücksichtigt werden muss. Das im Beispiel 12.9 entwickelte Petrinetz ist auf diese Weise entstanden. Es beschreibt den Batchprozess unter der Wirkung einer Regelung, die das gegebene Rezept realisiert.

Für die Realisierung des Rezepts wird das Modell des gesteuerten Systems als Reglergesetz interpretiert. Der Regelkreis ist zu den Zeitpunkten geschlossen, an denen die Strecke einen neuen Ausgabe erzeugt, der einen Markenfluss im Petrinetz (also im Regler) bewirkt und dadurch eine neue Ausgabe des Petrinetzes hervorruft, die als Stelleingriff an die Regelstrecke weitergeleitet wird.  $\square$

Bei diesem Entwurfsweg wird die diskrete Steuerung gar nicht entworfen, sondern auf heuristische Weise festgelegt. Das entstehende Steuerungsgesetz ist so allgemein, dass es sowohl Komponenten enthalten kann, die im geschlossenen Kreis arbeiten, als auch Komponenten, die in der offenen Kette wirken.

Dieses Vorgehen lässt sich folgendermaßen zusammenfassen:

---

**Entwurfsverfahren 14.2** *Heuristische Festlegung des Steuerungsgesetzes*

---

- Gegeben:** Modell der Steuerstrecke  $\Sigma_S$   
Spezifikation des Verhaltens des gesteuerten Systems  $\bar{\Sigma}$
1. Stellen Sie ein Modell für das gesteuerte System  $\bar{\Sigma}$  auf, das das gewünschte Verhalten wiedergibt.
  2. Interpretieren Sie das erhaltene Modell als Steuerungsgesetz  $\Sigma_R$ .
  3. Analysieren Sie das Verhalten des gesteuerten Systems.
  4. Bewerten Sie die Güte des gesteuerten Systems: Sind die Spezifikationen erfüllt, so ist das Entwurfsproblem gelöst. Andernfalls müssen Sie den Entwurf ab Schritt 1 wiederholen.

**Ergebnis:** Steuerungsgesetz  $\Sigma_R$ .

---

Das Modell des gesteuerten Systems wird als Steuerungsgesetz verwendet, wobei das Schalten des Modells durch die Ausgaben der Strecke bewirkt wird und dabei die Eingaben für die Strecke erzeugt. Zu diesem Zweck werden an die Zustandsübergänge des Modells die dabei auftretenden Eingaben und Ausgaben der Strecke geschrieben.

Die Auswahl des Steuerungsgesetzes ist bei diesem Vorgehen ein Bestandteil der Modellbildung des gesteuerten Systems. Dies gelingt nur deshalb, weil bei diskreten Systemen einzelne Zustandsübergänge der Strecke oder eines Teils der Strecke relativ unabhängig von anderen Zustandsübergängen bzw. von Zustandsübergängen anderer Teile der Strecke ablaufen. Man kann die Modellbildung deshalb schrittweise für kleine Teile der betrachteten Anlage und nacheinander für die einzelnen Zustandsübergänge durchführen.

**Rechnergestützter Entwurf diskreter Steuerungen.** Das Entwurfsverfahren 14.2 enthält den Entwurf des Steuerungsgesetzes nicht mehr als selbstständigen Schritt. Dies hat die wichtige Konsequenz, dass die Wahl der Steuereingriffe nicht mehr formalisierbar ist, d. h., man kann für den ersten Schritt des Entwurfsvorgehens keine allgemeingültigen Methoden angeben und daraus rechnergestützte Werkzeuge entwickeln. Die Unterstützung des heuristischen Steuerungsentwurfs für diskrete Systeme durch moderne Softwarepakete bezieht sich deshalb nicht auf den Modellbildungsschritt oder die Auswahl des Steuerungsgesetzes, sondern auf die Art und Weise, wie das dabei entstehende Modell des gesteuerten Systems notiert und für die im zweiten Schritt durchzuführende (rechnergestützte) Analyse aufbereitet wird.

Eine aktuelle Entwicklungslinie des Steuerungsentwurfs konzentriert sich auf die Erarbeitung von Normen, die sich mit der Notation von Steuerungsalgorithmen beschäftigen und die Überprüfbarkeit der entwickelten Algorithmen sowie ihre Übertragbarkeit von einer auf eine andere Gerätetechnik ermöglichen. Diese Methoden garantieren jedoch keinen fehlerfreien Entwurf. Ausgiebig untersucht wird derzeit das Thema, aus einer Beschreibung des Steuerungs-

algorithmus durch einen Automaten bzw. ein Petrinetz das Programm für eine speicherprogrammierbare Steuerung zu generieren.

Ein weiteres wichtiges Problem des heute praktizierten heuristischen Steuerungsentwurfs besteht deshalb in der Überprüfung der Funktionsweise der Steuerung. Wenn man schon nicht auf systematische Syntheseverfahren zurückgreifen kann, will man wenigstens die heuristisch erhaltene Steuerung automatisch auf Richtigkeit überprüfen. Diesen Schritt nennt man *Verifikation* (Abschn. 14.4).

## 14.2 Reglerentwurf für Automaten

### 14.2.1 Entwurfsproblem

Dieser Abschnitt behandelt das Entwurfsverfahren 14.1 für Regelstrecken, die durch einen Automaten beschrieben sind und bei denen das Steuerungsziel durch

- eine Menge  $\bar{Z}$  verbotener Zustände,
- eine Menge  $\bar{\mathcal{E}}$  verbotener Zustandsübergänge (Ereignisse),
- einen Zielzustand  $z_e$

vorgegeben ist. Die Steuereinrichtung soll also durch eine geeignete Wahl der Eingaben erreichen, dass die Strecke eine Zustandsfolge

$$Z = (z(0), z(1), z(2), \dots, z(k_e)) \quad \text{mit } z(0) = z_0 \quad \text{und } z(k_e) = z_e$$

erzeugt, in der kein verbotener Zustand auftritt und kein verbotener Zustandsübergang vorkommt:

$$\begin{aligned} z(k) &\notin \bar{Z} && \text{für } k = 1, 2, \dots, k_e - 1 \\ (z(k+1), z(k)) &\notin \bar{\mathcal{E}} && \text{für } k = 0, 1, 2, \dots, k_e - 2. \end{aligned}$$

Es handelt sich dabei um eine Steuerung im geschlossenen Wirkungskreis (Regelung).

### 14.2.2 Reglerentwurf für deterministische Automaten

Das Entwurfsvorgehen wird zunächst für Regelstrecken eingeführt, die durch deterministische Automaten beschrieben sind, und später für nichtdeterministische Automaten erweitert. Es wird vorausgesetzt, dass die Ausgabe den Streckenzustand beschreibt ( $w(k) = z(k)$ ), so dass die Strecke durch einen Automaten

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, G, z_0)$$

beschrieben wird.

Falls der Zustand  $z(k)$  nicht gemessen werden kann, so kann ein Zustandsbeobachter verwendet werden, der den Zustand  $z(k)$  aus einer Ausgabefolge rekonstruiert (Kap. 15). Wenn der

Anfangszustand  $z_0$  bekannt ist, kann der Zustand der Regelstrecke auf Grund des deterministischen Verhaltens mit dem Modell  $\mathcal{A}$  verfolgt werden, wobei dieses Modell dann ein Bestandteil der zu entwerfenden Regelung ist.

**Entwurfsvorgehen.** Die Entwurfsschritte resultieren aus der folgenden Überlegung. Für jeden Zustand  $z$  der Regelstrecke beschreibt der deterministische Automat  $\mathcal{A}$  für jede Eingabe  $v \in \mathcal{V}$  eindeutig, welchen Zustandsübergang die Strecke ausführt. Das zugelassene Verhalten der Strecke wird folglich durch einen Automaten repräsentiert, der aus dem Modell der Regelstrecke durch Streichen aller verbotenen Zustände und Zustandsübergänge entsteht und der als reduzierter Automat bezeichnet wird. Dieses Vorgehen wird im Folgenden anhand des Automatengraphen der Regelstrecke erläutert:

**1. Streichen aller Zustandsübergänge  $(z', z) \in \bar{\mathcal{E}}$ .**

Diesen Zustandsübergängen ist eine Eingabe  $v$  zugeordnet, die man an der Kante  $z \xrightarrow{v} z'$  des Automatengraphen ablesen kann. Verbotene Zustandsübergänge führen also auf die Menge

$$\bar{\mathcal{V}}_Z(z) = \{v \mid z' = G(z, v), (z', z) \in \bar{\mathcal{E}}\} \quad (14.1)$$

der im Zustand  $z$  verbotenen Eingaben. Sie werden aus dem Automatengraphen gestrichen.

**2. Streichen aller Zustände  $z \in \bar{\mathcal{Z}}$ .**

Die zu verbotenen Zuständen  $z' \in \bar{\mathcal{Z}}$  führenden Zustandsübergänge resultieren in der zweiten Menge verbotener Eingaben, die durch

$$\bar{\mathcal{V}}_E(z) = \{v \mid z' = G(z, v), z' \in \bar{\mathcal{Z}}\} \quad (14.2)$$

beschrieben wird. Diese Zustände sowie alle Zustandsübergänge, die zu diesen Zuständen führen oder von ihnen ausgehen werden aus dem Automatengraphen gestrichen.

**3. Streichen aller nicht erreichbaren Zustände.**

Wenn in dem verbliebenen Graphen Zustände existieren, zu denen keine Kanten führen, können diese Zustände zusammen mit den von ihnen ausgehenden Kanten aus dem Graphen gestrichen werden.

Der reduzierte Automat beschreibt alle zugelassenen Zustände und Zustandsübergänge der Regelstrecke. Er wird mit

$$\tilde{\mathcal{A}} = (\tilde{\mathcal{Z}}, \tilde{\mathcal{V}}, \tilde{G}, z_0)$$

bezeichnet, wobei  $\tilde{\mathcal{Z}} = \mathcal{Z} \setminus \bar{\mathcal{Z}}$  und  $\tilde{\mathcal{V}} \subseteq \mathcal{V}$  gilt und die Zustandsübergangsfunktion  $\tilde{G}$  die im reduzierten Automaten „übrig gebliebenen“ Zustandsübergänge darstellt.

Um die gewünschte Überführung des zu steuernden Prozesses vom Anfangszustand  $z_0$  in den Zielzustand  $z_e$  zu realisieren, sucht man im reduzierten Automatengraphen einen Pfad

$$Z(0 \dots k_e) = (z_0, z(1), z(2), \dots, z_e)$$

von  $z_0$  nach  $z_e$  und liest an den Kanten dieses Pfades die Steuerfolgen  $V(0 \dots k_e)$  ab, wobei  $k_e$  die Anzahl der dabei notwendigen Zustandsübergänge beschreibt. Die Steuerfolge kann als Rückführung

$$v(k) = H_s(z(k)) \quad (14.3)$$

des gemessenen Zustands  $z$  auf den Eingang  $v$  der Regelstrecke realisiert werden, wobei das Reglergesetz  $H_s$  so gewählt wird, dass es für jedes Zustandspaar  $(z(k+1), z(k))$ , das in der Zustandsfolge  $Z$  aufeinander folgt, dem Zustand  $z(k)$  diejenige Eingabe  $v(k)$  zuordnet, die den Zustandsübergang zu  $z(k+1)$  hervorruft:

$$v(k) : z(k+1) = G(z(k), v(k)) \quad \text{für } (z(k+1), z(k)) \in Z(0 \dots k_e). \quad (14.4)$$

Das Reglergesetz hat den Charakter einer Verknüpfungssteuerung (13.1), denn der Regler hat keine eigene Dynamik und bestimmt die aktuelle Eingabe  $v(k)$  der Regelstrecke aus der aktuellen Messung  $z(k)$  des Zustands.

Das Entwurfsvorgehen führt auf den folgenden Algorithmus:

---

### Entwurfsverfahren 14.3 Reglerentwurf für deterministische Automaten

---

**Gegeben:** Deterministischer Automat  $\mathcal{A}$ , der die Regelstrecke beschreibt  
 Menge  $\bar{Z}$  verbotener Zustände  
 Menge  $\bar{E}$  verbotener Zustandsübergänge  
 Zielzustand  $z_e$

1. Ermitteln Sie den reduzierten Automaten  $\tilde{\mathcal{A}}$  durch Streichen der verbotenen Zustände und Zustandsübergänge.
2. Suchen Sie im reduzierten Automatengraphen einen Pfad von  $z_0$  nach  $z_e$ .
3. Stellen Sie das Reglergesetz  $H_s$  entsprechend Gln. (14.3), (14.4) auf.

**Ergebnis:** Verknüpfungssteuerung (14.3), die die Strecke vom aktuellen Zustand  $z_0$  in den Zielzustand  $z_e$  überführt.

---

Das auf diese Weise ermittelte Reglergesetz existiert, wenn es im Graphen des reduzierten Automaten  $\tilde{\mathcal{A}}$  einen Pfad von  $z_0$  nach  $z_e$  gibt. Wenn kein derartiger Pfad existiert, so gibt es keine Steuerung, die die Strecke unter Umgehung der verbotenen Zustände und Zustandsübergänge in den Zielzustand überführt. Der Vorteil des hier beschriebenen systematischen Reglerentwurfs gegenüber dem im Abschn. 14.1.2 beschriebenen heuristischen Vorgehen besteht also nicht nur in dem direkten, in einen Algorithmus überführbaren und deshalb auch rechnergestützt durchführbaren Entwurfsweg, sondern auch in der Tatsache, dass dabei gleichzeitig die Existenz der Lösung geprüft werden kann. Der Preis dafür liegt, wie im Abschn. 14.1 erläutert wurde, im Aufwand für die Modellbildung der Regelstrecke.

Das Ergebnis ist eine Verknüpfungssteuerung, wenn voraussetzungsgemäß der Zustand der Regelstrecke messbar ist. Ist die Voraussetzung nicht erfüllt, muss der Zustand mit dem Modell der Strecke aus den Messgrößen rekonstruiert werden. Der dabei verwendete Beobachter zusammen mit dem Reglergesetz (14.3) ergibt einen Steuerungsalgorithmus mit eigenem „Gedächtnis“, also eine Ablaufsteuerung.



**Verhinderung verbotener Zustände.** Obwohl die erhaltene Regelung die verbotenen Zustände und Zustandsübergänge meidet, ist man in vielen Anwendungen daran interessiert, auch im Fehlerfalle sicherzustellen, dass diese Zustände bzw. Zustandsübergänge nicht aktiviert werden. Dafür kann man eine Verriegelungssteuerung verwenden, deren Funktion  $H_s$  direkt aus der Menge  $\bar{V}_Z(z) \cup \bar{V}_E(z)$  der verbotenen Eingaben abgelesen wird. Für jeden Zustand  $z$  soll die Verriegelungssteuerung verhindern, dass die Eingaben

$$v \in \bar{V}_Z(z) \cup \bar{V}_E(z)$$

von der Steuereinrichtung an die Strecke weitergeleitet werden. Die Verriegelungssteuerung filtert also aus den Eingaben, die der Regler erzeugt, diejenigen heraus, die erlaubt sind, und verhindert, dass verbotene Eingriffe an der Strecke wirksam werden. Dies ist ein zusätzlicher Schutz der Strecke, der bei dem hier beschriebenen Entwurfsvorgehen als Nebenergebnis entsteht und der die Toleranz des Regelkreises in Bezug auf Fehler in den Stell- und Messgliedern sowie im Prozess erhöht (fehlertolerante Steuerung). Er ist auch dann sinnvoll, wenn zusätzlich zu den Stelleingriffen des Reglers operative Stelleingriffe durch das Bedienpersonal möglich sind.

### 14.2.3 Erweiterungen

Bei der beschriebenen Vorgehensweise ist das Reglergesetz  $H_s$  nur für alle Zustände  $z$  definiert, die auf der Zustandsfolge  $Z(0 \dots k_e)$  von  $z_0$  nach  $z_e$  liegen. Wenn man jedoch den Entwurf für andere Anfangszustände  $z_0$  wiederholt, so wird die Funktion  $s$  auch für die anderen Zustände  $z \in \tilde{Z}$  definiert und ist dann ein Reglergesetz, das die Regelstrecke von einem beliebigen Anfangszustand in den gewünschten Endzustand steuert.

Andererseits kann man das Entwurfsvorgehen auch für andere Zielzustände wiederholen und das Reglergesetz dann in Abhängigkeit vom Zielzustand formulieren, so dass der aktuelle Stelleingriff  $v(k)$  nicht mehr nur vom aktuellen Zustand  $z(k)$ , sondern auch vom gewünschten Endzustand  $z_e$  abhängt:

$$v(k) = H_s(z(k), z_e).$$

#### Beispiel 14.2 Steuerung eines Personenaufzugs

Ein Personenaufzug soll je nach Anforderung der Nutzer zwischen dem Erdgeschoss (EG) und dem ersten Obergeschoss (OG) eines Gebäudes hin- und herfahren. Die Steuerung muss dafür den Fahrstuhlmotor in der entsprechenden Richtung an- und abschalten und die Tür des Fahrkorbs öffnen und schließen (Abb. 14.1). Dabei darf der Fahrkorb nicht bei offener Tür bewegt, die Fahrt nicht zwischen den Etagen unterbrochen und die Tür nicht während der Fahrt geöffnet werden.

Um diese Steuerung mit dem Entwurfverfahren 14.3 entwerfen zu können, muss zunächst ein Modell des Personenaufzugs aufgestellt werden. Der Zustand des Aufzugs wird für dieses Beispiel durch die Position bzw. Bewegungsrichtung des Fahrkorbs und die Position der Tür beschrieben:

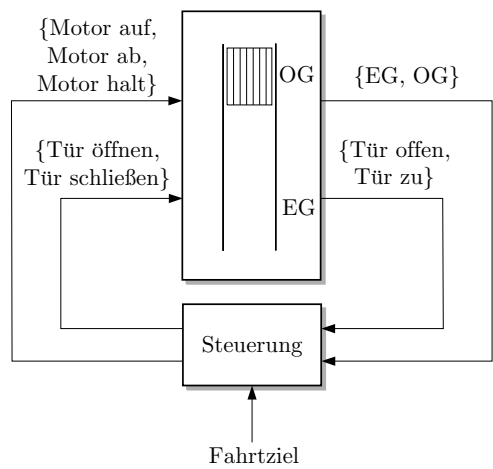


Abb. 14.1: Blockschaltbild des gesteuerten Personenaufzugs

Variable	Wertemenge	Abkürzung
Fahrkorb	steht im OG fährt vom OG ins EG steht zwischen OG und EG fährt vom EG ins OG steht im EG	OG OG → EG OG/EG EG → OG EG
Tür	offen geschlossen	Tür offen Tür zu

Beide Größen sind in einem zweidimensionalen Vektor zusammengefasst, der den Zustand des Aufzugs beschreibt. Im Automatengraphen in Abb. 14.2 sind die dabei entstehenden Zustände mit  $z_1$  bis  $z_{10}$  bezeichnet und so angeordnet, dass sich nach oben die Position des Fahrkorbs verändert und nach rechts die Türposition.

Die Steuereinrichtung kann folgende Eingaben für den Fahrstuhl erzeugen:

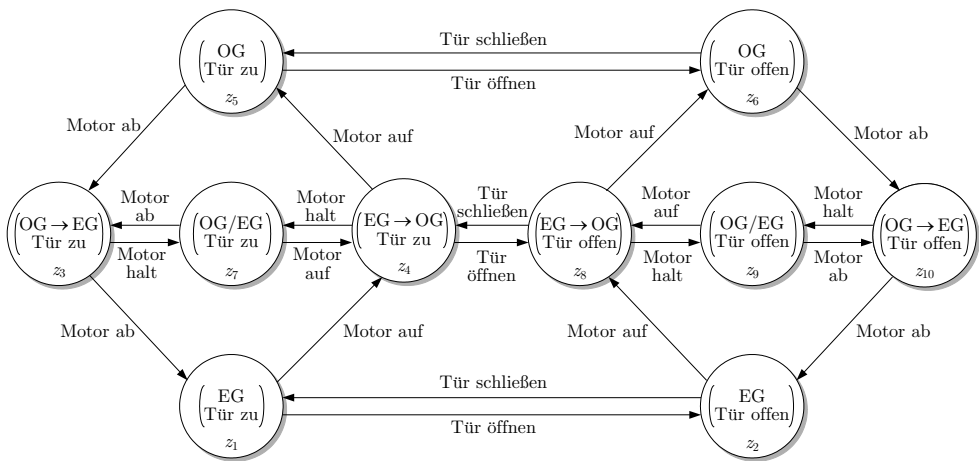
$v$	Bedeutung	Abkürzung
1	Tür öffnen	Tür öffnen
2	Tür schließen	Tür schließen
3	Fahrkorbmotor für Aufwärtsbewegung anschalten	Motor auf
4	Fahrkorbmotor für Abwärtsbewegung anschalten	Motor ab
5	Fahrkorbmotor während der Bewegung abschalten	Motor halt

Zur Vereinfachung der Darstellung wird davon ausgegangen, dass der Fahrkorbmotor und der Türmotor durch Endlagenschalter des Fahrkorbs bzw. der Tür automatisch abgeschaltet werden, so dass diese Aktionen nicht von der hier betrachteten Steuerung vorgegeben werden müssen und deshalb beim Entwurf unberücksichtigt bleiben können. Auch sorgen unterlagerte Regelungen dafür, dass die Geschwindigkeiten des Fahrkorbs bzw. der Tür unabhängig von der Belastung und von äußeren Störungen in vorgegebenen Bereichen liegen.

Die Ausgaben des Personenaufzugs werden durch Endlagenschalter des Fahrkorbs und der Tür erzeugt:

$w$	Bedeutung	Abkürzung
1	Die Tür ist offen.	Tür offen
2	Die Tür ist geschlossen.	Tür zu
3	Der Fahrkorb steht im Erdgeschoss.	EG
4	Der Fahrkorb steht im Obergeschoss.	OG

Aus diesen Ausgaben kann der Zustand des Personenaufzugs vollständig bestimmt werden, so dass im Folgenden davon ausgegangen werden kann, dass der Regler den Zustand  $z(k)$  des Fahrstuhls kennt.



**Abb. 14.2:** Automat zur Beschreibung der möglichen Bewegungen des Personenaufzugs

Abbildung 14.2 zeigt das Modell des Personenaufzugs. Da der Fahrkorbmotor zu beliebigen Zeiten abgeschaltet werden kann, kann der Fahrkorb auch zwischen den Etagen anhalten. Natürlich ist ohne eine Steuerung nicht gesichert, dass der Fahrkorb nur mit geschlossener Tür bewegt wird. Deshalb beschreibt das Modell auch die Fälle, in denen der Fahrkorb mit offener Tür fährt und die Tür beim Stillstand zwischen den Etagen geöffnet bzw. geschlossen wird. Das Öffnen und Schließen der Tür während der Fahrt ist auch möglich und wurde nur zur Vereinfachung der Abbildung weggelassen.

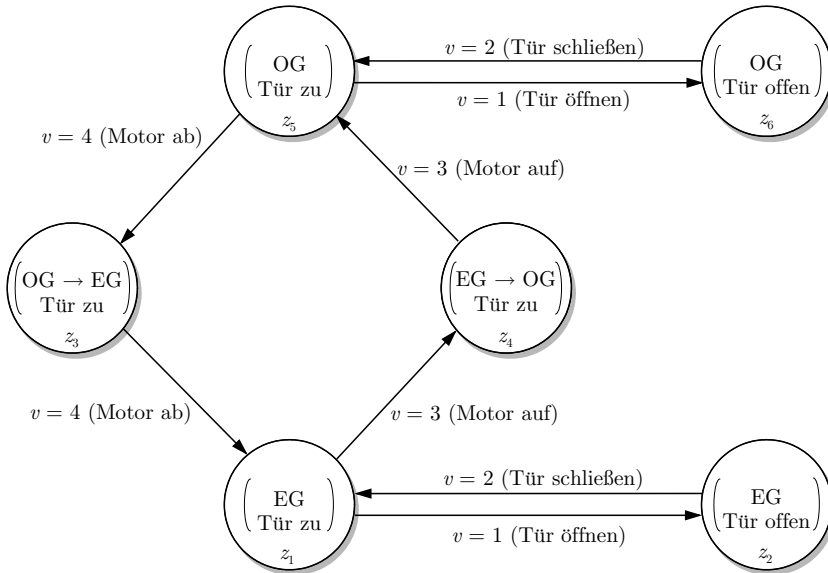
Der Regler soll so entworfen werden, dass folgende Zustände und Zustandsübergänge vermieden werden:

- **Verbotene Zustände:** Anhalten des Fahrkorbs zwischen den Etagen

$$\bar{Z} = \{z_7, z_9\}$$

- **Verbotene Zustandsübergänge:** Fahrt mit offener Tür; Öffnen der Tür zwischen den Etagen

$$\bar{E} = \{z_2 \rightarrow z_8, z_6 \rightarrow z_{10}, z_4 \rightarrow z_8, z_3 \rightarrow z_{10}\}$$



**Abb. 14.3:** Reduzierter Automat zur Beschreibung der erlaubten Bewegungen des Personenaufzugs

Der Entwurfsalgorithmus 14.3 beinhaltet folgende Schritte:

1. Es wird der reduzierte Automat durch Streichen der verbotenen Zustände und Zustandsübergänge gebildet. Der Graph dieses Automaten ist in Abb. 14.3 zu sehen. Es wurden auch die Zustände  $z_8$  und  $z_{10}$  gestrichen, weil diese Zustände im resultierenden Automatengraphen von den anderen nicht mehr erreichbar sind. Der reduzierte Automat beschreibt nur noch erlaubte Bewegungen. Es ist zu erkennen, dass entsprechend dieses Modells die Tür nur dann geöffnet und geschlossen werden kann, wenn der Fahrkorb im Erdgeschoss oder im Obergeschoss steht. Auch ist ein Anhalten des Fahrkorbs zwischen den Etagen ausgeschlossen.
2. Das Reglergesetz, durch das eine gewünschte Bewegung des Fahrstuhls realisiert wird, hängt vom Steuerungsziel  $z_e$  ab. Wenn der Fahrkorb im Obergeschoss steht und das Fahrziel „OG“ heißt, so ist selbstverständlich nichts zu tun. Gleiches gilt für das Erdgeschoss.

Es wird jetzt angenommen, dass sich der Fahrstuhl im Zustand

$$z_0 = z_2 = \begin{pmatrix} \text{EG} \\ \text{Tür offen} \end{pmatrix}$$

befindet und ins Obergeschoss fahren soll. Das heißt, er soll den Zielzustand

$$z_e = z_6 = \begin{pmatrix} \text{OG} \\ \text{Tür offen} \end{pmatrix}$$

annehmen. Die dafür notwendige Steuerfolge liest man an dem diese Zustände verbindenden Pfad im reduzierten Automatengraphen ab:

$$V(0...3) = (2, 3, 3, 1)$$

Außerdem kann man ablesen, welche Zustandsfolge der Personenaufzug dann durchläuft:

$$Z(0...4) = (z_2, z_1, z_4, z_5, z_6).$$

Daraus erhält man das folgende Reglergesetz:

$$s(z, z_6) = \begin{array}{|c|c|} \hline v & z \\ \hline 2 & z_2 \\ 3 & z_1 \\ 3 & z_4 \\ 1 & z_5 \\ \hline \end{array}$$

Dieses Reglergesetz gilt nur für den hier beschriebenen Fall, dass der Fahrstuhl ins Obergeschoss fahren soll ( $z_e = z_6$ ). Die Fahrkorbbewegung wird durch die Betätigung der Taste, die die Fahrt ins Obergeschoss signalisiert, aktiviert.

**Diskussion.** Das Beispiel zeigt, wie man ein diskretes Reglergesetz *systematisch* aus einer Beschreibung der Regelstrecke und der Spezifikation für den Regelkreis ableiten kann.

Ein wichtiger Vorteil dieses Entwurfsvorgehens ist die Einfachheit des Reglergesetzes, das hier als eine kleine Tabelle notiert ist, die dem aktuellen Zustand den Steuereingriff zuordnet. Beispiele mit heuristisch entworfenen Steuerungen zeigen, dass das dabei erhaltene Reglergesetz auch bei einfachen Steuerungsaufgaben viel komplizierter ist (siehe z. B. Beispiele 13.2 und 14.1 sowie Aufg. 14.4).

Das hier betrachtete Beispiel ist allerdings sehr einfach, denn der Fahrstuhl kann nur zwei Bewegungen ausführen (Fahrkorb nach oben oder unten bewegen, Tür öffnen und schließen) und es wurde ein Personenaufzug betrachtet, der nur zwischen zwei Etagen pendelt. Dennoch hat der Automat, der die Bewegungsmöglichkeiten des Fahrstuhls beschreibt, 10 Zustände und 22 Zustandsübergänge. Man kann sich vorstellen, wie groß das Modell eines über 25 Etagen fahrenden Fahrstuhls ist.

Die große Modellkomplexität entsteht dadurch, dass das Modell auch viele nicht genutzte, nicht erwünschte oder verbotene Bewegungen beschreibt. Darunter sind auch Zustandsübergänge, die in Bezug zur technischen Funktion der Regelstrecke unsinnig sind, aber bei einer vollständigen Modellbildung mit betrachtet werden müssen. Bei diesem Beispiel fallen in diese Rubrik die Zustandswechsel, die das Anhalten bzw. Weiter- oder Zurückfahren des Fahrkorbs zwischen den Etagen beschreiben. Die Komplexität des vollständigen Modells ist der Grund dafür, dass diskrete Regler in der Praxis häufig nicht auf dem hier beschriebenen Weg entworfen, sondern heuristisch festgelegt werden.  $\square$

#### 14.2.4 Reglerentwurf für nichtdeterministische Automaten

Das Entwurfsvorgehen muss für nichtdeterministische Automaten erweitert werden, weil bei diesen Automaten zu einem Paar  $(z, v)$  mehr als ein Nachfolgezustand  $z'$  gehören kann. Deshalb reicht es nicht mehr aus, die verbotenen Zustände und Zustandsübergänge aus dem Automaten zu streichen. Man muss darüber hinaus auch diejenigen Zustandsübergänge verbieten, die für dieselben Eingaben  $v \in \bar{V}_Z(z) \cup \bar{V}_E(z)$  wie die bereits gestrichenen Elemente auftreten (vgl. Gl. (14.1), (14.2)).

Der Grund dafür liegt in der Vorgehensweise, dass im zweiten Entwurfsschritt für einen gegebenen Anfangszustand  $z_0$  und Endzustand  $z_e$  im reduzierten Graphen nach Pfaden gesucht und von diesen Pfaden die Steuereingriffe abgelesen werden. Wenn sich darunter Steuereingriffe  $v$  befinden, die im reduzierten Graphen zu einem gewünschten Nachfolgezustand

$z' \in G(z, v)$  führen, aber auf Grund des Nichtdeterminismus auch einen verbotenen Zustand  $\bar{z} \in G(z, v)$  hervorrufen können, so ist nicht gesichert, dass der gewünschte Zustand  $z'$  und nicht der verbotene Zustand bzw. Zustandsübergang eintritt.

Auch im zweiten Entwurfsschritt ist eine Erweiterung notwendig. Um zu sichern, dass die Regelstrecke tatsächlich den gewünschten Endzustand  $z_e$  erreicht, muss man außer einem Pfad von  $z_0$  nach  $z_e$  alle nichtdeterministischen Zustandsübergänge mit denselben Eingaben betrachten. Nur wenn man entweder einen Pfad ohne nichtdeterministische Zustandsübergänge findet oder wenn das Modell der Strecke zeigt, dass auch die durch die nichtdeterministischen Übergänge entstehenden Pfade zum Zielzustand führen, wird der Endzustand  $z_e$  für das ausgewählte Reglergesetz erreicht.

## 14.3 Reglerentwurf für Petrinetze

### 14.3.1 Entwurf mit Hilfe des Erreichbarkeitsgraphen

Das im Abschn. 14.2 für Automaten beschriebene Entwurfsvorgehen kann auf Petrinetze übertragen werden, wenn man an Stelle des Automatengraphen mit dem Erreichbarkeitsgraphen des Petrinetzes arbeitet. Der dabei erhaltene Pfad von der Anfangsmarkierung zur Zielmarkierung beschreibt, welche Transitionenfolge durch den Eingriff der Steuereinrichtung realisiert werden muss. Auch hier entsteht eine Steuerung im geschlossenen Wirkungskreis.

**Regelungsaufgabe.** Um diese Idee im Einzelnen zu erläutern, wird zunächst die Regelungsaufgabe für das Petrinetz (11.65) formuliert. Gegeben sind

- eine Anfangsmarkierung  $p_0$ ,
- eine Menge  $\bar{\mathcal{P}}$  verbotener Markierungen,
- eine Menge  $\bar{\mathcal{T}}$  verbotener Transitionen,
- eine Zielmarkierung  $p_e$ .

Gesucht ist ein Regler

$$\Sigma_R : v(k) = H_s(p(k)),$$

der die Anfangsmarkierung  $p_0$  in die Zielmarkierung  $p_e$  überführt, wobei keine verbotene Transition  $t \in \bar{\mathcal{T}}$  schaltet und keine verbotene Markierungen  $p \in \bar{\mathcal{P}}$  angenommen wird. Das angegebene Reglergesetz ist realisierbar, wenn der Regler den aktuellen Markierungsvektor  $p(k)$  kennt ( $w(k) = p(k)$ ) und jeder Transition  $t_i$  eine boolesche Eingangsgröße  $v_i$  zugeordnet ist, die für alle Transitionen gemeinsam den Vektor  $v(k)$  der Stellgrößen bilden (vgl. Abschn. 11.5.2).

**Entwurf.** Beim Entwurf können verbotene Transitionen des Petrinetzes ähnlich wie verbotene Zustandsübergänge von Automaten behandelt werden. Auf Grund der o. g. Voraussetzung kann man durch die Festlegung der Eingabe  $v_i(k) = 0$  für alle  $k \geq 0$  erreichen, dass die Transition  $t_i$  niemals schaltet, und man kann deshalb diese Transition aus dem Petrinetz streichen. Dies kann man durch eine entsprechende Verriegelungssteuerung realisieren.

Besitzt eine verbotene Transition keine Eingangsgröße, so kann sie nicht durch die Regelung direkt beeinflusst werden. Dann muss die gemeinsame Markierung aller ihrer Prästellen wie eine verbotene Markierung behandelt, also die Menge  $\bar{\mathcal{P}}$  um diese Markierungen erweitert werden.

Um zu erkennen, ob verbotene Markierungen auftreten, muss man den Erreichbarkeitsgraphen bilden. Durch das Herausstreichen der verbotenen Markierungen einschließlich der zu den entsprechenden Knoten führenden und von ihnen ausgehenden Kanten erhält man einen reduzierten Erreichbarkeitsgraphen, in dem man einen Pfad für die Überführung der Anfangsmarkierung in die Zielmarkierung sucht und an den Kanten dieses Pfades das Reglergesetz abliest.

---

#### Entwurfsverfahren 14.4 Reglerentwurf für Petrinetze

---

**Gegeben:** Petrinetz, das die Regelstrecke beschreibt, mit Anfangsmarkierung  $p_0$   
 Menge  $\bar{\mathcal{P}}$  verbotener Markierungen  
 Menge  $\bar{\mathcal{T}}$  verbotener Transitionen  
 Zielmarkierung  $p_e$

1. Streichen Sie die verbotenen Transitionen aus dem Petrinetz, wodurch ein reduziertes Petrinetz entsteht. Erweitern Sie bei nicht steuerbaren verbotenen Transitionen die Menge  $\bar{\mathcal{P}}$  um die Markierung der Prästellen dieser Transitionen.
2. Bilden Sie den Erreichbarkeitsgraphen des reduzierten Petrinetzes.
3. Streichen Sie im Erreichbarkeitsgraphen die Knoten, die verbotene Markierungen enthalten, und die zugehörigen Kanten.
4. Suchen Sie einen Pfad von der Anfangsmarkierung  $p_0$  zur Zielmarkierung  $p_e$  und stellen Sie die Folge der auf diesem Pfad schaltenden Transitionen auf.
5. Bestimmen Sie das Reglergesetz aus der zur Aktivierung der ermittelten Transitionsfolge notwendigen Eingabefolge.

**Ergebnis:** Ablaufsteuerung, die die Strecke von dem durch die Anfangsmarkierung  $p_0$  dargestellten Zustand in den durch die Markierung  $p_e$  beschriebenen Endzustand überführt.

---

**Vergleich zum Reglerentwurf für Automaten.** Da das hier beschriebene Entwurfsvorgehen dem für Automaten sehr ähnlich ist, stellt sich die Frage, inwieweit sich der Entwurf einer Steuerung für einen Automaten von dem für ein Petrinetz unterscheidet. Der wichtigste Unterschied besteht in der Komplexität der Modelle der Regelstrecke und der Spezifikationen. Bezüglich des Modells wurde bereits gezeigt, dass Petrinetze häufig eine kompaktere Darstellung ereignisdiskreter Systeme ermöglichen als Automaten, insbesondere bei nebenläufigen Prozessen. Ähnliches trifft auf die Spezifikation zu. Verbotene Zustände werden beim Petrinetz dadurch spezifiziert, dass die zugehörigen Markierungen verboten werden. Außerdem kann

man das Schalten von Transitionen verbieten. Diese Forderungen sind häufig kompakter zu formulieren als die Spezifikation, die alle verbotenen Automatenzustände und Zustandsübergänge einzeln aufführt. Auch kann in Erweiterung der hier betrachteten Aufgabe an Stelle eines Zielzustands beim Petrinetz eine Zielmarkierung vorgegeben werden, die lediglich fordert, dass eine oder mehrere Stellen markiert sind, aber offen lässt, welche weiteren Stellen Marken besitzen.

Beim Entwurf verliert man allerdings die Kompaktheit der Darstellung des Streckenverhaltens durch ein Petrinetz, weil man den Erreichbarkeitsgraphen bilden und damit auf die Beschreibungsebene von Automaten zurückkehren muss.

### 14.3.2 Steuerungsentwurf unter Nutzung von S-Invarianten

In diesem Abschnitt wird ein Entwurfsverfahren für Steuerungen angegeben, das auf der Nutzung von S-Invarianten beruht und die Bildung des Erreichbarkeitsgraphen umgeht. Die entstehende Steuerung soll verhindern, dass ein System in eine vorgegebene Menge von Zuständen gelangen kann. Die Steuerstrecke wird durch ein Petrinetz beschrieben. Verbotene Zustände werden deshalb dadurch beschrieben, dass man bestimmte Markierungsvektoren für dieses Petrinetz verbietet. Das erhaltene Steuerungsgesetz kann sowohl in der offenen Kette als auch im geschlossenen Kreis angewendet werden (Beispiel 14.3).

**Steuerungsaufgabe.** Es wird davon ausgegangen, dass die Steuerstrecke durch ein Petrinetz (11.65) mit der Netzmatrix  $N$  beschrieben wird:

$$\Sigma_S : \quad p(k+1) = p(k) + N t(k), \quad p(0) = p_0. \quad (14.5)$$

Das Steuerungsziel ist so formuliert, dass der Markierungsvektor  $p(k)$  die Ungleichung

$$Lp(k) \leq b, \quad k = 0, 1, 2, \dots \quad (14.6)$$

erfüllen soll. Dabei ist  $L$  eine  $(N_s \times N)$ -Matrix, bei der  $N$  die Anzahl der Stellen im Streckenmodell und  $N_s$  die Zahl der Ungleichungsbeschränkungen angibt.  $b$  ist ein  $N_s$ -dimensionaler ganzzahliger Vektor.

Mit einer derartigen Ungleichung kann man beispielsweise ausdrücken, dass die Stellen  $p_1$  und  $p_2$  nicht gleichzeitig markiert sein sollen, indem man die Einhaltung der Beziehung

$$p_1(k) + p_2(k) \leq 1$$

fordert. Man drückt dies in der Ungleichung (14.6) durch eine Zeile  $l^T$  der Matrix  $L$  aus, in der die ersten beiden Elemente gleich eins, die anderen gleich null sind. Das zugehörige Element des Vektors  $b$  wird gleich eins gesetzt.

**Steuerungsentwurf.** Die Grundidee des Steuerungsentwurfs besteht in der Einführung von  $N_s$  zusätzlichen Stellen, auf die überzählige Marken verschoben werden, so dass die Ungleichungsbeschränkung (14.6) stets eingehalten wird. Dies geschieht zunächst dadurch, dass man einen Vektor  $p_s(k)$  einführt, mit dem die Ungleichungsbeschränkung zur Gleichung



$$Lp(k) + p_s(k) = b, \quad k = 0, 1, 2, \dots \quad (14.7)$$

ergänzt wird. Für die Steuerung macht man dann den Ansatz

$$\Sigma_R : p_s(k+1) = p_s(k) + N_s t(k), \quad p_s(0) = p_{s0}, \quad (14.8)$$

wobei  $N_s$  eine noch zu bestimmende Netzmatrix ist. Der Schaltvektor  $t(k)$  ist derselbe wie im Petrinetz (14.5) der Strecke, d. h., die Transitionen der Strecke sind auch mit den Stellen der Steuereinrichtung verbunden. Welche Verbindungen zwischen diesen Transitionen und Stellen bestehen, wird durch die noch zu bestimmende Netzmatrix  $N_s$  festgelegt.

Fasst man die Modelle (14.5) und (14.8) der Strecke und der Steuerung zusammen, so erhält man die Beziehung

$$\begin{pmatrix} p(k+1) \\ p_s(k+1) \end{pmatrix} = \begin{pmatrix} p(k) \\ p_s(k) \end{pmatrix} + \begin{pmatrix} N \\ N_s \end{pmatrix} t(k). \quad (14.9)$$

Mit diesem Modell kann man die Einhaltung der Gleichungsbedingung (14.7) überprüfen, indem man es von links mit der Matrix  $(L \ I)$  multipliziert, wobei  $I$  eine  $(N_s, N_s)$ -Einheitsmatrix ist. Dabei erhält man folgende Bedingung:

$$\underbrace{Lp(k+1) + p_s(k+1)}_{=b} = \underbrace{Lp(k) + p_s(k)}_{=b} + (LN + N_s) t(k).$$

Wenn die Bedingung (14.7) eingehalten ist, so sind die beiden markierten Summen gleich dem Vektor  $b$  aus der Ungleichungsbeschränkung (14.6). Daraus entstehen zwei Forderungen an die Steuerung. Erstens muss der rechte Ausdruck für beliebiges Schalten der Transitionen verschwinden:

$$(LN + N_s) t(k) = 0.$$

Aus der Bedingung

$$LN + N_s = 0$$

folgt für die Netzmatrix der Steuerung die Beziehung

$$N_s = -L \cdot N.$$

Zweitens muss die Gleichungsbeschränkung (14.7) auch für  $k = 0$  gelten, woraus man die Anfangsmarkierung des Reglers

$$p_s(0) = b - Lp(0)$$

berechnen kann. Zusammengefasst erhält man das folgende Steuerungsgesetz:

$$\boxed{\text{Steuerung } \Sigma_R : \begin{cases} p_s(k+1) = p_s(k) - L \cdot N t(k) \\ p_s(0) = b - Lp(0). \end{cases}} \quad (14.10)$$

**Satz 14.1 (Steuerungsentwurf mit Petrinetzen)**

Wenn die durch das Petrinetz (14.5) beschriebene Steuerstrecke einen Anfangszustand  $\mathbf{p}(0)$  hat, der der Bedingung

$$\mathbf{L}\mathbf{p}(0) \leq \mathbf{b}$$

genügt, so erfüllt das gesteuerte System (14.5), (14.10) die Spezifikation (14.6).

Dieser Satz beschreibt eine analytische Lösung der Entwurfsaufgabe, die durch das Petrinetz (14.5) für die Strecke und die Spezifikation (14.6) gegeben ist. Die Steuerungsgesetz (14.10) repräsentiert die eindeutige Lösung dieser Aufgabe.

**Interpretation.** Die entworfene Steuerung sichert, dass die Gleichungsbedingung (14.7) für alle Zeiten  $k$  durch das gesteuerte System eingehalten wird und folglich

$$(\mathbf{L} \quad \mathbf{I}) \begin{pmatrix} \mathbf{p}(k) \\ \mathbf{p}_s(k) \end{pmatrix} = \mathbf{0}$$

gilt. Das heißt, dass das Petrinetz (14.9) des gesteuerten Systems mehrere S-Invarianten besitzt, die mit den  $N_s$  Zeilen der Matrix  $(\mathbf{L} \quad \mathbf{I})$  übereinstimmen. Es gibt also  $N_s$  Mengen von Stellen, für die die gewichtete Summe aller Marken konstant ist. In diese Mengen gehen sowohl Stellen der Strecke (14.5) als auch Stellen der Steuerung (14.10) ein. Die Wichtung der Markensumme ist durch die Matrix  $\mathbf{L}$  festgelegt.

Die Konstanz der Marken in diesen Stellenmengen wird dadurch erreicht, dass die durch das Schalten der Strecke entstehende Vergrößerung oder Verkleinerung der Markenanzahl in den durch  $\mathbf{L}$  gewichteten Stellen durch einen Markenfluss von bzw. auf Stellen der Steuereinrichtung ausgeglichen wird. Multipliziert man das Modell (14.5) der Strecke von links mit der Matrix  $\mathbf{L}$ , so erhält man für die Veränderung der gewichteten Markensumme in der Strecke die Beziehung

$$\mathbf{L}(\mathbf{p}(k+1) - \mathbf{p}(k)) = \mathbf{L}\mathbf{N}\mathbf{t}(k).$$

Andererseits erhält man aus dem Steuerungsgesetz (14.10) die Beziehung

$$\mathbf{p}_s(k+1) - \mathbf{p}_s(k) = -\mathbf{L}\mathbf{N}\mathbf{t}(k).$$

Die Veränderung der gewichteten Markensumme in der Strecke wird also gerade durch eine gleich große Änderung der Markierung in der Steuereinrichtung (mit dem entgegengesetzten Vorzeichen) kompensiert. Dabei werden Marken zeitweise in Stellen des Reglers verschoben, wenn andernfalls die Markierung der Strecke die gegebene Ungleichung (14.6) verletzen würde.

Die Veränderung  $p_{si}(k+1) - p_{si}(k)$  der Markierung der  $i$ -ten Stelle der Steuerung hängt von den Matrizen  $\mathbf{L}$  und  $\mathbf{N}$  ab und kann größer als eins sein. Dann müssen mehr als eine Marke auf die  $i$ -te Stelle des Reglers fließen oder von dort abfließen. Das ist bei den hier betrachteten Petrinetzen nicht möglich, da voraussetzungsgemäß jede Stelle höchstens eine Marke aufnehmen darf, kann aber einfach dadurch realisiert werden, dass in diesem Falle das  $i$ -te Element des Vektors  $\mathbf{p}_s$  als Summe der Marken mehrerer Stellen der Steuereinrichtung interpretiert wird.

Alle diese Stellen sind mit den durch die  $i$ -te Zeile der Netzmatrix  $N_s = -LN$  beschriebenen Transitionen verbunden. Diese Erweiterung des die Steuerung beschreibenden Petrinetzes ist überflüssig, wenn man Petrinetze verwendet, deren Stellen mehr als eine Marke enthalten dürfen.

### Beispiel 14.3 Steuerung einer Sicherheitsschleuse

Bei der in Abb. 14.4 gezeigten Sicherheitsschleuse darf stets nur eine der beiden Türen geöffnet sein, was durch eine Steuerung sichergestellt werden soll. Die Türen können durch das Betätigen der innen und außen angebrachten Taster geöffnet und geschlossen werden, wobei die Kommandos von den Tastern nur dann durch die zu entwerfende Steuerung an die Türmotoren weitergeleitet werden, wenn die genannte Bedingung erfüllt ist.

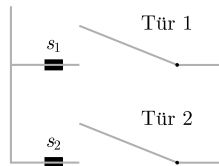


Abb. 14.4: Sicherheitsschleuse

Für die Türen kann das in Abb. 14.5 links dargestellte Petrinetz aufgestellt werden, bei dem die Stellen  $p_{11}$  und  $p_{21}$  bei geschlossener Tür und die Stellen  $p_{12}$  und  $p_{22}$  bei geöffneter Tür markiert sind. Beide Teile des Petrinetzes zusammen haben die Netzmatrix

$$N = \begin{matrix} & \begin{matrix} t_{11} & t_{12} & t_{21} & t_{22} \end{matrix} \\ \begin{matrix} p_{11} \\ p_{12} \\ p_{21} \\ p_{22} \end{matrix} & \begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \end{matrix}.$$

Die binären Eingangsgrößen  $s_{ij}$  der Transitionen repräsentieren den über die Taster abgegebenen Wunsch zum Öffnen und Schließen der Türen mit der folgenden Bedeutung:

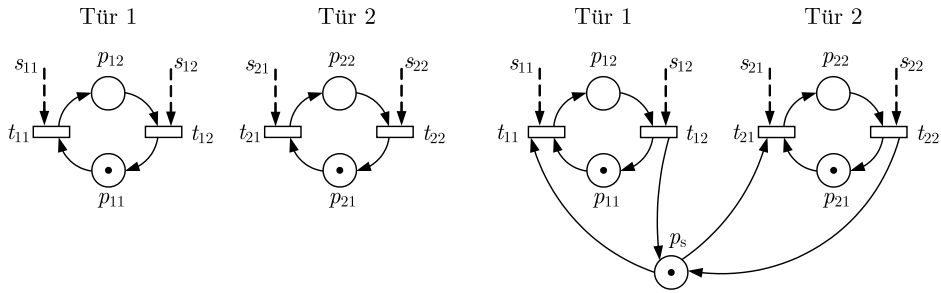
$s_{ij}$	Bedeutung
$s_{11} = 1$	Öffnen der Tür 1
$s_{12} = 1$	Schließen der Tür 1
$s_{21} = 1$	Öffnen der Tür 2
$s_{22} = 1$	Schließen der Tür 2

Die durch die Steuerung zu erfüllende Spezifikation heißt

$$p_{12}(k) + p_{22}(k) \leq 1,$$

denn wenn diese Ungleichung erfüllt ist, kann nur eine der beiden Stellen  $p_{12}$ ,  $p_{22}$  markiert und folglich nur eine der beiden Türen geöffnet sein. Die Spezifikation kann in der Form (14.6) mit

$$l^T = (0 \ 1 \ 0 \ 1) \quad \text{und} \quad b = 1$$



**Abb. 14.5:** Steuerung einer Sicherheitsschleuse: Modell der Strecke (links) und gesteuerte Tür (rechts)

geschrieben werden, wobei anstelle der Matrix  $L$  jetzt der Zeilenvektor  $l^T$  und anstelle des Vektors  $b$  der Skalar  $b$  steht.

Nach Satz 14.1 erfüllt die Steuerung

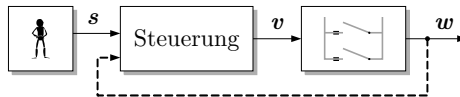
$$\Sigma_R : \begin{cases} p_s(k+1) = p_s(k) + n_s^T t(k) \\ p_s(0) = 1 - (0 \ 1 \ 0 \ 1) \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = 1 \end{cases} \quad (14.11)$$

mit der zu einem Zeilenvektor geschrumpften Netzmatrix

$$n_s^T = -l^T N = -(0 \ 1 \ 0 \ 1) \cdot \begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} = (-1 \ 1 \ -1 \ 1) \quad (14.12)$$

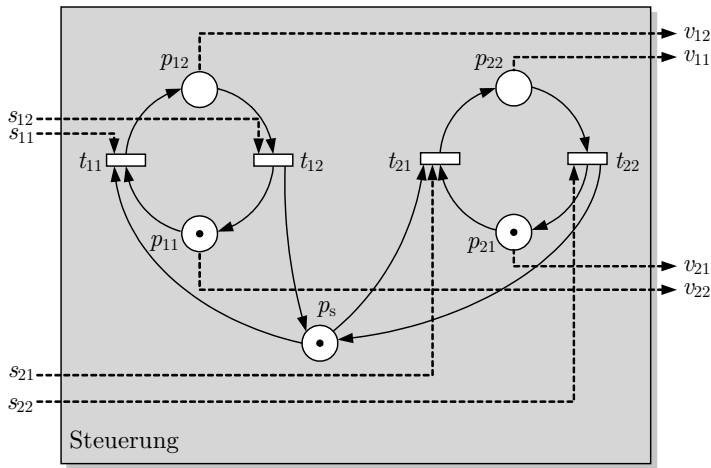
die Spezifikation. Die Dynamik der Steuereinrichtung ist durch die in Abb. 14.5 (rechts) dargestellte Stelle  $p_s$  beschrieben, die an die vier Transitionen des Streckenmodells angekoppelt ist.

Wenn eine Tür geöffnet wird, wandert die Marke von der Stelle  $p_s$  zur Stelle  $p_{12}$  bzw.  $p_{22}$ . Damit ist das Öffnen der anderen Tür blockiert. Die Summe der auf den Stellen  $p_{12}$  und  $p_{22}$  liegenden Marken ist kleiner oder höchstens gleich eins, wie es in der Spezifikation gefordert wurde. Die Stelle der Steuereinrichtung wird genutzt, um eine Marke zu „parken“, wenn beide Türen geschlossen sind. Die drei Stellen  $p_{12}$ ,  $p_{22}$  und  $p_s$  besitzen insgesamt stets eine Marke, was einer S-Invariante des Petrinetzes entspricht.



**Abb. 14.6:** Steuerungsstruktur für die Sicherheitsschleuse

**Steuerungsstrukturen.** Die Steuerung (14.11), (14.12) kann sowohl als Steuerung in der offenen Wirkungskette als auch als Steuerung im geschlossenen Wirkungskreis realisiert werden (Abb. 14.6). In beiden Fällen ist die Steuerung zwischen dem Nutzer der Sicherheitsschleuse und den Türen angeordnet. Sie gibt den Wunsch  $\mathbf{s} = (s_{11} \ s_{12} \ s_{21} \ s_{22})^T$  des Nutzers an die Türen weiter, wenn dabei die Spezifikation nicht verletzt wird. Der Vektor  $\mathbf{v} = (v_{11} \ v_{12} \ v_{21} \ v_{22})^T$  enthält die Kommandos an die Türmotoren mit genau einer Eins und sonst Nullen, wobei bei  $v_{i1} = 1$  der  $i$ -te Türmotor die Tür öffnet und bei  $v_{i2} = 1$  die  $i$ -te Tür geschlossen wird.

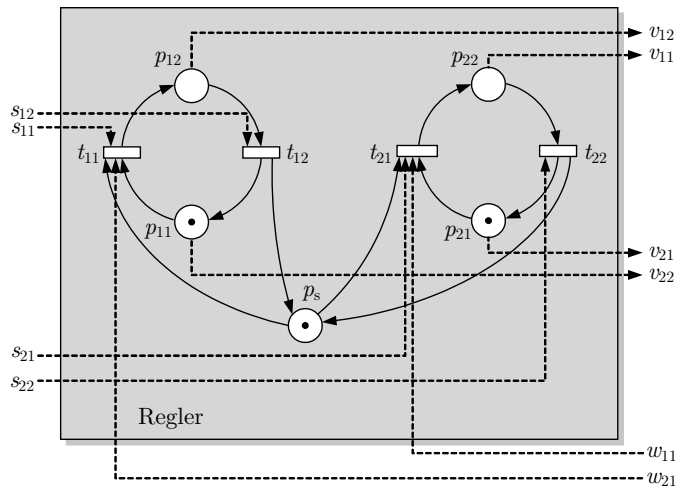


**Abb. 14.7:** Steuerung der Sicherheitsschleuse in der offenen Wirkungskette

Wie Abb. 14.7 zeigt, enthält die Steuereinrichtung das Petrinetz aus Abb. 14.5, wobei die Vorgabe  $s_{ij}$ , ( $i, j = 1, 2$ ) des Nutzers als boolesche Eingangsgröße der Transition  $t_{ij}$  zugeordnet ist. Die Ausgänge des Petrinetzes sind mit  $v_{ij}$  bezeichnet, weil sie die Eingänge der Steuerstrecke sind. Wenn die Steuerung mit der Anfangsmarkierung (14.11) initialisiert wird, arbeitet sie bestimmungsgemäß.

Das in Abb. 14.7 gezeigte Steuerungsgesetz berechnet nach jedem Eingangsereignis (Veränderung von  $s$ ) die neue Stellgröße  $v(k)$  aus  $s(k)$  und dem internen Zustand  $p(k)$ ,  $p_s(k)$  der Steuerung. Da in diese Berechnung keine Messwerte der Steuerstrecke eingehen, handelt es sich um eine Steuerung in der offenen Wirkungskette. Diese Steuerung kann nicht auf Störungen reagieren, die sich beispielsweise darin äußern, dass eine Tür nicht richtig geschlossen wird. Im Petrinetz der Steuereinrichtung sind dann zwar die Stellen  $p_{11}$  und  $p_{21}$  markiert, aber dieser Zustand stimmt nicht mit dem der Sicherheitsschleuse überein und es ist entgegen der Spezifikation möglich, die zweite Tür zusätzlich zu öffnen.

Eine Robustheit gegenüber derartigen Störungen erreicht man durch die Verwendung der entworfenen Steuerung im geschlossenen Kreis. Die Ausgaben  $w_{i1}$  der Sicherheitsschleuse werden durch Endlagenschalter an den Türen erzeugt, wobei  $w_{i1} = 1$  bedeutet, dass die  $i$ -te Tür ordnungsgemäß geschlossen ist. Die Steuereinrichtung arbeitet dann als Regler, dessen Funktion durch das Petrinetz in Abb. 14.8 beschrieben wird. Aufgrund der Rückführung kann beispielsweise die Marke von der Stelle  $p_{11}$  nur dann zur Stelle  $p_{12}$  wandern und das Öffnen von Tür 1 auslösen, wenn die Messinformation der Sicherheitsschleuse bestätigt, dass die Tür 2 geschlossen ist ( $w_{21} = 1$ ).  $\square$



**Abb. 14.8:** Steuerung der Sicherheitsschleuse im geschlossenen Kreis

Das hier behandelte Entwurfsverfahren geht von der Voraussetzung aus, dass alle Transitionen des Petrinetzes steuerbar sind. Welche Transitionen tatsächlich durch den Regler beeinflusst werden, ergibt sich aus dem Entwurfsergebnis. Die Netzmatrix  $N_s$  zeigt, welche Transitionen der Strecke mit der Steuerung verbunden werden. Die Eingangsgrößen der betreffenden Transitionen werden dabei durch Kanten zwischen den Stellen der Steuereinrichtung und den Transitionen ersetzt. Damit ist auch offensichtlich, unter welchen Bedingungen die Transition des Regelkreises schalten kann. Der Steuerungsalgorithmus ist dann ein Programm, das die zusätzlichen Stellen als Speicherplätze realisiert und die Zustandsübergänge der Strecke entsprechend den durch die Netzmatrix beschriebenen Verbindungen zwischen den zusätzlichen Stellen und den Transitionen frei gibt oder blockiert.

**Aufgabe 14.1\*** Verhinderung verbotener Zustände bei einem Personenaufzug

Durch eine Steuerung soll verhindert werden, dass der Fahrkorb eines Personenaufzugs bei offener Tür fahren kann. Verwenden Sie das in der Aufgabe 11.22 aufgestellte Petrinetz (Abb. A.55 auf S. 643), um eine Steuerung zu entwerfen, die diese Funktion erfüllt. □

**Aufgabe 14.2\*** Steuerung einer Schiffsschleuse

In einer Schleuse können Schiffe Kanalstufen überwinden. Die Schleuse verfügt über zwei Tore, die nur unter bestimmten Bedingungen geöffnet werden dürfen. Tor 1 darf nur geöffnet werden, wenn der Wasserstand in der Schleusenkammer auf dem Niveau des Oberwassers liegt, Tor 2 nur dann, wenn sich der Wasserstand auf dem Niveau des Unterwassers befindet.

1. Beschreiben Sie das Verhalten der Schleuse durch ein Petrinetz, wobei Sie die Kommandos zum Öffnen und Schließen der Tore und zum Verändern des Wasserpegels in der Schleusenkammer als Eingangsgrößen berücksichtigen.

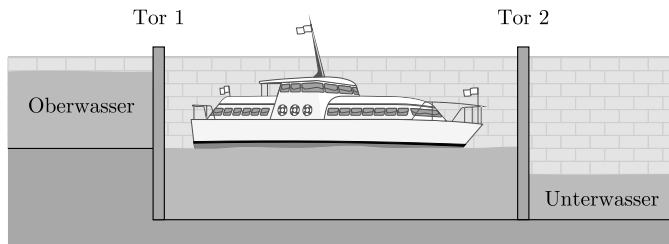


Abb. 14.9: Schiffsschleuse

2. Stellen Sie die in der Aufgabenstellung genannte Spezifikation als Ungleichungsbedingungen an die Petrinetzmarkierung dar.
3. Entwerfen Sie eine Steuerung, die die verbotenen Zustandsübergänge verhindert.
4. Arbeitet Ihre Steuerung in der offenen Wirkungskette oder im geschlossenen Kreis? □

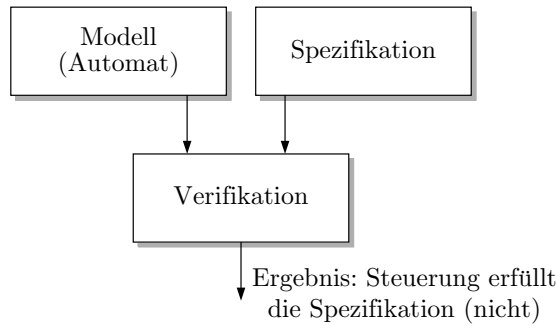
## 14.4 Verifikation diskreter Steuerungen

Während für die mit den systematischen Entwurfsverfahren aus den Abschn. 14.2 und 14.3 erhaltenen Steuerungen garantiert ist, dass die Spezifikationen erfüllt werden, muss bei heuristisch festgelegten Steuerungsgesetzen die Einhaltung der Spezifikation überprüft werden. Diesen Vorgang nennt man Verifikation.

Für eine Verifikation müssen die Spezifikationen, die typischerweise in Textform gegeben sind, in eine formalisierte Form überführt werden, wofür man die in diesem Buch eingeführten Automaten oder Petrinetze verwenden kann. Wenn man auch zeitliche Abstände zwischen bestimmten Ereignissen vorschreiben will, muss man die hier behandelten Modellformen auf zeitbewertete Automaten bzw. zeitbewertete Petrinetze erweitern. Da die Spezifikation typischerweise nicht die Arbeitsweise des Steuerungsalgorithmus selbst, sondern das Verhalten des gesteuerten Systems beschreibt, braucht man für die Verifikation auch ein diskretes Modell der Steuerstrecke (Abb. 14.10).

Für die Verifikation können die im Kap. 12 angegebenen Analyseverfahren für diskrete Systeme eingesetzt werden. Im Folgenden werden die dort erläuterten Verfahren mit den Steuerungszielen in Verbindung gebracht, die bei der Analyse des Regelkreises überprüft werden müssen, um das Grundprinzip der Verifikation zu erläutern.

Es wird davon ausgegangen, dass das gesteuerte System durch einen deterministischen Automaten bzw. ein Petrinetz beschrieben ist und dass die Spezifikationen bezüglich dieser Modelle durch die Mengen verbotener Zustände bzw. Markierungen oder Zustandsübergänge bzw. Transitionen und durch Zielzustände bzw. Zielmarkierungen formuliert sind. Die betreffenden Automaten bzw. Petrinetze sind autonom, wenn der Regelkreis bei den betrachteten Steuerungsaufgaben keine äußeren Eingaben besitzt. Wenn dem Kreis durch Nutzereingaben neue Zielzustände vorgeschrieben werden, kann man die Einhaltung der Spezifikationen dadurch prüfen, dass man den ihn für alle möglichen Eingaben nacheinander untersucht.



**Abb. 14.10:** Verifikation diskreter Steuerungen

**Analyse des durch Automaten beschriebenen Regelkreises.** Der Regelkreis sei durch den Automaten

$$\bar{\mathcal{A}} = (\bar{\mathcal{Z}}, \bar{G}, \bar{z}_0)$$

beschrieben, bei dem der Querstrich den Bezug zum Regelkreis (im Unterschied zur Regelstrecke) kennzeichnet. Wenn der Regler eine eigene Dynamik besitzt, setzt sich die Zustandsmenge  $\bar{\mathcal{Z}}$  des Regelkreises aus den Zuständen der Regelstrecke und den Zuständen des Reglers zusammen:

$$\bar{z} = \begin{pmatrix} z \\ z_s \end{pmatrix} \in \bar{\mathcal{Z}}.$$

Die Überprüfung, ob der Regler einen geforderten Zustandsübergang  $z_0 \rightarrow z_e$  der Strecke realisiert und dabei verbotene Zustände  $z \in \bar{\mathcal{Z}} \subset \mathcal{Z}$  und verbotene Zustandsübergänge  $(z', z) \in \bar{\mathcal{E}}$  meidet, bezieht sich dann nur auf die erste Komponente  $z$  des Zustands  $\bar{z}$ .

Folgende Analysemethoden können für den Regelkreis eingesetzt werden:

- Wenn die Strecke von einem Anfangszustand  $z_0$  in einen gewünschten Endzustand  $z_e$  überführt werden soll, wird die Zustandsübergangsfunktion  $\bar{G}$  genutzt, um für den Anfangszustand

$$\bar{z}_0 = \begin{pmatrix} z_0 \\ z_{s0} \end{pmatrix}$$

die Zustandstrajektorie entsprechend Gl. (12.3) auszurechnen und dann zu überprüfen, dass verbotene Zustände oder Zustandsübergänge gemieden werden.

- Durch eine Erreichbarkeitsanalyse des Automaten kann man erkennen, ob von allen möglichen Anfangszuständen  $\bar{z}_0$  die verbotenen Zustände forderungsgemäß nicht erreichbar sind.

**Analyse des durch Petrinetze beschriebenen Regelkreises.** Wenn der Regelkreis durch ein Petrinetz

$$\overline{\mathcal{PN}} = (\bar{P}, \bar{T}, \overline{\mathcal{P}rae}, \overline{\mathcal{P}ost})$$

beschrieben ist, können folgende Analyseverfahren angewendet werden:



- Mit dem Petrinetz können ausgehend von einer gegebenen Anfangsmarkierung die entstehenden Markierungsfolgen bestimmt und als Zustandsübergänge des Regelkreises interpretiert werden.
- Die Bestimmung des Erreichbarkeitsgraphen zeigt, welche Zustände das Petrinetz annehmen kann. Daran kann überprüft werden, ob verbotene Markierungen angenommen werden oder verbotene Transitionen schalten.
- Durch die Bestimmung von T- oder S-Invarianten kann das zyklische Verhalten des Regelkreises analysiert und überprüft werden, welche Stellen stets gemeinsam oder nur alternativ markiert sein können. Dieses Verfahren ist einfacher als die Analyse des Erreichbarkeitsgraphen, weil der Erreichbarkeitsgraph i. Allg. sehr groß ist.

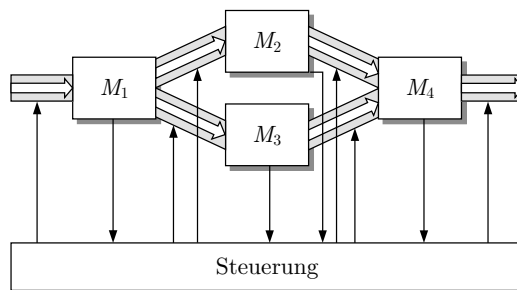
#### Aufgabe 14.3 Analyse der gesteuerten Stanze

Im Beispiel 11.2 auf S. 348 ist ein autonomer deterministischer Automat angegeben, der eine gesteuerte Stanze beschreibt. Welche Art von Analyseaufgaben können Sie mit Hilfe dieses Automaten lösen?

Stellen Sie ein Petrinetz auf, das diese gesteuerte Stanze beschreibt. Welche Aussagen können Sie anhand des Petrinetzes machen? Inwiefern unterscheiden sich diese Aussagen von denen, die Sie mit Hilfe des Automaten erhalten können?  $\square$

#### Aufgabe 14.4 Materialflusssteuerung für vier Werkzeugmaschinen

Für die vier in Abb. 14.11 gezeigten Werkzeugmaschinen ist das in Abb. 11.36 auf S. 397 dargestellte Petrinetz aufgestellt worden, bei dem die Markierung der Stelle  $p_{i1}$  anzeigt, dass die Maschine  $M_i$  ein Werkstück bearbeitet, und die Markierung der Stelle  $p_{i2}$  signalisiert, dass die Bearbeitung abgeschlossen ist. Die Transitionen  $t_0, t_1, t_2, t_3, t_4$  und  $t_5$  werden durch die Steuerung über die ihnen zugeordneten Eingangsgrößen  $v_i$  beeinflusst, wobei  $v_i = 1$  die Bewegung des entsprechenden Förderbandes auslöst.



**Abb. 14.11:** Steuerung des Materialflusses zwischen vier Werkzeugmaschinen

In dieser Aufgabe soll die Steuerung des Materialflusses entworfen werden, wobei die Steuereinrichtung entsprechend Abb. 14.11 von den vier Maschinen Meldungen  $w_{i2} = 1, (i = 1, \dots, 4)$  über



# 15

## Zustandsbeobachtung diskreter Systeme

*Dieses Kapitel zeigt, wie aus den Folgen von Eingaben und Ausgaben der Zustand eines diskreten Systems berechnet werden kann. Für deterministische und nichtdeterministische Automaten erhält man als Beobachtungsergebnis Zustandsmengen, in denen sich der aktuelle Zustand befindet. Bei stochastischen Automaten unterscheiden sich die Zustände dieser Mengen bezüglich der vom Beobachter berechneten Wahrscheinlichkeit, mit der sich der Automat in ihnen aufhält.*

### 15.1 Beobachtungsaufgabe

Die hier betrachtete Beobachtungsaufgabe ist sehr ähnlich der für kontinuierliche Systeme. Es werden die Folgen  $V$  und  $W$  der Eingaben und Ausgaben gemessen und es soll mit Hilfe des Modells der aktuelle Zustand des diskreten Systems rekonstruiert werden (Abb. 1.15 auf S. 30). Als Modell wird in diesem Kapitel ein Automat verwendet.

#### **Beobachtungsaufgabe für diskrete Systeme**

Gegeben:   Automat  
              Folge der Eingangssymbole  $V(0 \dots k_e)$   
              Folge der Ausgangssymbole  $W(0 \dots k_e)$   
Gesucht:   aktueller Zustand  $z(k_e)$

Beobachtungsaufgaben werden vor allem bei Systemen gelöst, deren Zustand nicht messbar ist. Bei nichtdeterministischen Systemen ist diese Aufgabe jedoch auch dann sinnvoll, wenn der Anfangszustand  $z(0)$  bekannt ist, denn für diese Systeme kann aus der Kenntnis des Anfangszustands und der Eingangsfolge  $V$  mit Hilfe des Modells die Zustandsentwicklung nicht eindeutig verfolgt, sondern nur eine Zustandsmenge berechnet werden, in der der aktuelle Zustand  $z(k)$  liegt. Durch die Nutzung der in der Ausgangsfolge  $W$  enthaltenen Informationen kann man mit einem Zustandsbeobachter den aktuellen Zustand genauer oder sogar exakt ermitteln.

Die Beobachtungsaufgabe soll während des Betriebs einer Anlage gelöst werden. Dabei wird der Zeithorizont  $k_e$ , für den die Eingangs- und Ausgangsfolge bekannt ist, immer länger. Für die Anwendung der Zustandsbeobachtung ist es deshalb wichtig, dass der aktuelle Zustand  $z(k_e)$  rekursiv aus dem vorherigen Zustand  $z(k_e - 1)$  unter Nutzung der aktuellen Eingabe  $v(k_e)$  und Ausgabe  $w(k_e)$  bestimmt wird und dass nicht nach jedem neuen Ereignis die gesamte Folge aller bisher erhaltenen Eingangs- und Ausgangssymbole erneut verarbeitet wird. Die in diesem Kapitel angegebenen Algorithmen besitzen diese Eigenschaft.

Die für die Lösung der Beobachtungsaufgabe verfügbaren Messinformationen stehen in den Folgen  $V$  und  $W$ . Diese werden als Paar  $(V, W)$  geschrieben und als Eingangs-Ausgangspaar (*E/A-Paar*) bezeichnet. Derselbe Begriff wird auch für das Paar  $(v(k_e), w(k_e))$  aktueller Werte verwendet, wobei aus dem Zusammenhang hervorgeht, ob das Paar der Folgen oder ein einzelnes Wertepaar gemeint ist.

Die Lösung der Beobachtungsaufgabe besteht i. Allg. nicht in einem einzigen Zustand, sondern in einer Menge von Zuständen, von der man weiß, dass sie den aktuellen Systemzustand  $z(k)$  enthält. Diese Menge wird durch  $\mathcal{Z}(k_e | k_e)$  symbolisiert, wobei sich die Zeitangabe  $k_e$  vor dem Strich auf den Zeitpunkt bezieht, für den der Zustand berechnet wurde, und die Zeitangabe hinter dem Strich den Zeithorizont angibt, für den die Folgen der Eingaben und Ausgaben bekannt sind. Mit  $\mathcal{Z}(0 | -1)$  wird dementsprechend die Zustandsmenge bezeichnet, von der man vor der Lösung der Beobachtungsaufgabe (Zeithorizont der E/A-Folgen ist  $-1$ ) weiß, dass sie den Anfangszustand (zum Zeitpunkt 0) enthält.

## 15.2 Beobachtung deterministischer und nichtdeterministischer Automaten

### 15.2.1 Beobachtungsalgorithmus für nichtdeterministische Automaten

Die Beobachtungsaufgabe wird in diesem Abschnitt für nichtdeterministische Automaten

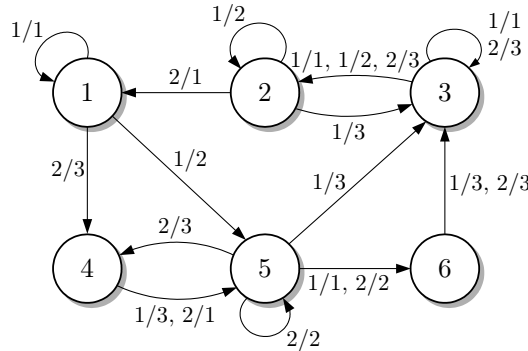
$$\mathcal{N} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, L, \mathcal{Z}_0)$$

gelöst, wobei der aktuelle Zustand  $z(k)$  in folgender Weise bestimmt wird. Wenn ein E/A-Paar  $(v(k), w(k))$  gemessen wurde, so hat der Automat einen der Zustandsübergänge

$$z(k) \xrightarrow{v(k)/w(k)} z(k+1)$$

ausgeführt, die zu diesem E/A-Paar passen. Nach der Messung eines E/A-Paares kann also eine Menge von Zuständen  $z(k)$  bestimmt werden, von denen die Bewegung des Automaten ausgegangen ist, und eine Menge von Zuständen  $z(k+1)$ , in die der Automat bei diesen

Zustandsübergängen gewechselt ist. Bevor dieses Vorgehen formal aufgeschrieben wird, soll es an einem einfachen Beispiel illustriert werden.



**Abb. 15.1:** Automatengraph für das Beispiel

**Beispiel 15.1** Zustandsbeobachtung eines nichtdeterministischen Automaten

Als Beispiel wird der in Abb. 15.1 gezeigte Automat betrachtet und angenommen, dass nichts über den Anfangszustand bekannt ist. Dann enthält die Menge

$$\mathcal{Z}(0 \mid -1) = \{1, 2, 3, 4, 5, 6\},$$

bei der die Beobachtung beginnt, alle Automatenzustände. Aus der Messung des ersten E/A-Paares ( $v(0) = 1, w(0) = 1$ ) erkennt man, dass sich der Automat nur entlang von Kanten bewegt haben kann, die mit 1/1 bewertet sind. Derartige Kanten gibt es zwischen folgenden Zuständen:

$$1 \xrightarrow{1/1} 1, \quad 3 \xrightarrow{1/1} 2, \quad 3 \xrightarrow{1/1} 3, \quad 5 \xrightarrow{1/1} 6$$

(siehe Hervorhebungen in Abb. 15.2). Auf Grund des ersten Messergebnisses ist also bekannt, dass sich der Automat zum Zeitpunkt  $k = 0$  nur in den Zuständen 1, 3 und 5 befunden haben kann

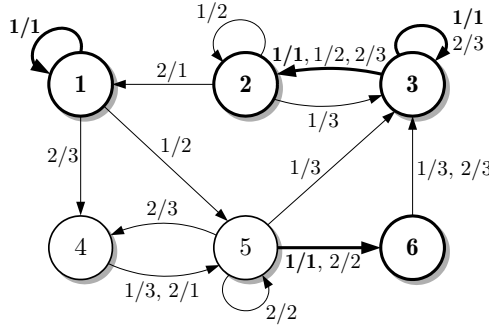
$$\mathcal{Z}(0 \mid 0) = \{1, 3, 5\}$$

und dass er sich nach dem Zustandsübergang zum Zeitpunkt  $k = 1$  in einem der Zustände 1, 2, 3 und 6 aufhält

$$\mathcal{Z}(1 \mid 0) = \{1, 2, 3, 6\},$$

die in Abb. 15.2 hervorgehoben sind.

Offensichtlich enthält die Menge  $\mathcal{Z}(0 \mid 0)$ , die man durch Auswertung der zum Zeitpunkt  $k = 0$  gehörenden Messwerte  $v(0)$  und  $w(0)$  erhält, weniger Zustände  $z(0)$  als die aus den Vorinformationen gebildete Menge  $\mathcal{Z}(0 \mid -1)$ . Die Menge  $\mathcal{Z}(1 \mid 0)$  beschreibt die nach der Auswertung der Messungen zum Zeitpunkt  $k = 0$  für den Nachfolgezeitpunkt  $k = 1$  in Frage kommenden Zustände  $z(1)$ .  $\square$



**Abb. 15.2:** Automatengraph mit Kennzeichnung der Zustandsmenge  $\mathcal{Z}(1|0)$

Dieses Vorgehen kann man rekursiv für steigenden Zeithorizont  $k_e = 0, 1, 2, \dots$  anwenden, wobei man nach jeder neuen Messung  $(v(k_e), w(k_e))$  nur noch diejenigen Zustandsübergänge betrachtet, die aus einem Zustand der vorher bestimmten Menge  $\mathcal{Z}(k_e | k_e - 1)$  möglich sind. Vor der Messung weiß man, dass

$$z(k_e) \in \mathcal{Z}(k_e | k_e - 1)$$

gilt. Nach der Messung  $(v(k_e), w(k_e))$  untersucht man für alle diese Zustände anhand der Verhaltensrelation  $L$  des nichtdeterministischen Automaten, ob es für die Eingabe  $v(k_e)$  einen Zustandsübergang zu irgendeinem Nachfolgezustand gibt, bei dem die Ausgabe  $w(k_e)$  erzeugt wird. Nur wenn dies möglich ist, gehört der Zustand  $z(k_e)$  auch zu der neu zu bestimmen Menge  $\mathcal{Z}(k_e | k_e)$  und die bei diesen Zustandsübergängen erreichbaren Nachfolgezustände  $z(k_e + 1)$  zur Menge  $\mathcal{Z}(k_e + 1 | k_e)$ .

Da nach jeder Erhöhung des Zeithorizonts  $k_e$  dieselben Berechnungsschritte ausgeführt werden, wird mit einheitlichen Bezeichnungen gearbeitet. Mit  $\bar{v}$  und  $\bar{w}$  werden die gemessenen Werte der Eingabe und Ausgabe bezeichnet, wobei der Querstrich kennzeichnet, dass es sich um bekannte Größen handelt. Der Zustand zum Zeitpunkt  $k_e$  wird durch  $z$ , ein möglicher Nachfolgezustand durch  $z'$  symbolisiert. Der Zustandsübergang

$$z \xrightarrow{\bar{v}/\bar{w}} z' \quad (15.1)$$

ist möglich, wenn das Tupel  $(z', \bar{w}, z, \bar{v})$  zur Verhaltensrelation  $L$  gehört:

$$L(z', \bar{w}, z, \bar{v}) = 1.$$

Nach der Messung des aktuellen E/A-Paares  $(v(k_e) = \bar{v}, w(k_e) = \bar{w})$  führt der Zustandsbeobachter also die folgenden zwei Schritte aus. Erstens überprüft er, welche Zustände  $z \in \mathcal{Z}(k_e | k_e - 1)$ , die entsprechend der Kenntnis der E/A-Folgen bis zum Zeitpunkt  $k_e - 1$  möglich waren, auch bei Kenntnis des nächsten E/A-Paares  $(v(k_e) = \bar{v}, w(k_e) = \bar{w})$  noch mögliche Automatenzustände zum Zeitpunkt  $k_e$  sind, woraus sich die Menge

$$\mathcal{Z}(k_e | k_e) = \{z \in \mathcal{Z}(k_e | k_e - 1) \mid \exists z' : L(z', \bar{w}, z, \bar{v}) = 1\} \quad (15.2)$$

ergibt. Diese Menge enthält i. Allg. nicht alle Elemente von  $\mathcal{Z}(k_e | k_e - 1)$ , so dass die Beziehung

$$\mathcal{Z}(k_e | k_e) \subseteq \mathcal{Z}(k_e | k_e - 1)$$

gilt, die deutlich macht, dass durch die Auswertung einer Messung die Kenntnis über den aktuellen Systemzustand verbessert wird.

Zweitens macht der Beobachter eine Vorhersage der Automatenzustände zum Zeitpunkt  $k_e + 1$ , indem er bestimmt, in welche Nachfolgezustände  $z'$  der Automat von den Zustände  $z \in \mathcal{Z}(k_e | k_e)$  aus mit einem Zustandsübergang (15.1) gelangen kann:

$$\mathcal{Z}(k_e + 1 | k_e) = \{z' | \exists z \in \mathcal{Z}(k_e | k_e) : L(z', \bar{w}, z, \bar{v}) = 1\}. \quad (15.3)$$

Ob diese Menge mehr oder weniger Zustände umfasst als die zuvor berechnete Menge  $\mathcal{Z}(k_e | k_e)$ , hängt von den Eigenschaften des Automaten und dem aktuellen E/A-Paar ab.

Ist nichts über den Anfangszustand bekannt, so beginnt die Zustandsbeobachtung bei

$$\mathcal{Z}(0 | -1) = \mathcal{Z}, \quad (15.4)$$

womit ausgedrückt wird, dass sich der Automat in einem beliebigen Zustand seiner Zustandsmenge  $\mathcal{Z}$  befindet. Ist etwas über den Anfangszustand bekannt, so kann diese Information in die Festlegung der Menge  $\mathcal{Z}(0 | -1)$  einfließen.

Der Beobachtungsalgorithmus kann folgendermaßen zusammengefasst werden, wobei die Abkürzungen

$$\begin{aligned} \mathcal{Z} &= \mathcal{Z}(k_e | k_e) \\ \mathcal{Z}' &= \mathcal{Z}(k_e + 1 | k_e) \end{aligned}$$

verwendet werden.

---

#### Algorithmus 15.1 Zustandsbeobachtung nichtdeterministischer Automaten

---

**Gegeben:** Nichtdeterministischer Automat mit der Verhaltensrelation  $L(z', w, z, v)$   
Menge möglicher Anfangszustände  $\mathcal{Z}(0 | -1)$ .

Initialisierung:  $\mathcal{Z}' = \mathcal{Z}(0 | -1)$   
 $k_e = 0$

Schleife: 1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .

2. Bestimmen Sie die Mengen

$$\mathcal{Z} := \{z \in \mathcal{Z}' | \exists z' : L(z', \bar{w}, z, \bar{v}) = 1\}$$

$$\mathcal{Z}' := \{z' | \exists z \in \mathcal{Z} : L(z', \bar{w}, z, \bar{v}) = 1\}.$$

3.  $k_e := k_e + 1$

Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $\mathcal{Z}(k_e | k_e) := \mathcal{Z}$  für  $k_e = 0, 1, \dots$

---

**Beispiel 15.1 (Forts.)** Zustandsbeobachtung eines nichtdeterministischen Automaten

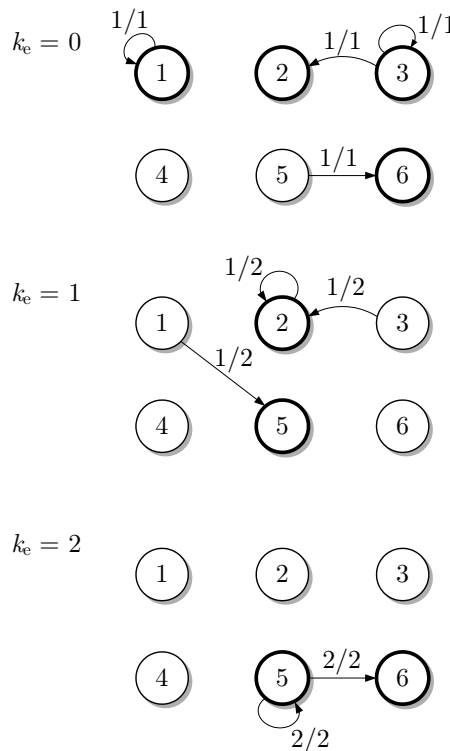
Die Beobachtungsaufgabe soll für den in Abb. 15.1 gezeigten Automaten für das E/A-Paar

$$(V, W) = ((1, 1, 2, 2), (1, 2, 2, 3)) \quad (15.5)$$

gelöst werden. Da keine Vorinformationen über den Zustand vorhanden sind, wird der Algorithmus 15.1 mit

$$\mathcal{Z}' = \mathcal{Z}(0 \mid -1) = \{1, 2, 3, 4, 5, 6\}$$

initialisiert. Durch Auswertung der nach und nach gemessenen E/A-Paare  $(\bar{v}, \bar{w})$  erhält man die folgenden Beobachtungsergebnisse:



**Abb. 15.3:** Ergebnis der Zustandsbeobachtung für das E/A-Paar (15.5)

1.  $k_e = 0$ :

Es wird das E/A-Paar  $(v(0) = 1, w(0) = 1)$  gemessen, womit man folgende Mengen erhält:

$$\mathcal{Z} = \mathcal{Z}(0 \mid 0) = \{1, 3, 5\}$$

$$\mathcal{Z}' = \mathcal{Z}(1 \mid 0) = \{1, 2, 3, 6\}.$$

Das Ergebnis ist in der oberen Zeile von Abb. 15.3 angegeben, wo der Automatengraph auf die für das E/A-Paar zutreffenden Kanten reduziert wurde. Die Menge  $\mathcal{Z}(0 \mid 0)$  ist die Menge von Zuständen, von denen diese Kanten ausgehen. Die zur Menge  $\mathcal{Z}(1 \mid 0)$  gehörenden Zustände, zu denen



diese Kanten führen, sind in der Abbildung hervorgehoben. In einem dieser Zustände befindet sich der nichtdeterministische Automat zum Zeitpunkt 1.

2.  $k_e = 1$ :

Es wird das E/A-Paar  $(v(1) = 1, w(1) = 2)$  gemessen, womit man folgende Mengen erhält:

$$\mathcal{Z} = \mathcal{Z}(1 | 1) = \{1, 2, 3\}$$

$$\mathcal{Z}' = \mathcal{Z}(2 | 1) = \{2, 5\}.$$

Das Ergebnis ist in der zweiten Zeile von Abb. 15.3 veranschaulicht. Durch die neue Messung kann die vierelementige Menge  $\mathcal{Z}(1 | 0)$  auf die dreielementige Menge  $\mathcal{Z}(1 | 1)$  reduziert werden. Nach dem nächsten Zustandsübergang ist der Automat in einem der beiden in der Abbildung hervorgehobenen Zustände.

3.  $k_e = 2$ :

Es wird das E/A-Paar  $(v(2) = 2, w(2) = 2)$  gemessen, womit man folgende Mengen erhält:

$$\mathcal{Z} = \mathcal{Z}(2 | 2) = \{5\}$$

$$\mathcal{Z}' = \mathcal{Z}(3 | 2) = \{5, 6\}.$$

Wie man in der dritten Zeile von Abb. 15.3 sieht, kennt man auf Grund der neuen Messung den Zustand  $z(2) = 5$  exakt. Allerdings enthält die Menge  $\mathcal{Z}(3 | 2)$  wieder zwei Elemente, weil auf Grund des nichtdeterministischen Verhaltens der Automat für das gemessene E/A-Paar vom Zustand 5 in die zwei möglichen Nachfolgezustände 5 oder 6 übergehen kann.

4.  $k_e = 3$ :

Es wird das E/A-Paar  $(v(3) = 2, w(3) = 3)$  gemessen, womit man folgende Mengen erhält:

$$\mathcal{Z} = \mathcal{Z}(3 | 3) = \{5, 6\}$$

$$\mathcal{Z}' = \mathcal{Z}(4 | 3) = \{4, 3\}.$$

Die zusätzliche Messinformation bringt keine Verbesserung der Kenntnisse über den Zustand  $z(3)$ . Als Nachfolgezustand  $z(4)$  kommen wieder zwei Zustände in Frage.

**Diskussion.** Das Beispiel zeigt, dass sich die Zahl der möglichen Zustände während der Beobachtung ändert. Wenn man mit der gesamten Zustandsmenge als mögliche Initialzustände beginnt, reduziert sich die Mächtigkeit der betrachteten Zustandsmengen nach den ersten Messungen häufig sehr stark, aber es ist charakteristisch für nichtdeterministische Automaten, dass die Menge in vielen Anwendungsfällen nicht auf ein Element schrumpft, was eine exakte Kenntnis des Automatenzustands bedeuten würde.  $\square$

<b>Aufgabe 15.1*</b> Zustandsbeobachtung eines nichtdeterministischen Automaten
---

Lösen Sie die Beobachtungsaufgabe für den in Abb. 15.1 gezeigten Automaten für das E/A-Paar

$$(V, W) = ((2, 2, 1, 2), (1, 2, 3, 3)). \quad \square \tag{15.6}$$

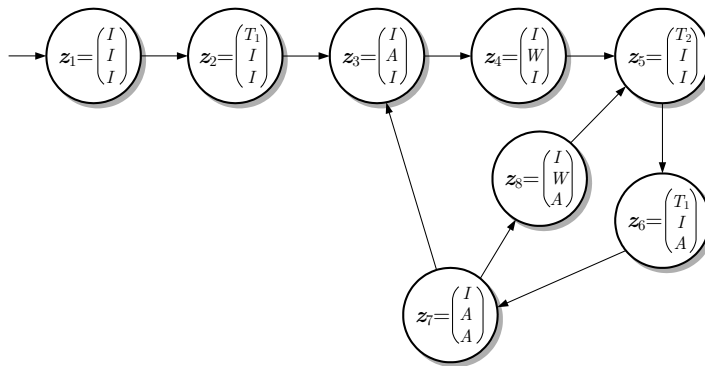
**Aufgabe 15.2\*** Zustandsbeobachtung einer Fertigungszelle

Es wird die im Beispiel 1.10 auf S. 38 beschriebene Fertigungszelle vom Standpunkt der Überwachungsebene betrachtet. Auf dieser Ebene stehen nicht alle Informationen zur Verfügung, die zwischen den Komponenten der Fertigungszelle und ihren dezentralen Steuereinrichtungen ausgetauscht werden, sondern nur der für die Überwachung der Gesamtanlage notwendige Teil. Als Beispiel wird im Folgenden angenommen, dass auf der Überwachungsebene nur die Meldungen der Maschinen  $M_1$  und  $M_2$  verfügbar sind, dass diese Maschinen die Bearbeitung eines Teiles abgeschlossen haben:

$w = 1$  – Die Maschine  $M_1$  hat die Bearbeitung eines Teiles abgeschlossen.

$w = 2$  – Die Maschine  $M_2$  hat die Bearbeitung eines Teiles abgeschlossen.

Deshalb gibt es Zustandsübergänge in der Fertigungszelle, die von der Überwachungsebene nicht bemerkt werden, weil bei ihnen keine Ausgabe erzeugt wird. Darauf muss man bei der Überwachung achten.



**Abb. 15.4:** Nichtdeterministischer Automat, der die Fertigungszelle beschreibt

Die Fertigungszelle wird durch den in Abb. 15.4 gezeigten Automaten beschrieben. Da sie einschließlich ihrer Steuerung betrachtet wird, ist sie ein autonomes System. Es soll untersucht werden, ob mit den beschriebenen Meldungen der aktuelle Zustand der Fertigungszelle beobachtet werden kann.

1. Bei welchen Zustandsübergängen werden die o. g. Ausgaben erzeugt? Markieren Sie diese Zustandsübergänge im Automatengraphen.
2. Zustandsübergänge, bei denen keine Ausgaben erzeugt werden, werden von der Prozessüberwachung nicht erkannt. Sie können deshalb aus dem Automaten gestrichen werden. Welcher Automat entsteht dadurch?
3. Die Fertigungszelle hat folgende Ausgabefolgen erzeugt:

(a)  $W = (1, 2)$

(b)  $W = (1, 1)$

In welchem Zustand befindet sich die Fertigungszelle danach?

□

### 15.2.2 Darstellung des Beobachtungsalgorithmus als Automat

Eine interessante Darstellung des Beobachtungsalgorithmus 15.1 führt auf einen deterministischen Automaten, dessen Zustände das dargestellte Beobachtungsergebnis  $\mathcal{Z}(k_e | k_e - 1)$  beschreiben. Abbildung 15.5 zeigt den Automaten, der den Beobachtungsalgorithmus für den Automaten aus Abb. 15.1 repräsentiert.

Die Beobachtung beginnt in dem Zustand, der die Menge  $\mathcal{Z}(0 | -1)$  darstellt, also beispielsweise im Zustand  $\{1, 2, 3, 4, 5, 6\}$ . Die Auswertung des E/A-Paars  $(\bar{v}, \bar{w})$  durch den Beobachtungsalgorithmus 15.1 bedeutet im Automatengraphen in Abb. 15.5 die Bewegung vom aktuellen Zustand entlang der mit  $\bar{v}/\bar{w}$  beschrifteten Kante und der neue Zustand repräsentiert das dabei erhaltene Beobachtungsergebnis. Im Beispiel ist dies für das E/A-Paar  $(\bar{v} = 1, \bar{w} = 1)$  die Menge

$$\mathcal{Z}(1 | 0) = \{1, 2, 3, 6\}$$

und für das E/A-Paar  $(\bar{v} = 2, \bar{w} = 1)$  die Menge

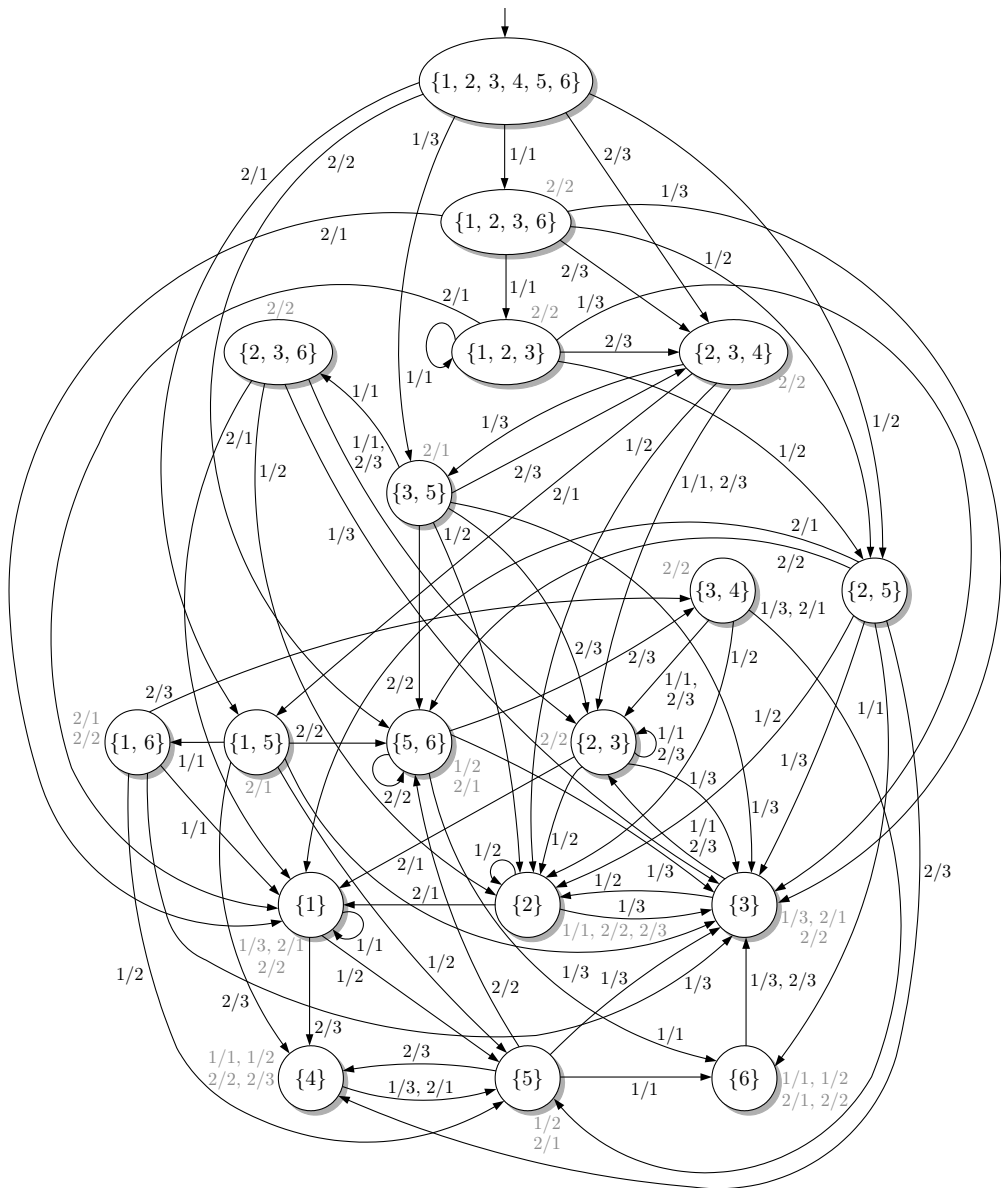
$$\mathcal{Z}(1 | 0) = \{1, 5\}.$$

Der Automatengraph ist so gezeichnet, dass für die weiter oben stehenden Knoten das Beobachtungsergebnis aus Mengen mit mehreren Automatenzuständen besteht, während im unteren Teil einelementige Mengen stehen. Dort ist ein Ausschnitt aus dem Automatengraphen aus Abb. 15.1 zu sehen. Die Knoten mit einelementigen Mengen sind durch die Kanten des Graphen aus Abb. 15.1 verbunden, die eindeutig durch das E/A-Paar gekennzeichnet sind. Für die anderen Übergänge „vergrößert“ sich das Beobachtungsergebnis in dem Sinne, dass das bisherige Beobachtungsergebnis mit einem Zustand auf ein neues Beobachtungsergebnis mit einer Menge mit mehreren Zuständen führt.

Beispielsweise führt vom Knoten 5 eine Kante zum Knoten  $\{5, 6\}$ , der mit dem E/A-Paar  $(\bar{v} = 2, \bar{w} = 2)$  markiert ist. Das heißt, wenn das Zwischenergebnis zum Zeitpunkt  $k_e - 1$  den aktuellen Zustand des Automaten eindeutig mit dem Zustand 5 auswies und im nächsten Schritt für die Eingabe  $\bar{v} = 2$  die Ausgabe  $\bar{w} = 2$  gemessen wurde, so ist auf Grund des nichtdeterministischen Verhaltens des Automaten zum Zeitpunkt  $k_e$  keine eindeutige Aussage über den aktuellen Automatenzustand möglich, sondern der Beobachtungsalgorithmus erkennt, dass der Automat jetzt in einem der beiden Zustände 5 oder 6 ist.

Der Vollständigkeit halber sei hier die im Abschn. 15.2.4 noch ausführlich behandelte Situation erwähnt, dass es vorkommen kann, dass der Beobachtungsalgorithmus bei bestimmten E/A-Paaren eine leere Menge möglicher Zustände erzeugt. Diese E/A-Paare werden als *inkonsistent* zu dem gegebenen Automaten bezeichnet. Für sie gibt es keine Kante vom aktuellen Zustand des Beobachtungsalgorithmus ausgehende Kante und sie sind deshalb direkt an den entsprechenden Zustandsknoten geschrieben. Ihrer Bedeutung entsprechend könnte man für den Beobachtungsalgorithmus auch den Zustand  $\emptyset$  einführen und alle Zustände für die inkonsistenten E/A-Paare mit diesem Zustand verbinden.

Die Abbildung zeigt, dass der Algorithmus 15.1 zusammen mit dem Modell des zu beobachtenden Systems wie ein deterministischer Automat arbeitet. Allerdings ist die Komplexität dieses für das Beispiel in Abb. 15.5 gezeigten Automaten viel größer als die Komplexität des zu beobachtenden Systems. Es ist deshalb wichtig zu erkennen, dass die Arbeitsweise des



**Abb. 15.5:** Darstellung des Beobachtungsalgorithmus für den Automaten nach Abb. 15.1 als deterministischer Automat

Beobachtungsalgorithmus der eines deterministischen Automaten gleicht, dass die Darstellung dieses Automaten durch den Algorithmus 15.1 zusammen mit dem Modell des zu beobachtenden Systems jedoch eine wesentlich kleinere Komplexität aufweist.

### 15.2.3 Anwendung des Beobachtungsalgorithmus auf deterministische Automaten

Der für nichtdeterministische Automaten entwickelte Beobachtungsalgorithmus kann direkt auf deterministische Automaten angewendet werden, wenn aus deren Beschreibung (11.12)

$$\mathcal{A} : \begin{cases} z(k+1) = G(z(k), v(k)), & z(0) = z_0 \\ w(k) = H(z(k), v(k)) \end{cases}$$

entsprechend Gl. (11.34) auf S. 369 die Verhaltensrelation

$$L_d : \mathcal{Z} \times \mathcal{W} \times \mathcal{Z} \times \mathcal{V} \rightarrow \{0, 1\}$$

bestimmt wird:

$$L_d(z(k+1), w(k), z(k), v(k)) = 1 \iff \begin{aligned} z(k+1) &= G(z(k), v(k)) \\ w(k) &= H(z(k), v(k)). \end{aligned}$$

Die Verhaltensrelationen  $L_d$  deterministischer Automaten unterscheiden sich von den Verhaltensrelationen  $L$  nichtdeterministischer Automaten dadurch, dass es zu jedem Zustands-Eingabe-Paar  $(z(k), v(k))$  genau ein Zustands-Ausgabe-Paar  $(z(k+1), w(k))$  gibt, so dass die Verhaltensrelation für das aus beiden Paaren gebildeten Tupel  $(z(k+1), w(k), z(k), v(k))$  den Wert eins besitzt. Bei nichtdeterministischen Automaten gibt es i. Allg. mehrere solche Paare.

Die Aufgabe der Zustandsbeobachtung deterministischer Automaten kann mit dem Algorithmus 15.1 gelöst werden, wenn man ihn auf die Verhaltensrelation  $L_d$  anwendet.

**Schrittweise Verbesserung des Beobachtungsergebnisses.** Die Eigenschaft des deterministischen Automaten, dass es zu jedem Zustand genau einen Nachfolgezustand gibt, erleichtert die Beobachtung. Auf Grund dieser Eigenschaft kann es nämlich nicht vorkommen, dass sich die Menge möglicher Zustände durch die Bewegung des Automaten vergrößert. Wenn man insbesondere die mögliche Zustandsmenge auf einen einzigen Zustand eingeschränkt hat, so kann man die zukünftige Bewegung des Automaten exakt verfolgen. Dafür muss dann nur noch die Eingangsfolge gemessen und so verarbeitet werden, wie es zur Lösung von Vorhersageaufgaben mit exakt bekanntem Anfangszustand gezeigt wurde.

Im Allgemeinen besitzt das Beobachtungsergebnis bei deterministischen Automaten folgende Monotonieeigenschaft bezüglich der Mächtigkeit der Ergebnismengen:

$$1 \leq |\mathcal{Z}(k_e + 1 | k_e + 1)| \leq |\mathcal{Z}(k_e | k_e)| \leq N. \quad (15.7)$$

Das heißt, dass die Anzahl der möglichen Zustände im Beobachtungsergebnis bei steigendem Zeithorizont  $k_e$  nur abnehmen kann. Die in der länger werdenden Messwertfolge enthaltenen

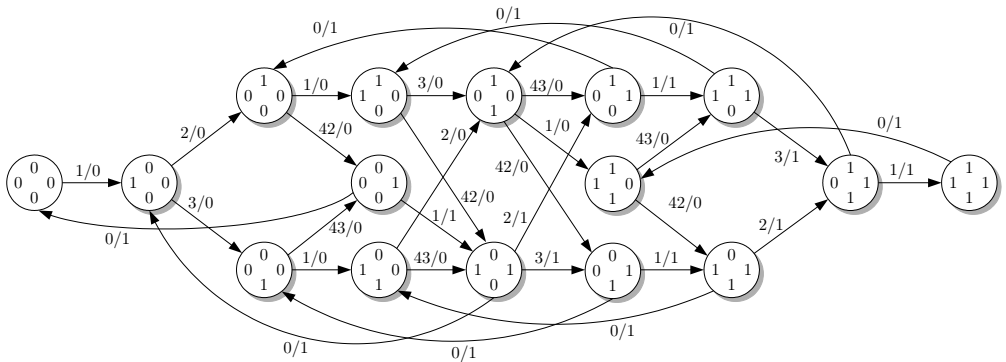
Informationen führen also zu einem immer besseren Beobachtungsergebnis. Es ist jedoch nicht gesichert, dass die Menge der möglichen Zustände auf einen einzigen Zustand „zusammenschrumpft“.

**Aufgabe 15.3** Zustandsbeobachtung von vier Werkzeugmaschinen

Die im Beispiel 11.4 auf S. 363 beschriebenen vier Werkzeugmaschinen werden von einer Automatisierungseinrichtung so gesteuert, dass der gewünschte Materialfluss von der Maschine  $M_1$  über eine der Maschinen  $M_2$  oder  $M_3$  zur Maschine  $M_4$  erfolgt. Die dafür von der Automatisierungseinrichtung erzeugten Steuereingriffe  $v$  sind folgendermaßen bezeichnet:

$v$	Bedeutung
0	Es wird ein Werkstück von $M_4$ abtransportiert.
1	Es wird ein Werkstück zu $M_1$ transportiert.
2	Es wird ein Werkstück von $M_1$ zu $M_2$ transportiert.
3	Es wird ein Werkstück von $M_1$ zu $M_3$ transportiert.
42	Es wird ein Werkstück von $M_2$ zu $M_4$ transportiert.
43	Es wird ein Werkstück von $M_3$ zu $M_4$ transportiert.

Aus der Sicht der Überwachungsebene erfolgt der Werkzeugtransport so schnell, dass er als Ereignis aufgefasst wird. Das System kann deshalb durch den in Abb. 15.6 gezeigten deterministischen Automaten beschrieben werden.



**Abb. 15.6:** Beschreibung der Arbeitsweise der vier gesteuerten Werkzeugmaschinen

Für die Zustandsbeobachtung steht außer dem Eingriff der Steuerung auch der Zustand der Maschine  $M_4$  als Messinformation zur Verfügung:  $w(k) = z_4(k)$ . Da diese Ausgabe nicht vom Eingang abhängt, ist der Automat ein Moore-Automat.

Lösen Sie die Beobachtungsaufgabe für folgende E/A-Paare:

$$V(0...3) = (1, 3, 1, 2), \quad W(0...3) = (1, 1, 1, 1)$$

$$V(0...3) = (1, 3, 1, 0), \quad W(0...3) = (1, 1, 1, 1).$$

Suchen Sie nach E/A-Paaren, bei denen bei möglichst kleinem bzw. möglichst großem Zeithorizont der Zustand der Werkzeugmaschinen eindeutig bekannt ist. Interpretieren Sie diese Ergebnisse bezüglich der Unsicherheiten, mit denen auf der Überwachungsebene die Arbeitsweise der Werkzeugmaschinen verfolgt werden kann.  $\square$

#### 15.2.4 Konsistenz von E/A-Paaren mit nichtdeterministischen Automaten

In diesem Abschnitt wird die Frage untersucht, unter welchen Umständen bei der Anwendung des Beobachtungsalgorithmus die berechneten Zustandsmengen leer sind. Offenbar gibt es dann keine Kante, die von einem der zum Zeitpunkt  $k_e$  für den Systemzustand in Frage kommenden Zustandsknoten ausgeht und mit dem gemessenen E/A-Paar  $(v(k_e), w(k_e))$  bewertet ist. Man sagt dann, dass das E/A-Paar  $(V, W)$  nicht mit dem Automaten *konsistent* ist.

**Definition 15.1 (Konsistenz von E/A-Paaren mit nichtdeterministischen Automaten)**

Das E/A-Paar  $(V(0...k_e), W(0...k_e))$  heißt *konsistent* mit dem nichtdeterministischen Automaten  $\mathcal{N} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, L, \mathcal{Z}_0)$ , wenn eine Zustandsfolge

$$Z(0...k_e + 1) = (z(0), z(1), \dots, z(k_e + 1))$$

existiert, mit der die folgenden Beziehungen gelten:

$$z(0) \in \mathcal{Z}_0 \tag{15.8}$$

$$L(z(k+1), w(k), z(k), v(k)) = 1, \quad k = 0, 1, \dots, k_e. \tag{15.9}$$

Andernfalls nennt man das E/A-Paar zum Automaten  $\mathcal{N}$  inkonsistent.

Die Konsistenz besagt, dass der Automat ausgehend von einem Zustand der Menge  $\mathcal{Z}(0|-1)$  für die gegebene Eingangsfolge  $V$  eine Zustandsfolge  $Z$  erzeugen kann, für die er die Ausgangsfolge  $W$  generieren kann. Die gemessene E/A-Folge „passt“ also zu dem betrachteten Automaten.

Es sei betont, dass sich die Konsistenz i. Allg. auf die Folgen  $V$  und  $W$  und nicht auf einzelne Messwertpaare bezieht, denn es kann im Automatengraphen durchaus Kanten geben, die mit dem letzten gemessenen Paar  $(v(k_e), w(k_e))$  bewertet sind, obwohl die gesamten Folgen  $(V, W)$  inkonsistent sind. Die Menge der möglichen Zustände ist dann durch die Verarbeitung der Messungen für  $k = 0, 1, \dots, k_e - 1$  soweit eingeschränkt, dass die mit  $(v(k_e), w(k_e))$  beschrifteten Kanten bei der Beobachtung keine Rolle mehr spielen.

Eine Inkonsistenz kann zwei Gründe haben:

- Der Automat hatte einen Anfangszustand, der nicht zur Menge  $\mathcal{Z}(0|-1)$  gehört. Hier beruht die Inkonsistenz auf falschen Vorinformationen über den Automatenzustand und Gl. (15.8) ist verletzt. Man kann diese Inkonsistenz bei der Wahl der Menge  $\mathcal{Z}(0|-1)$  dadurch vermeiden, dass man entsprechend Gl. (15.4) alle Automatenzustände als möglichen Anfangszustand betrachtet.

- Die sich in den E/A-Folgen äußernde Dynamik des Systems stimmt nicht mit der durch die Verhaltensrelation  $L$  beschriebenen Dynamik überein, so dass Gl. (15.9) für keine Zustandsfolge erfüllt werden kann. Hier zeigt die Inkonsistenz der E/A-Folge mit dem Automaten, dass der Automat das gegebene System fehlerhaft beschreibt.

Die zweite Tatsache kann für die Prozessdiagnose genutzt werden. Man überprüft, ob die gemessene E/A-Folgen mit dem Automaten für den fehlerfreien Betrieb der Anlage konsistent sind. Wird eine Inkonsistenz festgestellt, so ist offensichtlich, dass sich die Anlage nicht so wie im fehlerfreien Fall verhält und offensichtlich ein Fehler aufgetreten ist (Abschn. 16.2).

Die Konsistenzprüfung kann mit dem Algorithmus 15.1 durchgeführt werden, wenn man mit der in Gl. (15.4) angegebenen Initialisierung arbeitet. Nach Schritt 2 der Iterationsschleife untersucht man in einem zusätzlichen Schritt, ob die berechneten Menge  $\mathcal{Z}$  leer sind. Ist dies der Fall, ist die Inkonsistenz erkannt und der Algorithmus wird beendet.

### Beispiel 15.2 Inkonsistenz eines E/A-Paares mit einem Automaten

Für den in Abb. 15.1 gezeigten Automaten wird das E/A-Paar

$$(V, W) = ((1, 1, 2), (3, 1, 2)) \quad (15.10)$$

gemessen. Dafür liefert der mit

$$\mathcal{Z} = \mathcal{Z}(0 \mid -1) = \{1, 2, 3, 4, 5, 6\}$$

begonnene Beobachtungsalgorithmus folgende Ergebnisse (Abb. 15.7):

1.  $k_e = 0$ :

Für das E/A-Paar  $(v(0) = 1, w(0) = 3)$  erhält man folgende Mengen:

$$\mathcal{Z} = \mathcal{Z}(0 \mid 0) = \{2, 4, 5, 6\}$$

$$\mathcal{Z}' = \mathcal{Z}(1 \mid 0) = \{3, 5\}.$$

Das erste Messergebnis  $(1, 3)$  ist also mit dem Automaten konsistent.

2.  $k_e = 1$ :

Für das E/A-Paar  $(v(1) = 1, w(1) = 1)$  erhält man folgende Mengen:

$$\mathcal{Z} = \mathcal{Z}(1 \mid 1) = \{3, 5\}$$

$$\mathcal{Z}' = \mathcal{Z}(2 \mid 1) = \{2, 3, 6\}.$$

Auch das E/A-Paar bis zum zweiten Element ist mit dem Automaten konsistent.

3.  $k_e = 2$ :

Es wird das E/A-Paar  $(v(2) = 2, w(2) = 2)$  gemessen. Da keine von den Zuständen 2, 3, und 6 abgehende Kante die Beschriftung 2/2 besitzt, erzeugt der Automat unter dem Einfluss der Eingabe 2 bei keiner in den Zuständen 2, 3 und 6 beginnenden Bewegung die Ausgabe 2. Folglich sind die gegebenen E/A-Folgen mit dem Automaten inkonsistent.

Das Ergebnis besagt, dass unabhängig davon, in welchem Zustand der Automat seine Bewegung unter dem Einfluss der Eingangsfolge  $V = (1, 1, 2)$  beginnt und welchem von gegebenenfalls mehreren Zustandsübergängen er folgt, der Automat die Ausgangsfolge  $W = (3, 1, 2)$  nicht erzeugen kann.  $\square$



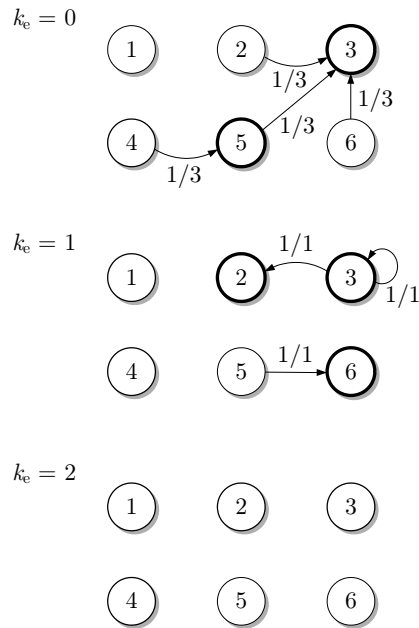


Abb. 15.7: Ergebnis der Konsistenzprüfung für das E/A-Paar (15.10)

### 15.3 Beobachtung stochastischer Automaten

**Beobachtungsaufgabe.** Die Beobachtung stochastischer Automaten (11.49)

$$\mathcal{S} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, L, p_0(z))$$

führt nicht nur auf eine Menge  $\mathcal{Z}(k_e | k_e)$  von möglichen Zuständen, sondern zusätzlich dazu auf eine Wahrscheinlichkeitsverteilung für diese Zustände. Die bedingte Wahrscheinlichkeit

$$\text{Prob}(Z(k_e) = z \mid V(0) = v_0, V(1) = v_1, \dots, V(k_e) = v_k, \\ W(0) = w_0, W(1) = w_1, \dots, W(k_e) = w_k)$$

gibt die Wahrscheinlichkeit dafür an, dass der Automat den Zustand  $z$  zum Zeitpunkt  $k_e$  einnimmt unter der Bedingung, dass er für die Eingangsfolge

$$V(0 \dots k_e) = (v_0, v_1, \dots, v_k)$$

die Ausgangsfolge

$$W(0 \dots k_e) = (w_0, w_1, \dots, w_k)$$

erzeugt hat. Diese Wahrscheinlichkeit wird kürzer mit  $p(z, k_e)$  bezeichnet. Die Funktion  $p(z, k_e)$  wird für jeden Zeitpunkt  $k_e$  für alle  $z \in \mathcal{Z}$  berechnet. Der Wert  $k_e$  gibt den Zeithorizont an, für den das E/A-Paar bekannt ist. Im Folgenden wird außerdem die Wahrscheinlichkeitsverteilung

$$p'(z', k_e) = \text{Prob}(Z(k_e + 1) = z' \mid V(0) = v_0, V(1) = v_1, \dots, V(k_e) = v_k, \\ W(0) = w_0, W(1) = w_1, \dots, W(k_e) = w_k)$$

gebraucht, die ebenfalls aus dem E/A-Paar bis zum Zeithorizont  $k_e$  berechnet wird. Der Apostroph an der Funktion  $p'$  dient wieder zur Kennzeichnung der Wahrscheinlichkeitsverteilung für den Nachfolgezustand.

**Beobachtungsalgorithmus.** Die Lösung der Beobachtungsaufgabe erhält man aus einer längeren Umformung unter Nutzung wahrscheinlichkeitstheoretischer Gesetze. Diese Umformung soll hier nicht im Einzelnen vorgeführt werden, weil das Ergebnis vor allem unter dem Gesichtspunkt diskutiert werden soll, welche zusätzlichen Informationen die wahrscheinlichkeitstheoretische Beschreibung der Zustandsübergänge für die Lösung der Beobachtungsaufgabe im Vergleich zum nichtdeterministischen Automaten bringt.

Die Beobachtung beginnt bei  $k_e = 0$  mit der A-priori-Wahrscheinlichkeit (Wahrscheinlichkeit vor der ersten Messung) für den Anfangszustand

$$p'(z, -1) := p_0(z), \quad z \in \mathcal{Z}. \quad (15.11)$$

Nach dem ersten E/A-Paar ( $V(0) = \bar{v}, W(0) = \bar{w}$ ) kann diese Wahrscheinlichkeit bereits korrigiert werden, so wie beim nichtdeterministischen Automaten die Menge  $\mathcal{Z}(0 \mid -1)$  auf die Menge  $\mathcal{Z}(0 \mid 0)$  reduziert wurde. Man erhält folgende A-posteriori-Wahrscheinlichkeitsverteilung (Wahrscheinlichkeit nach der Auswertung einer Messung):

$$p(z, 0) = \frac{\sum_{z'} L(z', \bar{w} \mid z, \bar{v}) \cdot p'(z, -1)}{\sum_{z, z'} L(z', \bar{w} \mid z, \bar{v}) \cdot p'(z, -1)}, \quad z \in \mathcal{Z}. \quad (15.12)$$

Im Zähler werden die Wahrscheinlichkeiten dafür addiert, dass der Automat im Zustand  $Z(0) = z$  ist und unter der Wirkung der Eingabe  $V(0) = \bar{v}$  in einen Nachfolgezustand  $z'$  übergeht und dabei die Ausgabe  $W(0) = \bar{w}$  erzeugt. Der Nenner gibt die Wahrscheinlichkeit dafür an, dass der Automat irgendeine Bewegung zwischen zwei beliebigen Zuständen  $z$  und  $z'$  unter der Wirkung von  $V(0) = \bar{v}$  und bei Erzeugung von  $W(0) = \bar{w}$  ausführt. Diese Summe erfasst im Automatengraphen also alle Kanten, die mit dem E/A-Paar  $(\bar{v}, \bar{w})$  bewertet sind und in einem Knoten  $z$  beginnen, in dem sich der Automat mit einer nicht verschwindenden Wahrscheinlichkeit  $p'(0, -1) > 0$  zur Zeit  $k = 0$  befunden hat. Ist der Nenner gleich null, so ist das E/A-Paar mit dem stochastischen Automaten inkonsistent.

Dieser Sachverhalt ist im linken Teil von Abb. 15.8 veranschaulicht, wobei jetzt  $k = 0$  gilt. Der Automat befindet sich mit der A-priori-Wahrscheinlichkeit  $p'(z, -1)$  im Zustand  $z$ . Für das E/A-Paar  $(\bar{v}, \bar{w})$  kann sich der Automat entlang von Kanten  $z \rightarrow z'$  bewegen, deren Bewertung  $L(z', \bar{w} \mid z, \bar{v})$  positiv ist. Wenn eine solche Bewegung zu einem oder mehreren Nachfolgezuständen  $z' \in \mathcal{Z}$  möglich ist, kann sich der Automat tatsächlich zur Zeit 0 im Zustand  $z$  befinden haben und die entsprechend Gl. (15.12) berechnete Wahrscheinlichkeit  $p(z, 0)$  ist positiv. Ist von  $z$  aus keine solche Bewegung möglich, gilt  $p(z, 0) = 0$ , obwohl der Zustand  $z$  vor der Messung noch eine positive Aufenthaltswahrscheinlichkeit  $p'(z, -1) > 0$  hatte.

Mit dem E/A-Paar ( $V(0) = \bar{v}, W(0) = \bar{w}$ ) wird außerdem die A-posteriori-Wahrscheinlichkeit  $p'(z', 0)$  berechnet, mit der sich der Automat zum Zeitpunkt 1 im Zustand  $z'$  befindet:

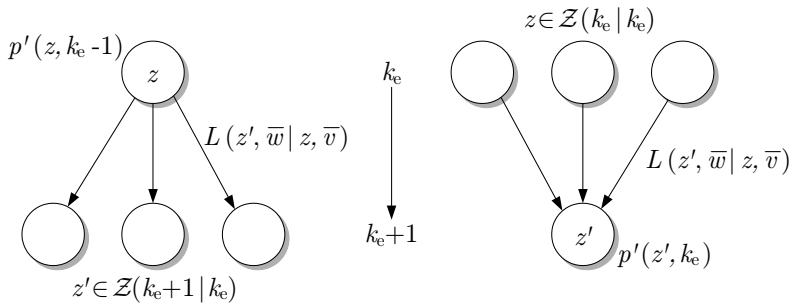


Abb. 15.8: Interpretation der Gl. (15.14)

$$p'(z', 0) = \frac{\sum_z L(z', \bar{w} | z, \bar{v}) \cdot p'(z, -1)}{\sum_{z, z'} L(z', \bar{w} | z, \bar{v}) \cdot p'(z, -1)}, \quad z' \in \mathcal{Z}. \quad (15.13)$$

Diese Gleichung führt für alle Zustände  $z'$ , die der Automat nach dem Durchlaufen der Kanten in Abb. 15.8 annehmen kann, auf einen von null verschiedenen Wert. Die Null als zweites Argument von  $p'(z', 0)$  besagt, dass die E/A-Folge bis zum Zeithorizont  $k_e = 0$  bekannt ist.

Die Beziehung (15.13) ist im rechten Teil von Abb. 15.8 veranschaulicht. Jetzt wird für einen Zustand  $z'$  die Bewegung von allen möglichen Vorgängerzuständen  $z \in \mathcal{Z}$  nach  $z'$  untersucht und über alle diese Zustandsübergänge summiert. Diese Bewegung erfolgt über Kanten, die mit einem positiven Wert von  $L(z', \bar{w} | z, \bar{v})$  beschriftet sind. Es tragen nur diejenigen Zustandsübergänge zum Wert von  $p'(z', 0)$  bei, die in einem Zustand  $z \in \mathcal{Z}_0$  beginnen und für die folglich  $p'(z, -1) > 0$  gilt.

Eine Verallgemeinerung dieser Beziehungen auf einen beliebigen Zeithorizont  $k_e$  ergibt die im Folgenden angegebenen Beziehungen, in denen  $\bar{v}$  und  $\bar{w}$  die zur Zeit  $k_e$  gemessene Eingabe bzw. Ausgabe sind:

Beobachterformel für stochastische Automaten:

$$p(z, k_e) = \frac{\sum_{z'} L(z', \bar{w} | z, \bar{v}) \cdot p'(z, k_e - 1)}{\sum_{z, z'} L(z', \bar{w} | z, \bar{v}) \cdot p'(z, k_e - 1)}, \quad z \in \mathcal{Z} \quad (15.14)$$

$$p'(z', k_e) = \frac{\sum_z L(z', \bar{w} | z, \bar{v}) \cdot p'(z, k_e - 1)}{\sum_{z, z'} L(z', \bar{w} | z, \bar{v}) \cdot p'(z, k_e - 1)}, \quad z' \in \mathcal{Z}.$$

Diese Beziehungen sind nur dann anwendbar, wenn der in beiden Gleichungen auftretende Nenner

$$D(k_e) = \sum_{z, z'} L(z', \bar{w} | z, \bar{v}) \cdot p'(z, k_e - 1) \quad (15.15)$$

ungleich null ist. Verschwindet dieser Nenner, so ist die E/A-Folge mit dem stochastischen Automaten inkonsistent. Die Menge der aktuellen Zustände kann dann nach der Vorschrift

$$\mathcal{Z}(k_e | k_e) = \{z | p(z, k_e) > 0\} \quad (15.16)$$

gebildet werden. Mit diesen Beziehungen kann man den Beobachtungsalgorithmus folgendermaßen formulieren:

---

**Algorithmus 15.2** Zustandsbeobachtung stochastischer Automaten

---

**Gegeben:** Stochastischer Automat mit der Verhaltensrelation  $L(z', w | z, v)$   
A-priori-Wahrscheinlichkeitsverteilung  $p_0(z)$ .

Initialisierung:  $p'(z, -1) := p_0(z)$  für alle  $z \in \mathcal{Z}$   
 $k_e = 0$

Schleife: 1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .  
2. Bestimmen Sie die Wahrscheinlichkeitsverteilungen  
 $p(z, k_e)$  für alle  $z \in \mathcal{Z}$  mit Gl. (15.14)  
 $p'(z', k_e)$  für alle  $z' \in \mathcal{Z}$  mit Gl. (15.14).  
3.  $k_e := k_e + 1$   
Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $p(z, k_e)$  für  $k_e = 0, 1, \dots$  für alle  $z \in \mathcal{Z}$ .

---

Der Algorithmus hat dieselbe Struktur wie der Algorithmus 15.1 für nichtdeterministische Automaten, nur dass jetzt mit den Wahrscheinlichkeitsverteilungen an Stelle der Mengen gerechnet wird. Als Konsistenztest kann vor dem Schritt 2 noch die Überprüfung der Bedingung

$$D(k_e) > 0$$

mit  $D(k_e)$  aus Gl. (15.15) aufgenommen werden.

**Beziehungen zwischen der Zustandsbeobachtung stochastischer und nichtdeterministischer Automaten.** Aus einem stochastischen Automaten kann man einen nichtdeterministischen Automaten dadurch bilden, dass man die Verhaltensrelation auf eins aufrundet, wenn sie größer als null ist:

$$\begin{aligned} L(z', w | z, v) > 0 &\Rightarrow L(z', w, z, v) = 1 \\ L(z', w | z, v) = 0 &\Rightarrow L(z', w, z, v) = 0. \end{aligned}$$

In diesen Beziehungen steht links die Verhaltensrelation des stochastischen Automaten und rechts die Verhaltensrelation des nichtdeterministischen Automaten, den man als den im stochastischen Automaten *eingebetteten* nichtdeterministischen Automaten bezeichnet. Wendet man die Zustandsbeobachtungsalgorithmen auf den stochastischen Automaten bzw. den eingebetteten nichtdeterministischen Automaten für dasselbe E/A-Paar  $(V, W)$  an und bildet für den stochastischen Automaten die Menge  $\mathcal{Z}(k_e | k_e)$  der aktuellen Zustände entsprechend Gl. (15.16), so erhält man in beiden Fällen dasselbe Ergebnis. Die zusätzliche Information

$p(z, k_e)$ , die der stochastische Automat liefert, ist eine Bewertung der in der Menge  $\mathcal{Z}(k_e | k_e)$  stehenden Zustände, mit der man die weniger wahrscheinlichen Zustände von den Zuständen, in denen sich der Automat mit höherer Wahrscheinlichkeit aufhält, unterscheiden kann. Die zusätzliche Information über die Wahrscheinlichkeiten führt nicht zu einer Änderung der Menge  $\mathcal{Z}(k_e | k_e)$ .

**Beispiel 15.3** Zustandsbeobachtung eines stochastischen Automaten

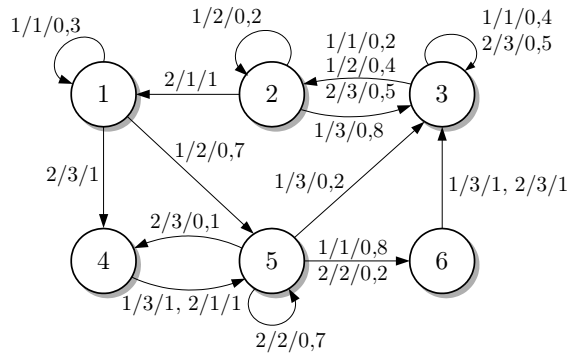
Die Beobachtungsaufgabe soll für den in Abb. 15.9 gezeigten Automaten für das E/A-Paar

$$(V, W) = ((1, 1, 2, 2), (1, 2, 2, 3)) \quad (15.17)$$

gelöst werden. Es ist dies der im Beispiel 15.2.1 betrachtete nichtdeterministische Automat, der um Übergangswahrscheinlichkeiten erweitert wurde. Die hier erhaltenen Beobachtungsergebnisse können deshalb sehr gut mit den für den nichtdeterministischen Automaten erhaltenen Ergebnissen verglichen werden. Da keine Vorinformationen über den Zustand vorhanden sind, wird mit der A-priori-Wahrscheinlichkeitsverteilung

$$p'(z, -1) = \text{Prob}(Z(0) = z) = \frac{1}{6}, \quad z \in \mathcal{Z}$$

gearbeitet, bei der alle Zustände gleich wahrscheinlich sind.



**Abb. 15.9:** Stochastischer Automat für das Beispiel 15.3

Der Beobachtungsalgorithmus 15.2 läuft für  $k_e = 0$  folgendermaßen ab. Es wird das E/A-Paar  $(V(0) = 1, W(0) = 1)$  gemessen, womit Gl. (15.12) für den Zustand  $z = 1$  die Beziehung

$$p(1, 0) = \frac{\sum_{z'} L(z', 1 | 1, 1) \cdot p'(1, -1)}{\sum_{z, z'} L(z', 1 | z, 1) \cdot p'(z, -1)}$$

liefert. Im Zähler steht die Summe

$$\begin{aligned}
& \sum_{z'} L(z', 1 | 1, 1) \cdot p'(1, -1) \\
&= (L(1, 1 | 1, 1) + L(2, 1 | 1, 1) + L(3, 1 | 1, 1) + L(4, 1 | 1, 1) + \\
&\quad + L(5, 1 | 1, 1) + L(6, 1 | 1, 1)) \cdot p'(1, -1) \\
&= (0,3 + 0 + 0 + 0 + 0 + 0) \cdot \frac{1}{6} \\
&= 0,05,
\end{aligned}$$

im Nenner die folgende Summe, in der nur die von null verschiedenen Elemente aufgeführt sind:

$$\begin{aligned}
D(0) &= \sum_{z, z'} L(z', 1 | z, 1) \cdot p'(z, -1) \\
&= L(1, 1 | 1, 1) \cdot p'(1, -1) + L(2, 1 | 3, 1) \cdot p'(3, -1) \\
&\quad + L(3, 1 | 3, 1) \cdot p'(3, -1) + L(6, 1 | 5, 1) \cdot p'(5, -1) \\
&= 0,3 \cdot \frac{1}{6} + 0,2 \cdot \frac{1}{6} + 0,4 \cdot \frac{1}{6} + 0,8 \cdot \frac{1}{6} \\
&= 0,05 + 0,0333 + 0,0666 + 0,1333 \\
&= 0,283.
\end{aligned}$$

In diese Summe gehen alle Zustandsübergänge  $z \rightarrow z'$  ein, die beim E/A-Paar  $(1, 1)$  auftreten können und in einem Zustand  $z$  mit  $p'(z, -1) > 0$  beginnen. Diese Übergänge sind im oberen Teil von Abb. 15.10 gekennzeichnet. Damit erhält man

$$p(1, 0) = \frac{0,05}{0,283} = 0,176.$$

Durch die Messung des E/A-Paares  $(1, 1)$  hat sich die Aufenthaltswahrscheinlichkeit im Zustand 1 von der A-priori-Wahrscheinlichkeit  $p'(1, -1) = 0,167$  auf den Wert  $p(1, 0) = 0,176$  erhöht.

Bei der Berechnung der Wahrscheinlichkeit  $p(z, 0)$  für die anderen Zustände tritt derselbe Nenner  $D(0)$  auf. Für den Zustand  $z = 2$  verschwindet der Zähler, weil der Automat keine Bewegung von diesem Zustand aus durchführen kann, wenn er die Eingabe  $\bar{v} = 1$  erhält und die Ausgabe  $\bar{w} = 1$  erzeugt. Deshalb gilt

$$p(2, 0) = 0.$$

Für den Zustand  $z = 3$  treten im Zähler zwei Summanden auf, weil der Automat von diesem Zustand unter dem gemessenen E/A-Paar zu zwei möglichen Nachfolgezuständen  $z' = 2$  oder  $z' = 3$  übergehen kann:

$$(L(2, 1 | 3, 1) + L(2, 1 | 3, 1)) \cdot p'(3, -1) = (0,2 + 0,4) \cdot \frac{1}{6} = 0,1.$$

Damit erhält man

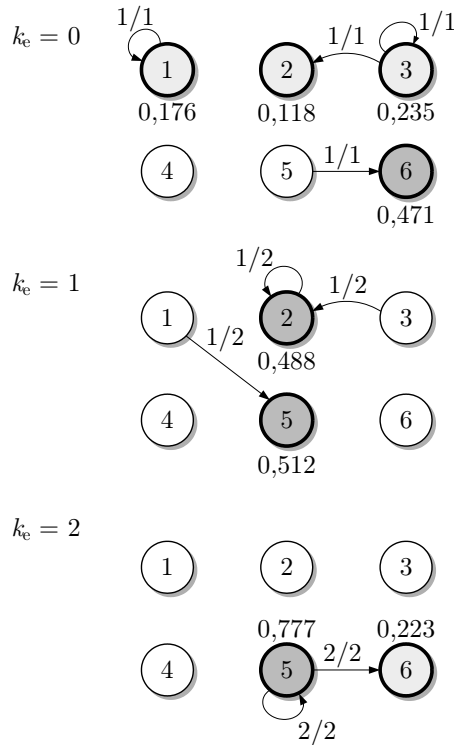
$$p(3, 0) = \frac{0,1}{0,283} = 0,353.$$

Die Wahrscheinlichkeit für den Aufenthalt im Zustand 3 hat sich also erheblich gegenüber der A-priori-Wahrscheinlichkeit erhöht.

In ähnlicher Weise erhält man für die anderen Zustände folgende Wahrscheinlichkeiten:

$$\begin{aligned}
p(4, 0) &= 0 \\
p(5, 0) &= 0,471 \\
p(6, 0) &= 0.
\end{aligned}$$

Verglichen mit dem im Beispiel 15.2.1 betrachteten nichtdeterministischen Automaten, der denselben Automatengraphen hat, sind jetzt alle diejenigen Zustände mit einer positiven Wahrscheinlichkeit bewertet, die beim nichtdeterministischen Automaten in der Menge  $\mathcal{Z}(0|0) = \{1, 3, 5\}$  lagen. Die Wahrscheinlichkeitsangaben erlauben es, die drei Zustände dieser Menge in Bezug auf die Wahrscheinlichkeit zu unterscheiden, mit der der Automat einen dieser Zustände als Anfangszustand angenommen hat.



**Abb. 15.10:** Ergebnis der Zustandsbeobachtung für den stochastischen Automaten

Im nächsten Schritt werden mit Gl. (15.13) die Wahrscheinlichkeiten  $p'(z', 0)$  für alle Nachfolgezustände  $z' = 1, 2, \dots, 6$  ermittelt. Für  $z' = 1$  muss dafür der Quotient

$$p'(1, 0) = \frac{\sum_z L(1, 1 | z, 1) \cdot p'(z, -1)}{\sum_{z, z'} L(z', 1 | z, 1) \cdot p'(z, -1)}.$$

berechnet werden. Für den Zähler erhält man

$$\sum_z L(1, 1 | z, 1) \cdot p'(z, -1) = L(1, 1 | 1, 1) \cdot p'(1, -1) = 0,3 \cdot \frac{1}{6} = 0,05.$$

Im Nenner steht wie zuvor  $D(0)$ , so dass gilt

$$p'(1, 0) = \frac{0,05}{0,283} = 0,176.$$

Für die anderen Zustände berechnet man auf dieselbe Weise folgende Aufenthaltswahrscheinlichkeiten:

$$p'(2, 0) = 0,118$$

$$p'(3, 0) = 0,235$$

$$p'(4, 0) = 0$$

$$p'(5, 0) = 0$$

$$p'(6, 0) = 0,471.$$

Diese Werte sind in Abb. 15.10 an die Knoten angetragen. Außerdem sind die Knoten entsprechend dieser Wahrscheinlichkeiten grau eingefärbt (je dunkler desto wahrscheinlicher). Die Wahrscheinlichkeit ist genau für diejenigen Zustände positiv, die beim nichtdeterministischen Automaten in der Menge

$$\mathcal{Z}(1 | 0) = \{1, 2, 3, 6\}$$

liegen.

Für  $k_e = 1$  erhält man nach der Messung des E/A-Paares ( $V(1) = 1, W(1) = 2$ ) auf dieselbe Weise folgende Wahrscheinlichkeitsverteilung:

$$p(1, 1) = 0,512$$

$$p(2, 1) = 0,098$$

$$p(3, 1) = 0,390$$

$$p(4, 1) = 0$$

$$p(5, 1) = 0$$

$$p(6, 1) = 0.$$

Durch die Messung wird offenkundig, dass der Automat zum Zeitpunkt 1 nicht im Zustand 6 gewesen sein kann, obwohl dieser Zustand die höchste Wahrscheinlichkeit

$$p'(6, 0) = 0,471$$

hatte. Deshalb erhöhen sich die Wahrscheinlichkeiten für die anderen Zustände.

Für die Nachfolgezustände ergeben sich folgende Wahrscheinlichkeiten

$$p'(1, 1) = 0$$

$$p'(2, 1) = 0,488$$

$$p'(3, 1) = 0$$

$$p'(4, 1) = 0$$

$$p'(5, 1) = 0,512$$

$$p'(6, 1) = 0.$$

Diese Ergebnisse sind in der zweiten Zeile von Abb. 15.10 veranschaulicht.

Für  $k_e = 2$  wird das E/A-Paar ( $V(2) = 2, W(2) = 2$ ) gemessen, womit man folgende Wahrscheinlichkeiten erhält:



$$\begin{aligned}
p(1, 2) &= 0 \\
p(2, 2) &= 0 \\
p(3, 2) &= 0 \\
p(4, 2) &= 0 \\
p(5, 2) &= 1 \\
p(6, 2) &= 0.
\end{aligned}$$

Zum Zeitpunkt 2 kann der Automat also nur im Zustand 5 gewesen sein. Weiterhin erhält man

$$\begin{aligned}
p'(1, 2) &= 0 \\
p'(2, 2) &= 0 \\
p'(3, 2) &= 0 \\
p'(4, 2) &= 0 \\
p'(5, 2) &= 0,777 \\
p'(6, 2) &= 0,223.
\end{aligned}$$

Obwohl der Automatenzustand für den Zeitpunkt 2 eindeutig ermittelt werden konnte, sind zum Zeitpunkt 3 auf Grund des nichtdeterministischen Verhaltens des Automaten wieder zwei Zustände möglich. Die Wahrscheinlichkeiten für den Aufenthalt in beiden Zuständen sind aber sehr unterschiedlich. Mit wesentlich höherer Wahrscheinlichkeit befindet sich der Automat im Zustand 5 (vgl. dritte Zeile von Abb. 15.10).

**Diskussion.** Das Beispiel zeigt, dass bei der Zustandsbeobachtung die Aufenthaltswahrscheinlichkeit für alle Zustände besser bestimmt werden kann als bei einer Vorhersage, denn hier werden die Informationen ausgenutzt, die in der gemessenen Ausgangsgröße stecken. Diese Tatsache kommt insbesondere dadurch zum Ausdruck, dass im Schritt 2 des Beobachtungsalgorithmus zunächst die bisher bestimmte Wahrscheinlichkeit  $p'(z, k_e)$  für den Aufenthalt im Zustand  $z$  zur Zeit  $k_e + 1$  durch Nutzung der nun bekannten Ausgabe  $w(k_e + 1)$  zur Wahrscheinlichkeit  $p(z, k_e + 1)$  korrigiert wird. Im Beispiel führt dies für den Zustand 2 von der A-priori-Wahrscheinlichkeit  $p'(2, -1) = 0,167$  auf den Wert  $p(2, 0) = 0$ . Die Messung  $w(0) = 1$  lässt also erkennen, dass sich der Automat gar nicht im Anfangszustand 2 befinden haben kann. Demgegenüber wird bei einer Simulationsuntersuchung davon ausgegangen, dass dieser Anfangszustand möglich ist und es werden auch Trajektorien berechnet, die im Anfangszustand 2 beginnen.  $\square$

## Literaturhinweise

Das Problem der Zustandsbeobachtung für ereignisdiskrete Systeme ist noch sehr wenig untersucht worden. Die erste Untersuchung hierzu stammt aus [77], wo die Idee beschrieben wird, durch Experimente mit einem System den aktuellen Zustand zu ermitteln. In [18] wird ein Beobachter für einen nichtdeterministischen Automaten beschrieben, der bei bekanntem Anfangszustand die Menge aller Automatenzustände liefert, in der sich der Automat befindet, wenn er eine bestimmte Symbolfolge akzeptiert. Der Beobachter ist ein deterministischer Automat, dessen Zustandsmenge die Potenzmenge  $2^Z$  der Zustandsmenge  $Z$  des beobachteten nichtdeterministischen Automaten ist. Für ein Beispiel ist ein derartiger Beobachter in Abb. 15.5 gezeigt.

In [62] und [81] wird die Beobachtbarkeit untersucht, wobei allerdings die eindeutige Bestimmung des Zustands auch für nichtdeterministische Automaten gefordert wird, was eine für die Praxis sehr einschränkende Eigenschaft ist. Gleiches gilt für Definitionen der Beobachtbarkeit, bei denen gefordert wird, dass

man die Zustandstrajektorie eindeutig anhand der Eingaben und Ausgaben bestimmen kann, wenn man den aktuellen Zustand erst einmal ermittelt hat. Es soll dann eine Funktion  $s$  geben, so dass  $z' = s(z, v, w)$  gilt [17, 21]. Diese Art der Beobachtbarkeitsdefinition lässt jedoch das Hauptproblem offen, nämlich die Bestimmung des aktuellen Zustands aus den Eingabe- und Ausgabefolgen, bevor die genannte Eigenschaft ausgenutzt werden kann, um die Zustandstrajektorie zu verfolgen.

Die hier angegebenen Methoden, die auch mehrdeutige Lösungen für die Beobachtungsaufgabe zulassen und für stochastische Automaten erweitert sind, wurden in [73] veröffentlicht.

Das Beobachtungsproblem kann auch für Petrinetze formuliert werden: Bei unbekannter Anfangsmarkierung  $M_0$  soll durch Auswertung der Eingabe- und Ausgabefolgen  $V(0 \dots k_e)$ ,  $W(0 \dots k_e)$  die aktuelle Markierung  $M(k_e)$  bestimmt werden. Diese Aufgabe ist sinnvoll, wenn in Erweiterung zu der hier beschriebenen Interpretation von Petrinetzen die Ausgabe nicht eindeutig einer Stelle zugeordnet werden kann, sondern unterschiedliche Markierungen auf denselben Ausgabewert  $w(k)$  führen können. Erste Ansätze wurden in [35, 36] beschrieben.

# 16

## Diagnose diskreter Systeme

*Durch die Interpretation der Diagnoseaufgabe als eine Aufgabe der Zustandsbeobachtung für ein möglicherweise fehlerbehaftetes System werden in diesem Kapitel die Beobachtungsmethoden für nichtdeterministische und stochastische Automaten zu Diagnosemethoden erweitert, mit denen Fehler erkannt und identifiziert werden können.*

### 16.1 Diagnoseaufgabe

Es wird in diesem Kapitel untersucht, wie Fehler in einem diskreten Prozess diagnostiziert werden können. Dabei wird – wie bei der Diagnose kontinuierlicher Prozesse im Kap. 10 – ein Beobachter verwendet. Die hier betrachtete Diagnoseaufgabe unterscheidet sich von der Diagnoseaufgabe für kontinuierliche Systeme nur dadurch, dass jetzt diskrete Eingangs- und Ausgangsfolgen als Messgrößen auftreten und dass ein Automat als Prozessmodell verwendet wird.

#### **Diagnoseaufgabe für diskrete Systeme**

Gegeben:	Prozessmodell (Automat)
	Folge der Eingangssymbole $V(0...k_e)$
	Folge der Ausgangssymbole $W(0...k_e)$
Gesucht:	Fehler $f$

Die Lösung der Diagnoseaufgabe beruht auf einer Konsistenzprüfung der gemessenen Eingangs- und Ausgangsfolgen mit dem Prozessmodell. Für die Fehlerdetektion wird über-

prüft, ob das gemessene E/A-Paar mit dem Modell für den fehlerfreien Betrieb der Anlage konsistent ist. Ist dies nicht der Fall, so ist erkannt, dass ein Fehler eingetreten ist.

Für die Fehleridentifikation braucht man Modelle des fehlerbehafteten Systems. Wird bei der Konsistenzprüfung der gemessenen E/A-Folge mit dem für den Fehler  $f$  aufgestellten Modell eine Inkonsistenz erkannt, so kann der Fehler  $f$  ausgeschlossen werden.

Die Lösung der Diagnoseaufgabe ist die Menge  $\mathcal{F}(k_e)$  der *Fehlerkandidaten*.  $f$  ist zur Zeit  $k_e$  ein Fehlerkandidat, wenn das E/A-Paar für den Zeithorizont  $k_e$  mit dem Modell, das den Prozess unter der Wirkung des Fehlers  $f$  beschreibt, konsistent ist.

## 16.2 Diagnose nichtdeterministischer Automaten

### 16.2.1 Modellierung fehlerbehafteter Systeme

Für die Fehleridentifikation muss das Systemverhalten durch Automaten beschrieben werden, die für die Fehler  $f \in \mathcal{F}$  gebildet werden. Dabei bezeichnet  $\mathcal{F}$  die Menge aller betrachteten Fehler (*Fehlerraum*). Ihre Elemente heißen  $f_i$

$$\mathcal{F} = \{f_0, f_1, f_2, \dots, f_q\}$$

oder die Fehler sind durchnummeriert:

$$\mathcal{F} = \{0, 1, 2, \dots, q\}.$$

Zur Vereinheitlichung der Darstellung wird auch der fehlerfreie Zustand in der Fehlermenge erfasst und durch das Symbol  $f_0$  oder  $f = 0$  gekennzeichnet. Mehrfachfehler, die das gleichzeitige Auftreten von zwei oder mehreren der zuvor definierten Fehler repräsentieren, lassen sich problemlos dadurch berücksichtigen, dass man für sie eigene Symbole  $f$  im Fehlerraum vorsieht.

Ob es sich bei der Diagnose um eine Fehlerlokalisierung oder eine Fehleridentifikation handelt, hängt von der Bedeutung des Fehlers  $f_i$  ab, denn bei der Modellierung von Fehlern in diskreten Systemen kann  $f_i$  sowohl für den Fehler in einer bestimmten Komponente als auch für eine konkrete Ausprägung eines Fehlers stehen. Es wird deshalb im Folgenden nicht zwischen Fehlerlokalisierung und Fehleridentifikation unterschieden und stets der zweite Begriff verwendet.

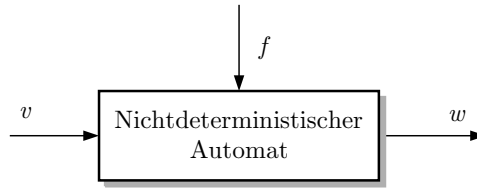
Die Systembeschreibung durch nichtdeterministische Automaten kann man auf fehlerbehaftete Systeme dadurch erweitern, dass man die Fehler als zusätzliche Eingangssymbole interpretiert (Abb. 16.1). Der Automat

$$\mathcal{N} = (\mathcal{Z}, \mathcal{V}, \mathcal{F}, \mathcal{W}, L, \mathcal{Z}_0) \quad (16.1)$$

hat die Verhaltensrelation  $L$ , für die die Beziehung

$$L(z(k+1), w(k), z(k), v(k), f(k)) = 1$$

gilt, wenn unter der Wirkung des Fehlers  $f(k)$  bei der Eingabe  $v(k)$  der Zustandsübergang  $z(k) \rightarrow z(k+1)$  auftreten kann, wobei die Ausgabe  $w(k)$  erzeugt wird.



**Abb. 16.1:** Nichtdeterministischer Automat unter der Wirkung des Fehlers  $f$

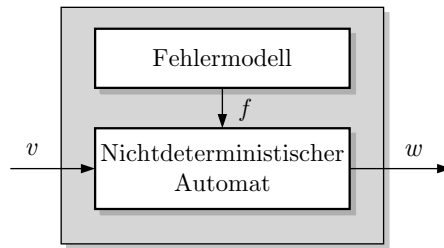
Das Modell  $\mathcal{N}$  lässt zu, dass sich der Fehler  $f(k)$  zeitlich beliebig verändert. Häufig macht man jedoch Annahmen über das Verhalten des Fehlers. In den nächsten Abschnitten wird die Diagnoseaufgabe für konstante Fehler gelöst. Diese Annahme bedeutet, dass sich der Fehler während der Anwendung des Diagnoseverfahrens nicht ändert. Für die Erweiterung der dabei entwickelten Methode auf zeitabhängige Fehler erfasst man die Eigenschaften des Fehlers in einem *Fehlermodell*. Da der Fehler diskret ist, eignet sich dafür ein autonomer nichtdeterministischer Automat, der mit

$$\mathcal{N}_F = (\mathcal{F}, G_F, \mathcal{F}_0) \quad (16.2)$$

bezeichnet wird.  $G_F$  ist die Zustandsübergangsrelation, die angibt, welche Werte  $f(k+1)$  das Fehlersignal zur Zeit  $k+1$  in Abhängigkeit vom Wert  $f(k)$  annehmen kann:

$$f(k+1) \in G_F(f(k)).$$

$\mathcal{F}_0$  ist die Menge von Anfangszuständen des Fehlermodells, die i. Allg. unbekannt ist ( $\mathcal{F}_0 = \mathcal{F}$ ).



**Abb. 16.2:** Nichtdeterministischer Automat mit Fehlermodell

Fasst man das Modell des zu diagnostizierenden Automaten und das Fehlermodell zusammen, so entsteht der nichtdeterministische Automat

$$\tilde{\mathcal{N}} = (\tilde{\mathcal{Z}}, \mathcal{V}, \mathcal{W}, \tilde{L}, \tilde{\mathcal{Z}}_0) \quad (16.3)$$

mit dem Zustand

$$\tilde{z} = \begin{pmatrix} z \\ f \end{pmatrix}$$

(Abb. 16.2), der nur noch den Eingang  $v(k)$  und den Ausgang  $w(k)$  hat. Die Verhaltensrelation  $\tilde{L}$  erhält man aus der Verhaltensrelation  $L$  und der Zustandsübergangsrelation  $G_F$  entsprechend folgender Beziehung:

$$\tilde{L}\left(\begin{pmatrix} z' \\ f' \end{pmatrix}, w, \begin{pmatrix} z \\ f \end{pmatrix}, v\right) = L(z', w, z, v, f) \cdot G(f', f), \quad (16.4)$$

wobei die Nachfolgezustände wieder mit einem Apostroph gekennzeichnet sind. Diese Gleichung besagt, dass die Verhaltensrelation  $\tilde{L}$  genau die Zustandsübergänge  $z \rightarrow z'$  und Änderungen  $f \rightarrow f'$  des Fehlers zulässt, die aus einem Zustandsübergang  $(z', w, z, v, f)$  des Systems für den Fehler  $f$  und einem Wechsel  $(f', f)$  des Fehlers entsprechend dem Fehlermodell bestehen:

$$\tilde{L}\left(\begin{pmatrix} z' \\ f' \end{pmatrix}, w, \begin{pmatrix} z \\ f \end{pmatrix}, v\right) = 1 \iff L(z', w, z, v, f) = 1 \text{ und } G(f', f) = 1.$$

Gleichung (16.4) zeigt, dass der Fehler jetzt ein Teil des unbekannten Zustands des zu diagnostizierenden Systems ist. Deshalb kann die Diagnoseaufgabe mit Hilfe des Beobachtungsalgorithmus für nichtdeterministische Automaten gelöst werden.

Absesehen von den Unterschieden, die sich aus den verwendeten Modellen ergeben, ist diese Vorgehensweise dieselbe wie bei kontinuierlichen Systemen (vgl. Abb. 10.25 auf S. 318).

### 16.2.2 Detektion konstanter Fehler

Die Diagnoseaufgabe wird zunächst für konstanten Fehler  $f(k) = \bar{f}$  gelöst. Die Annahme eines konstanten Fehlers ist so zu interpretieren, dass sich das Fehlersignal  $f$  im Zeitintervall  $0 \dots k_e$ , für das die Eingangs- und Ausgangsfolgen vom Diagnosealgorithmus ausgewertet werden, nicht verändert. In Anwendungen kann man dieser Voraussetzung dadurch Rechnung tragen, dass man den Diagnosealgorithmus von Zeit zu Zeit initialisiert, wodurch das angegebene Zeitintervall von Neuem beginnt.

Für die Fehlererkennung braucht man nur das Modell (16.1) des fehlerfreien Systems ( $f = f_0$ ), dessen Verhaltensrelation als  $L_0$  bezeichnet wird:

$$L_0(z', w, z, v) = L(z', w, z, v, f_0).$$

Die Fehlererkennung erfolgt durch einen Konsistenztest des gemessenen E/A-Paares mit diesem Automaten, wofür der Beobachtungsalgorithmus 15.1 geringfügig erweitert wird. Der Algorithmus arbeitet mit der Zustandsmenge  $\mathcal{Z}$  als Initialisierung für die Menge  $\mathcal{Z}(0 \mid -1)$ . Nach dem Schritt 2 der Iterationsschleife wird getestet, ob die Zustandsmengen  $\mathcal{Z}$  und  $\mathcal{Z}'$  leer sind. Ist dies der Fall, so ist die Inkonsistenz festgestellt.

Der Fehlerfall wird vom fehlerfreien Fall durch die Größe  $p(k_e)$  unterschieden.  $p(k_e) = 1$  bedeutet, dass das E/A-Paar bis zum Zeitpunkt  $k_e$  mit dem Automaten konsistent ist und das System folglich als fehlerfrei betrachtet wird.  $p(k_e) = 0$  heißt, dass das E/A-Paar mit dem für das fehlerfreie System aufgestellten Automaten inkonsistent ist und folglich ein Fehler aufgetreten sein muss.

Mit diesen Ergänzungen entsteht aus dem Algorithmus 15.1 der folgende Diagnosealgorithmus, in dem wieder die Abkürzungen

$$\begin{aligned}\mathcal{Z} &= \mathcal{Z}(k_e \mid k_e) \\ \mathcal{Z}' &= \mathcal{Z}(k_e + 1 \mid k_e)\end{aligned}$$

verwendet werden.

---

**Algorithmus 16.1** *Erkennung konstanter Fehler in nichtdeterministischen Automaten*


---

**Gegeben:** Automat mit der Verhaltensrelation  $L(z', w \mid z, v)$ .

Initialisierung:  $\mathcal{Z}' := \mathcal{Z}$   
 $k_e := 0$

Schleife: 1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .

2. Bestimmen Sie die Mengen

$$\mathcal{Z} := \{z \in \mathcal{Z}' \mid \exists z' : L_0(z', \bar{w}, z, \bar{v}) = 1\}$$

$$\mathcal{Z}' := \{z' \mid \exists z \in \mathcal{Z} : L_0(z', \bar{w}, z, \bar{v}) = 1\}.$$

$$3. p(k_e) := \begin{cases} 0 & \text{wenn } \mathcal{Z} = \emptyset \\ 1 & \text{sonst.} \end{cases}$$

$$4. k_e := k_e + 1$$

Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $p(k_e)$  für  $k_e = 0, 1, \dots$

---

Mit diesem Algorithmus kann die Fehlererkennung durchgeführt werden. Ist  $p(k_e) = 1$ , so kann kein Fehler im System festgestellt werden. Für  $p(k_e) = 0$  ist das System fehlerbehaftet. Der Schluss, dass aus der Inkonsistenz des E/A-Paares die Existenz eines Fehlers folgt, ist natürlich an die Voraussetzung gebunden, dass auf das System keine so große Störung einwirkt, dass das fehlerfreie System ein inkonsistentes E/A-Paar erzeugt.

### 16.2.3 Identifikation konstanter Fehler

Für die Fehleridentifikation ist eine Beschreibung des Systemverhaltens unter der Wirkung der Fehler  $f \in \mathcal{F}$  notwendig. Dafür wird im Folgenden der nichtdeterministische Automat (16.1)

$$\mathcal{N} = (\mathcal{Z}, \mathcal{V}, \mathcal{F}, \mathcal{W}, L, \mathcal{Z}_0),$$

verwendet, dessen Verhaltensrelation jetzt für konstante Fehler  $f$  angewendet wird. Man kann die Diagnose deshalb so durchführen, dass man den Algorithmus 16.1 für alle Fehler  $f \in \mathcal{F}$  getrennt anwendet und überprüft, für welche Fehler das E/A-Paar mit dem für  $f$  gültigen Automat konsistent ist. Dabei wird die Verhaltensrelation  $L(z', w, z, v, f)$  für alle konstanten Fehler  $f \in \mathcal{F}$  die Verhaltensrelation

$$L_f(z', w, z, v) = L(z', w, z, v, f)$$

gebildet und der Algorithmus 16.1 auf diese Verhaltensrelation angewendet. Dies entspricht in der Grundidee der in Abb. 10.24 auf S. 316 dargestellten Diagnose kontinuierlicher Systeme mit einer Beobachterbank, bei der der Konsistenztest mit Modellen, die für die einzelnen Fehler  $f \in \mathcal{F}$  gelten, getrennt durchgeführt wird und aus der Inkonsistenz der Messungen mit dem Modell für den Fehler  $f$  gefolgert wird, dass der Fehler  $f$  nicht im System aufgetreten ist.

Für eine kompaktere Darstellung wird der im Algorithmus 16.1 auftretende Fehlerindikator  $p(k_e)$  jetzt auf eine Funktion des Fehlers  $f$  erweitert.  $p(f, k_e) = 1$  bedeutet, dass das E/A-Paar bis zum Zeithorizont  $k_e$  mit dem für den Fehler  $f$  geltenden Modell konsistent ist.  $p(f, k_e) = 0$  zeigt eine Inkonsistenz an. Diese Größe wird im Schritt 3 des Algorithmus anhand der für den Fehler  $f$  gebildeten Mengen  $\mathcal{Z}(k_e | k_e)$  und  $\mathcal{Z}(k_e + 1 | k_e)$  berechnet, die ebenfalls für alle Fehler getrennt gebildet werden müssen und deshalb  $f$  als zusätzliches Argument erhalten. In Erweiterung der Gln. (15.2) und (15.3) erhält man jetzt für das Modell des fehlerbehafteten Systems

$$\mathcal{Z}(f, k_e | k_e) = \{z \in \mathcal{Z}(f, k_e | k_e - 1) \mid \exists z' : L(z', \bar{w}, z, \bar{v}, f) = 1\} \quad (16.5)$$

$$\mathcal{Z}(f, k_e + 1 | k_e) = \{z' \mid \exists z \in \mathcal{Z}(f, k_e | k_e - 1) : L(z', \bar{w}, z, \bar{v}, f) = 1\}. \quad (16.6)$$

Diese Mengen werden im folgenden Algorithmus durch die Symbole

$$\mathcal{Z}(f) = \mathcal{Z}(f, k_e | k_e)$$

$$\mathcal{Z}'(f) = \mathcal{Z}(f, k_e + 1 | k_e)$$

repräsentiert. Die Menge  $\mathcal{F}(k_e)$  enthält diejenigen Fehler  $f$ , zu deren Modell das E/A-Paar mit dem Zeithorizont  $k_e$  konsistent ist (Fehlerkandidaten).

**Diagnoseergebnis.** Der Algorithmus enthält keine Abbruchbedingung, weil er, im Prinzip, einen Prozess für beliebig lange Eingangs-Ausgangsfolgen überwachen kann. Er liefert zu jedem Zeitpunkt  $k_e$  die Menge  $\mathcal{F}(k_e)$  der Fehlerkandidaten, die für die Fehlerdiagnose auf die folgenden Aussagen führt:

- **Fehlerdetektion:** Wenn  $f_0 \notin \mathcal{F}(k_e)$  gilt, wobei  $f_0$  den fehlerfreien Betriebszustand bezeichnet, so ist erwiesen, dass der Automat fehlerbehaftet ist.
- **Fehleridentifikation:** Der im System vorhandene Fehler  $f$  gehört zur Menge  $\mathcal{F}(k_e)$ . Wenn  $\mathcal{F}(k_e) = \{f_i\}$  eine einelementige Menge ist, so ist erwiesen, dass der Fehler  $f = f_i$  aufgetreten ist (vorausgesetzt, dass der aufgetretene Fehler zum betrachteten Fehlerraum  $\mathcal{F}$  gehört).

Wenn im Schritt 3 des Algorithmus festgestellt wird, dass die betrachteten Mengen für alle Fehler  $f$  leer sind, so ist ein Fehler detektiert, der außerhalb des Fehlerraumes  $\mathcal{F}$  liegt. Man kann dann den Algorithmus durch eine zusätzlich Abbruchbedingung beenden, weil mit den vorhandenen Modellen keine genauere Aussage über den Fehler möglich ist.



**Algorithmus 16.2** Identifikation konstanter Fehler in nichtdeterministischen Automaten

**Gegeben:** Automat für die Fehler  $f \in \mathcal{F}$  mit der Verhaltensrelation  $L(z', w, z, v, f)$ .

Initialisierung:  $\mathcal{Z}'(f) = \mathcal{Z}$  für alle  $f \in \mathcal{F}$   
 $k_e = 0$

Schleife: 1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .

2. Bestimmen Sie für alle  $f \in \mathcal{F}$  die Mengen

$$\mathcal{Z}(f) := \{z \in \mathcal{Z}(f) \mid \exists z' : L_f(z', \bar{w}, z, \bar{v}) = 1\}$$

$$\mathcal{Z}'(f) := \{z' \mid \exists z \in \mathcal{Z}(f) : L_f(z', \bar{w}, z, \bar{v}) = 1\}.$$

3. Bestimmen Sie für alle  $f \in \mathcal{F}$

$$p(f, k_e) = \begin{cases} 0 & \text{wenn } \mathcal{Z}(f) = \emptyset \\ 1 & \text{sonst.} \end{cases}$$

4. Bestimmen Sie die Menge der Fehlerkandidaten

$$\mathcal{F}(k_e) = \{f \mid p(f, k_e) = 1\}.$$

5.  $k_e := k_e + 1$

Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $\mathcal{F}(k_e)$  für  $k_e = 0, 1, \dots$

**Beispiel 16.1** Diagnose eines Batchprozesses

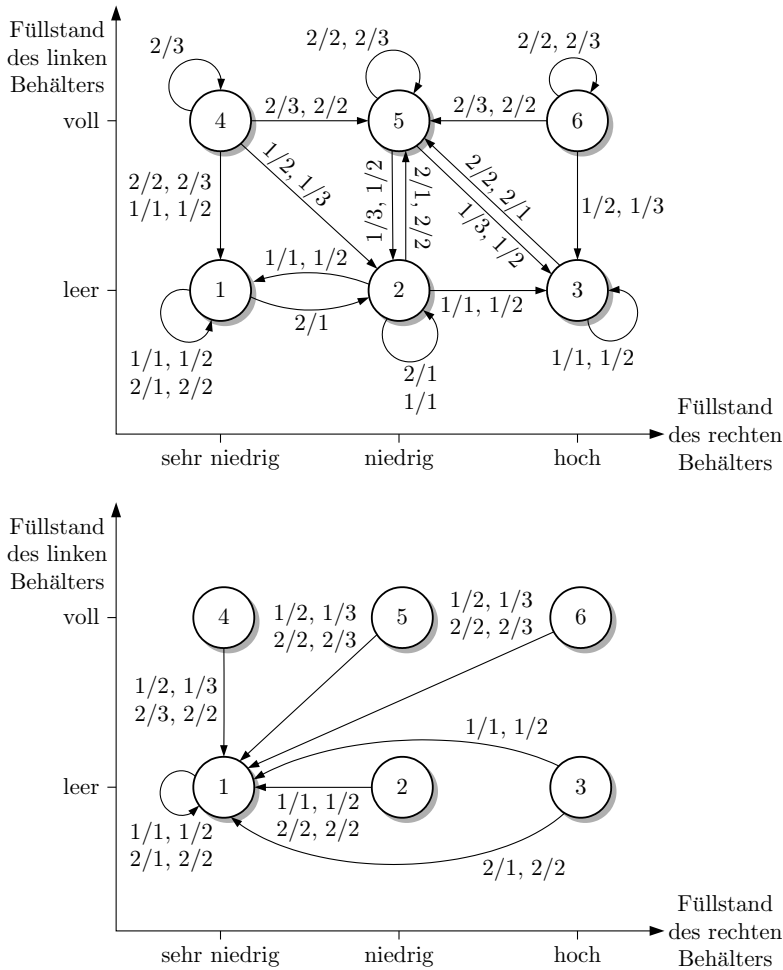
Es wird ein Batchprozess betrachtet, der ähnlich dem im Beispiel 12.5 auf S. 439 gezeigten System aufgebaut ist, nur dass die Füllstände des linken Behälters jetzt durch zwei diskrete Größen („voll“, „leer“) beschrieben werden. Das diskrete Modell hat deshalb  $2 \cdot 3 = 6$  Zustände. Das Ventil  $V_2$  ist offen. Die Ausgabe beschreibt den Durchfluss durch dieses Ventil in drei Stufen („groß“ (3), „mittelgroß“ (2), „klein“ (1)).

Ein fehlerhaftes Verhalten kann dadurch verursacht sein, dass das Ventil  $V_1$ , das beispielsweise beim Durchlaufen eines Rezepts zum Leeren des linken Behälters verwendet wird, klemmt und folglich den Steuereingriff „Ventil  $V_1$  schließen“ nicht ausführt. Dieser Fehler wird mit  $f = 1$ , der fehlerfreie Zustand mit  $f = 0$  bezeichnet.

Das System ist für den fehlerfreien Fall und den Fehlerfall durch die in Abb. 16.3 gezeigten Automaten beschrieben. Diese beiden Automaten haben die Verhaltensrelation  $L_0$  bzw.  $L_1$ . Man erkennt, dass sich beide Automaten wesentlich voneinander unterscheiden. Im Fehlerfall werden beide Behälter entleert, weil die Pumpenleistung nicht ausreicht, um die durch das offene Ventil abfließende Flüssigkeit im linken Behälter zu ersetzen. Beide Modelle sind nichtdeterministisch, weil die Füllstände und die Pumpenleistung nur qualitativ beschrieben werden.

Die Fehleridentifikation soll für den Fall betrachtet werden, dass der Batchprozess mit konstantem Eingang  $v = 2$  („Die Pumpe ist eingeschaltet.“) betrieben wird und die Ausgangsfolge

$$W(0\dots 3) = (3, 2, 2, 3)$$



**Abb. 16.3:** Automatengraph für den fehlerfreien Zustand  $f = 0$  (oben) und den Fehlerfall  $f = 1$  (unten)

erzeugt. Durch Anwendung des Algorithmus 16.2 kann man schrittweise überprüfen, ob dieses E/A-Paar mit dem Modell des fehlerfreien und des fehlerbehafteten Systems konsistent ist.

1.  $k_e = 0$ :

Für das E/A-Paar ( $v(0) = 2, w(0) = 3$ ) erhält man folgende Mengen:

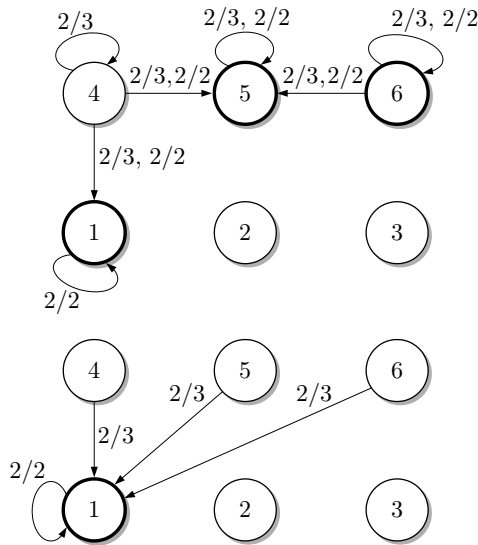
$$\mathcal{Z}(0) = \{4, 5, 6\}$$

$$\mathcal{Z}'(0) = \{1, 4, 5, 6\}$$

$$\mathcal{Z}(1) = \{4, 5, 6\}$$

$$\mathcal{Z}'(1) = \{1\}.$$

Das erste Messergebnis (2, 3) ist also mit beiden Automaten konsistent, was in Abb. 16.5 durch



**Abb. 16.4:** Teile des Automatengraphen, die für die Lösung der Diagnoseaufgabe wesentlich sind

ein schwarzes Rechteck für beide Fehlerfälle gekennzeichnet ist. Für den Fehlerfall ist der Zustand zum Zeitpunkt 1 durch dieses E/A-Paar bereits eindeutig festgelegt.

2.  $k_e = 1$ :

Für das E/A-Paar  $(v(1) = 2, w(1) = 2)$  erhält man folgende Mengen:

$$\mathcal{Z}(0) = \{1, 4, 5, 6\}$$

$$\mathcal{Z}'(0) = \{1, 5, 6\}$$

$$\mathcal{Z}(1) = \{1\}$$

$$\mathcal{Z}'(1) = \{1\}.$$

Auch das zweite E/A-Paar ist konsistent mit beiden Automaten. Der Algorithmus zeigt, dass sich das fehlerfreie System in einem der Zustände 1, 5 oder 6 und das fehlerbehaftete System im Zustand 1 befindet (Abb. 16.4).

3.  $k_e = 2$ :

Für das E/A-Paar  $(v(2) = 2, w(2) = 2)$  erhält man folgende Mengen:

$$\mathcal{Z}(0) = \{1, 5, 6\}$$

$$\mathcal{Z}'(0) = \{1, 5, 6\}$$

$$\mathcal{Z}(1) = \{1\}$$

$$\mathcal{Z}'(1) = \{1\}.$$

Das E/A-Paar ist weiterhin mit beiden Automaten konsistent.

4.  $k_e = 3$ :

Für das E/A-Paar ( $v(3) = 2, w(3) = 3$ ) erhält man folgende Mengen:

$$\mathcal{Z}(0) = \{5, 6\}$$

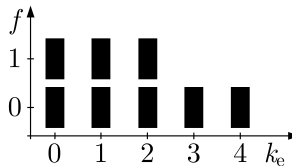
$$\mathcal{Z}'(0) = \{5, 6\}$$

$$\mathcal{Z}(1) = \emptyset$$

$$\mathcal{Z}'(1) = \emptyset.$$

Die vierte Messung zeigt, dass das Verhalten des Systems inkonsistent mit dem Modell für den Fehlerfall  $f = 1$  ist, aber weiterhin konsistent mit dem Modell des fehlerfreien Systems ( $f = 0$ ). Damit ist gezeigt, dass das System fehlerfrei arbeitet (zumindest, wenn man nur den einen Fehler in die Untersuchungen einbezieht).

**Diskussion.** In Abb. 16.5 ist die Konsistenz des E/A-Paares in Abhängigkeit vom Zeithorizont  $k_e$  durch Balken markiert. Die Konsistenz wird für beide Modelle bis zum Zeithorizont  $k_e = 2$  festgestellt. Für den Zeithorizont bis zur Zeit  $k = 3$  ist die Messung mit dem Modell des fehlerbehafteten Prozesses inkonsistent. Das Ergebnis der Diagnose besagt also, dass das System *nicht* fehlerbehaftet ist. Auch das ist ein Diagnoseergebnis. Diagnose heißt nicht zwangsläufig, dass das Ergebnis stets in einer Fehlerangabe bestehen muss.



**Abb. 16.5:** Diagnoseergebnis für  $v = 2$  und  $f = 0$

Die Fehlerfreiheit wurde übrigens daran erkannt, dass der Wasserstand im zweiten Behälter wieder anstieg ( $w(2) = 2 \rightarrow w(3) = 3$ ). Das kann nicht geschehen, wenn das Ventil des linken Behälters geöffnet ist.

Wie in diesem Beispiel lässt sich bei vielen Anwendungen das Diagnoseergebnis gut im Nachhinein interpretieren. Das darf jedoch nicht zu der Folgerung verleiten, man könne die Diagnose auch ohne Modell durch „scharfes Hinsehen“ lösen. Selbst bei den hier behandelten sehr einfachen Beispielen ist die Dynamik der Systeme so vielfältig, dass man das Diagnoseergebnis nicht auf heuristischem Wege ermitteln kann. Welcher Leser wäre wohl beim Ansehen der Ausgabefolge ohne Modell auf dieses Ergebnis gekommen? □

#### **Aufgabe 16.1\*** Diagnose eines Batchprozesses

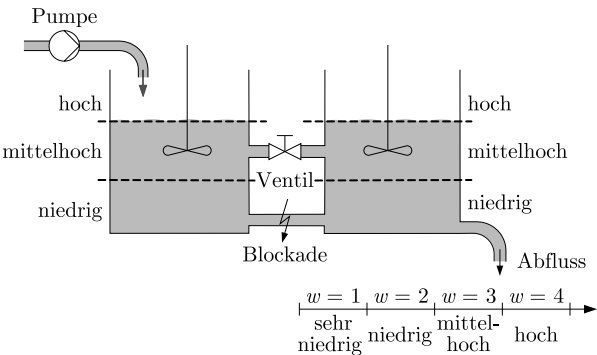
Lösen Sie die Diagnoseaufgabe für den im Beispiel 16.1 betrachteten Batchprozess für die folgenden E/A-Paare:

$$V(0...3) = (2, 2, 2, 2), W(0...3) = (3, 3, 2, 2) \quad (16.7)$$

$$V(0...3) = (1, 1, 1, 1), W(0...3) = (3, 2, 2, 1). \quad \square \quad (16.8)$$

**Aufgabe 16.2\***    *Diagnose zweier gekoppelter Reaktoren*

Betrachten Sie die in Abb. 16.6 gezeigten gekoppelten Reaktoren, die für die dort angegebenen Füllstandsgrenzwerte durch einen Automaten mit der folgenden Zustandsmenge beschrieben werden, wobei  $h_1$  den Füllstand des linken Reaktors und  $h_2$  den Füllstand des rechten Reaktors bezeichnet.



**Abb. 16.6:** Zwei gekoppelte Reaktoren

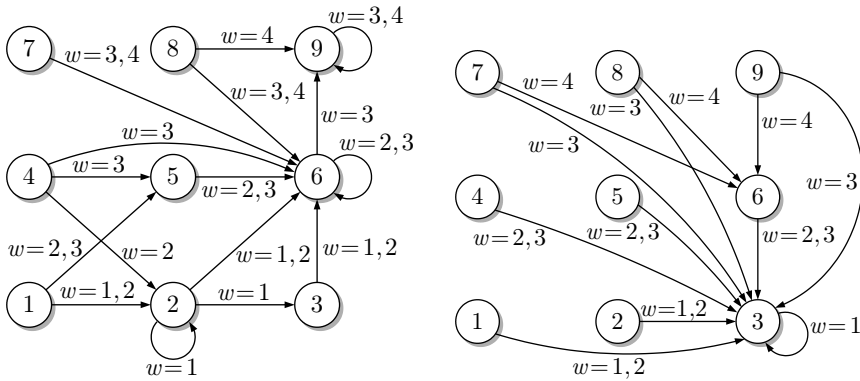
$z$	Füllstand $h_1$	Füllstand $h_2$
1	niedrig	niedrig
2	mittelhoch	niedrig
3	hoch	niedrig
4	niedrig	mittelhoch
5	mittelhoch	mittelhoch
6	hoch	mittelhoch
7	niedrig	hoch
8	mittelhoch	hoch
9	hoch	hoch

Der gemessene Ausfluss aus dem rechten Reaktor wird in vier Quantisierungsstufen gemessen, wobei der Ausgabe die folgenden Werte zugewiesen werden:

$w$	Ausfluss aus dem rechten Reaktor
1	sehr niedrig
2	niedrig
3	mittelhoch
4	hoch

Das System soll diagnostiziert werden, während das Ventil geschlossen und die Pumpe in ständigem Betrieb ist. Dabei ist zu untersuchen, ob die untere Verbindungsleitung zwischen den Behältern blockiert ist.

Unter den angegebenen konstanten Eingangsgrößen können die Reaktoren wie ein autonomes System behandelt werden. Die in Abb. 16.7 angegebenen Automatengraphen betreffen gerade diese konstanten Eingangsgrößen. An den Kanten stehen deshalb nur die Ausgaben.



**Abb. 16.7:** Automatengraph der fehlerfreien Reaktoren (links) und der Reaktoren mit blockiertem Verbindungsrohr (rechts)

Führen Sie eine Fehlerdiagnose für folgende Messfolgen durch

- (a)  $W = (4, 3, 1)$
- (b)  $W = (2, 1, 1, 1)$
- (c)  $W = (4, 2, 1, 1)$ .  $\square$

### Aufgabe 16.3 Diagnose einer Raumbelichtung

Betrachten Sie die Raumbelichtung aus Aufgabe 11.8 und formen Sie das Modell so um, dass Sie mit dem in diesem Kapitel angegebenen Diagnosealgorithmus den Fehler finden können. Was sind die Ausgangsgrößen des Systems, die Sie für die Fehlerdiagnose nutzen können? Wie verläuft die Diagnose? Unter welcher Bedingung kann der Fehler eindeutig erkannt werden?  $\square$

## 16.2.4 Identifikation zeitabhängiger Fehler

Die Diagnose bei zeitabhängigen Fehlern nutzt das Fehlermodell (16.2), das die Folgen

$$F(0 \dots k_e) = (f(0), f(1), f(2), \dots, f(k_e)),$$

die der Fehler  $f$  durchlaufen kann, einschränkt. Die Diagnose erfolgt deshalb unter Nutzung des Modells (16.3) des zu diagnostizierenden Systems, das das Fehlermodell einschließt und die Verhaltensrelation (16.4) besitzt.

Die Fehlerdiagnose kann durch direkte Anwendung des Beobachtungsalgorithmus 15.1 auf das Modell  $\tilde{\mathcal{N}}$  durchgeführt werden. Da der Modellzustand  $\tilde{z} = (z, f)^T$  den Systemzustand und den Fehler umfasst, stehen in den Mengen  $\tilde{\mathcal{Z}}(k_e | k_e)$  und  $\tilde{\mathcal{Z}}(k_e + 1 | k_e)$  die zur Zeit  $k_e$  bzw.  $k_e + 1$  möglichen Zustands-Fehler-Vektoren, weswegen diese Mengen wie der Zustand mit

einer Tilde versehen sind. Eine direkte Anwendung der Gln. (15.2) und (15.3) führt auf die Beziehungen

$$\begin{aligned}\tilde{Z}(f, k_e | k_e) &= \{\tilde{z} \in \tilde{Z}(f, k_e | k_e - 1) \mid \exists \tilde{z}' : \tilde{L}(\tilde{z}', \bar{w}, \tilde{z}, \bar{v}) = 1\} \\ \tilde{Z}(f, k_e + 1 | k_e) &= \{\tilde{z}' \mid \exists \tilde{z} \in \tilde{Z}(f, k_e | k_e) : \tilde{L}(\tilde{z}', \bar{w}, \tilde{z}, \bar{v}) = 1\},\end{aligned}$$

in denen  $\bar{v}$  und  $\bar{w}$  wieder das aktuelle E/A-Paar bilden sowie  $\tilde{z}$  den aktuellen Zustand und  $\tilde{z}'$  den Nachfolgezustand beschreiben. Unter Nutzung der Gl. (16.4) kann man diese Mengen in einer für die Fehlerdiagnose übersichtlicheren Form schreiben:

$$\tilde{Z}(k_e | k_e) = \{(z, f)^T \in \tilde{Z}(k_e | k_e - 1) \mid \exists z', f' : L(z', \bar{w}, z, \bar{v}, f) \cdot G(f', f) = 1\} \quad (16.9)$$

$$\tilde{Z}(k_e + 1 | k_e) = \{(z', f')^T \mid \exists (z, f)^T \in \tilde{Z}(k_e | k_e) : L(z', \bar{w}, z, \bar{v}, f) \cdot G(f', f) = 1\}. \quad (16.10)$$

Um die Fehlerkandidaten zum Zeitpunkt  $k_e$  zu bestimmen, müssen alle in der Menge  $\tilde{Z}(k_e | k_e)$  vorkommenden Fehlersymbole zur Menge  $\mathcal{F}$  vereinigt werden:

$$\mathcal{F}(k_e) = \{f \mid \exists (z, f)^T \in \tilde{Z}(k_e | k_e)\}. \quad (16.11)$$

Führt man wieder die Abkürzungen  $\tilde{Z} = \tilde{Z}(k_e | k_e)$  und  $\tilde{Z}' = \tilde{Z}(k_e + 1 | k_e)$  ein, so erhält man den folgenden Diagnosealgorithmus:

### Algorithmus 16.3 Diagnose nichtdeterministischer Automaten mit zeitabhängigen Fehlern

**Gegeben:** Automat mit der Verhaltensrelation  $L(z', w, z, v, f)$   
Fehlermodell mit der Zustandsübergangsrelation  $G(f', f)$ .

Initialisierung:  $\tilde{Z}' = \mathcal{Z} \times \mathcal{F}$   
 $k_e = 0$

Schleife: 1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .

2. Bestimmen Sie die Mengen

$$\tilde{Z} := \{(z, f)^T \in \tilde{Z}' \mid \exists z', f' : L(z', \bar{w}, z, \bar{v}, f) \cdot G(f', f) = 1\}$$

$$\tilde{Z}' := \{(z', f')^T \mid \exists (z, f)^T \in \tilde{Z} : L(z', \bar{w}, z, \bar{v}, f) \cdot G(f', f) = 1\}.$$

3. Bestimmen Sie die Menge der Fehlerkandidaten

$$\mathcal{F}(k_e) = \{f \mid \exists (z, f)^T \in \tilde{Z}\}.$$

4.  $k_e := k_e + 1$

Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $\mathcal{F}(k_e)$  für  $k_e = 0, 1, \dots$

Die Initialisierung des Algorithmus erfolgt mit dem kartesischen Produkt der Systemzustände und der Fehler, weil davon ausgegangen wird, dass sich der Prozess zum Zeitpunkt  $k_e = 0$  in einem beliebigen Zustand befinden kann und ein beliebiger Fehler aufgetreten sein kann.

## 16.3 Diagnose stochastischer Automaten

### 16.3.1 Beschreibung fehlerbehafteter Systeme durch stochastische Automaten

In Analogie zu den im Abschn. 16.2.1 eingeführten Modellen fehlerbehafteter Systeme muss für die Fehlerdiagnose die Verhaltensrelation stochastischer Automaten um den Eingang  $f$  erweitert werden, wodurch der Automat

$$\mathcal{S} = (\mathcal{Z}, \mathcal{V}, \mathcal{F}, \mathcal{W}, L, p_0(z))$$

entsteht. Die Verhaltensrelation  $L$  beschreibt jetzt den Zustandsübergang unter der zusätzlichen Wirkung des Fehlers  $f$ :

$$L(z', w | z, v, f) = \text{Prob}(Z(k+1)=z', W(k)=w | Z(k)=z, V(k)=v, F(k)=f).$$

Als Fehlermodell wird ein autonomer stochastischer Automat

$$\mathcal{S}_F = (\mathcal{F}, G_F, p_F(f))$$

verwendet, dessen Zustandsübergangswahrscheinlichkeitsverteilung

$$G_F(f' | f) = \text{Prob}(F(k+1) = f' | F(k) = f)$$

die bedingte Wahrscheinlichkeit dafür angibt, dass der Fehler zum Zeitpunkt  $k+1$  den Wert  $f'$  hat, wenn er zum Zeitpunkt  $k$  den Wert  $f$  hatte.

Fasst man den Automaten  $\mathcal{S}$  und das Fehlermodell  $\mathcal{S}_F$  zusammen (vgl. Abb. 16.1 auf S. 527), so erhält man den stochastischen Automaten

$$\tilde{\mathcal{S}} = (\tilde{\mathcal{Z}}, \mathcal{V}, \mathcal{W}, \tilde{L}, p_0(\tilde{z})) \quad (16.12)$$

mit dem Zustand

$$\tilde{z} = \begin{pmatrix} z \\ f \end{pmatrix} \in \mathcal{Z} \times \mathcal{F}$$

und der Verhaltensrelation

$$\tilde{L} \left( \begin{pmatrix} z' \\ f' \end{pmatrix}, w \middle| \begin{pmatrix} z \\ f \end{pmatrix}, v \right) = L(z', w | z, v, f) \cdot G(f' | f). \quad (16.13)$$

Entsprechend der Abbildung ist der Zustandsübergang des Fehlermodells nicht vom Prozesszustand  $z$  abhängig. Wenn dies in bestimmten Anwendungen der Fall ist, weil beispielsweise ein Zustand  $z$  eine Überlastung des Prozesses beschreibt, die das Auftreten eines Fehlers provozieren kann, dann muss die Modellstruktur und damit das Modell des zu diagnostizierenden Prozesses einschließlich des Fehlermodells erweitert werden.



### 16.3.2 Grundidee der Diagnose stochastischer Systeme

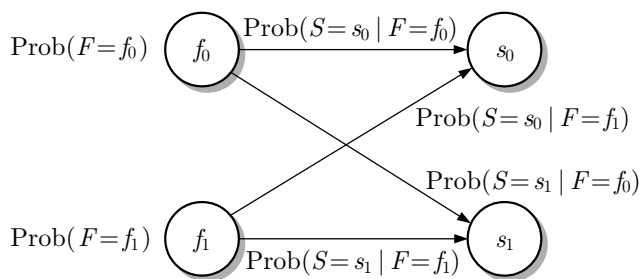
Der im nachfolgenden Abschn. 16.3.3 angegebene Diagnosealgorithmus enthält umfangreiche Rechnungen mit Wahrscheinlichkeitsverteilungen, die schwer durchschaubar erscheinen, jedoch auf einem einfachen Grundprinzip beruhen. Bevor dieser Algorithmus erläutert wird, soll hier zunächst das Diagnoseprinzip beschrieben werden.

Es wird ein System betrachtet, das in den beiden Fehlerzuständen  $f_0$  und  $f_1$  sein kann, wobei  $f_0$  wieder den fehlerfreien Betrieb kennzeichnet. Als Information über den Fehlerzustand wird eine Größe gemessen, die die Werte  $s_0$  und  $s_1$  annehmen kann. Das Diagnoseproblem wäre sehr einfach, wenn es eine eindeutige Zuordnung des Messwertes  $s_0$  zum fehlerfreien Zustand  $f_0$  und des Messwertes  $s_1$  zum Fehlerzustand  $f_1$  gäbe, denn dann wüsste man beim Messwert  $s_0$ , dass das System fehlerfrei arbeitet, und beim Messwert  $s_1$ , dass der Fehler  $f_1$  vorliegt. Die beiden Messwerte charakterisieren die beiden Fehlerzustände jedoch nur mit einer bestimmten Wahrscheinlichkeit, wobei beide Messwerte in beiden Fehlerzuständen auftreten können. Die Ursache dafür kann vielfältig sein. Man denke beispielsweise daran, dass der Messwert durch äußere Einflüsse, die nichts mit dem Fehler zu tun haben, verfälscht wird. An Stelle von Messwerten spricht man bei Diagnoseaufgaben dieses Typs deshalb auch von Symptomen. In der Bedeutung dieses Begriffs ist eingeschlossen, dass das Auftreten eines Symptoms den Fehlerzustand nicht eindeutig beschreibt.

Solange man nichts über die Häufigkeit des Auftretens der beiden Symptome in beiden Fehlerfällen weiß, kann man nach der Messung eines der beiden Symptome nichts über den Fehlerzustand aussagen. Diese Situation ändert sich, wenn man weiß, dass das Symptom  $s_0$  mit größerer Wahrscheinlichkeit im fehlerfreien Zustand als im Fehlerzustand auftritt und das Symptom  $s_1$  mit größerer Wahrscheinlichkeit den Fehlerzustand anzeigt als den fehlerfreien Zustand. Diese Informationen stecken in der bedingten Wahrscheinlichkeitsverteilung der beiden stochastischen Größen  $S$  und  $F$ , also in den vier bedingten Wahrscheinlichkeiten

$$\begin{aligned} &\text{Prob}(S = s_0 \mid F = f_0), \quad \text{Prob}(S = s_0 \mid F = f_1), \\ &\text{Prob}(S = s_1 \mid F = f_0), \quad \text{Prob}(S = s_1 \mid F = f_1), \end{aligned} \quad (16.14)$$

von denen die erste angibt, mit welcher Wahrscheinlichkeit die Messung den Wert  $s_0$  ergibt, wenn sich das System im fehlerfreien Zustand  $f_0$  befindet.



**Abb. 16.8:** Bayesnetz für zwei Fehler und zwei Symptome

Man kann den Zusammenhang zwischen den Fehlerzuständen und den Symptomen wie in Abb. 16.8 grafisch darstellen, wobei die linken Knoten die beiden möglichen Werte für den Fehlerzustand  $F$  und die beiden rechten Knoten die Werte der Messgröße  $S$  symbolisieren. Die vier angegebenen Wahrscheinlichkeiten sind an die vier gerichteten Kanten angetragen. Der als *Bayesnetz* bezeichnete Graph ist das Modell des zu diagnostizierenden Systems.

Um den Diagnosealgorithmus zu verstehen, wird jetzt angenommen, dass der Messwert  $s_0$  vorliegt und die Frage zu beantworten ist, mit welcher Wahrscheinlichkeit das System fehlerfrei arbeitet. Es ist also die bedingte Wahrscheinlichkeit

$$\text{Prob}(F = f_0 \mid S = s_0) \quad (16.15)$$

zu berechnen.

Um diese Wahrscheinlichkeit zu bestimmen, sei zunächst an die Definition der bedingten Wahrscheinlichkeit erinnert, die für die hier betrachtete Situation durch

$$\text{Prob}(F = f_0 \mid S = s_0) = \frac{\text{Prob}(F = f_0, S = s_0)}{\text{Prob}(S = s_0)} \quad (16.16)$$

beschrieben ist. Dabei ist  $\text{Prob}(F = f_0, S = s_0)$  die gemeinsame Wahrscheinlichkeitsverteilung der stochastischen Größen  $F$  und  $S$ . Analog dazu gilt

$$\text{Prob}(S = s_0 \mid F = f_0) = \frac{\text{Prob}(F = f_0, S = s_0)}{\text{Prob}(F = f_0)}$$

und die „Multiplikationsregel“

$$\text{Prob}(F = f_0, S = s_0) = \text{Prob}(S = s_0 \mid F = f_0) \cdot \text{Prob}(F = f_0).$$

In Gl. (16.16) eingesetzt ergibt sich

$$\text{Prob}(F = f_0 \mid S = s_0) = \frac{\text{Prob}(S = s_0 \mid F = f_0) \cdot \text{Prob}(F = f_0)}{\text{Prob}(S = s_0)}$$

und

$$\begin{aligned} \text{Prob}(F = f_0 \mid S = s_0) = & \quad (16.17) \\ & \frac{\text{Prob}(S = s_0 \mid F = f_0) \cdot \text{Prob}(F = f_0)}{\text{Prob}(S = s_0 \mid F = f_0) \cdot \text{Prob}(F = f_0) + \text{Prob}(S = s_0 \mid F = f_1) \cdot \text{Prob}(F = f_1)}, \end{aligned}$$

wobei die letzte Umformung hier gilt, weil es nur zwei Fehlerfälle gibt und man die Wahrscheinlichkeit  $\text{Prob}(S = s_0)$  deshalb entsprechend

$$\begin{aligned} \text{Prob}(S = s_0) = & \\ & \text{Prob}(S = s_0 \mid F = f_0) \cdot \text{Prob}(F = f_0) + \text{Prob}(S = s_0 \mid F = f_1) \cdot \text{Prob}(F = f_1) \end{aligned}$$

aus den beiden rechts stehenden bedingten Wahrscheinlichkeiten und den A-priori-Wahrscheinlichkeiten  $\text{Prob}(F = f_0)$  und  $\text{Prob}(F = f_1)$  berechnen kann. Die Beziehung (16.17) ist eine

Anwendung der *Bayes-Formel* der Wahrscheinlichkeitsrechnung. Für andere Argumente ergeben sich analoge Beziehungen wie beispielsweise

$$\text{Prob}(F = f_1 | S = s_0) = \frac{\text{Prob}(S = s_0 | F = f_1) \cdot \text{Prob}(F = f_1)}{\text{Prob}(S = s_0 | F = f_0) \cdot \text{Prob}(F = f_0) + \text{Prob}(S = s_0 | F = f_1) \cdot \text{Prob}(F = f_1)}.$$

Das Bayesnetz in Abb. 16.8 ist in der Pfeilrichtung ein Modell, das angibt, wie die Symptome durch die Fehlerzustände hervorgerufen werden. Es zeigt beispielsweise, mit welcher Wahrscheinlichkeit der Fehlerzustand  $f_0$  das Symptom  $s_0$  nach sich zieht. Mit der Bayesformel in der oben angegebenen Form wird das Umgekehrte berechnet, nämlich die Wahrscheinlichkeit  $\text{Prob}(F = f_0 | S = s_0)$  dafür, dass der Fehlerzustand  $f_0$  vorhanden ist, wenn der Messwert  $s_0$  auftritt. Für die Anwendung ist wichtig, dass in der Beziehung (16.17) die in Ursache-Wirkungsrichtung geltenden Größen eingehen, die an den Pfeilen des Bayesnetzes stehen, obwohl bei der Diagnose in umgekehrter Richtung vom Symptom auf den Fehler geschlussfolgert wird.

**Abhängigkeit des Diagnoseergebnisses von den A-priori-Informationen.** Wichtig ist außerdem die Abhängigkeit des Ergebnisses  $\text{Prob}(F = f_0 | S = s_0)$  von der A-priori-Wahrscheinlichkeitsverteilung  $\text{Prob}(F = f_0)$ ,  $\text{Prob}(F = f_1)$ . Um die Diagnoseaufgabe lösen zu können, muss man also wissen, mit welcher Wahrscheinlichkeit der Fehlerzustand überhaupt möglich ist, d. h., bei wie vielen Anwendungen des Diagnosealgorithmus sich das System im Fehlerzustand  $f_0$  bzw.  $f_1$  befindet. In Abb. 16.8 sind diese Wahrscheinlichkeiten dem Knoten  $f_0$  bzw.  $f_1$  zugeordnet.

Da es in technischen Anwendungen schwierig ist, eine Aussage über die A-priori-Wahrscheinlichkeit für das Auftreten des Fehlers zu machen, soll die Notwendigkeit dieser Angabe für das Diagnoseergebnis demonstriert werden. Angenommen, man wüsste mit Sicherheit, dass der Fehlerzustand ausgeschlossen ist. Dann sähe die A-priori-Wahrscheinlichkeitsverteilung folgendermaßen aus

$$\begin{aligned} \text{Prob}(F = f_0) &= 1 \\ \text{Prob}(F = f_1) &= 0 \end{aligned} \tag{16.18}$$

und man erhielte aus Gl. (16.17)

$$\begin{aligned} &\text{Prob}(F = f_0 | S = s_0) \\ &= \frac{\text{Prob}(S = s_0 | F = f_0) \cdot \text{Prob}(F = f_0)}{\text{Prob}(S = s_0 | F = f_0) \cdot \text{Prob}(F = f_0) + \text{Prob}(S = s_0 | F = f_1) \cdot \text{Prob}(F = f_1)} \\ &= \frac{\text{Prob}(S = s_0 | F = f_0) \cdot 1}{\text{Prob}(S = s_0 | F = f_0) \cdot 1 + \text{Prob}(S = s_0 | F = f_1) \cdot 0} \\ &= \frac{\text{Prob}(S = s_0 | F = f_0)}{\text{Prob}(S = s_0 | F = f_0)} \\ &= 1 \end{aligned}$$

und

$$\begin{aligned}
 & \text{Prob}(F=f_0 | S=s_1) \\
 &= \frac{\text{Prob}(S=s_1 | F=f_0) \cdot \text{Prob}(F=f_0)}{\text{Prob}(S=s_1 | F=f_0) \cdot \text{Prob}(F=f_0) + \text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)} \\
 &= \frac{\text{Prob}(S=s_1 | F=f_0) \cdot 1}{\text{Prob}(S=s_1 | F=f_0) \cdot 1 + \text{Prob}(S=s_1 | F=f_1) \cdot 0} \\
 &= \frac{\text{Prob}(S=s_1 | F=f_0)}{\text{Prob}(S=s_1 | F=f_0)} = 1.
 \end{aligned}$$

Das heißt, ob man den Messwert  $s_0$  oder  $s_1$  erhält ist gleichgültig, denn in beiden Fällen ergibt die Diagnose, dass sich das System im fehlerfreien Zustand befindet. Dieses Ergebnis ist unabhängig von den bedingten Wahrscheinlichkeiten (16.14) – es ist eine Konsequenz aus der Annahme (16.18), dass das System gar nicht im Fehlerzustand sein kann.

Nur wenn man beide Fehlerzustände zulässt

$$\begin{aligned}
 \text{Prob}(F=f_0) &> 0 \\
 \text{Prob}(F=f_1) &> 0,
 \end{aligned}$$

kann das System im Fehlerzustand sein und man erhält aus der Diagnoseformel (16.17) eine positive Wahrscheinlichkeit dafür, dass der Fehler aktuell aufgetreten ist.

Die berechnete Wahrscheinlichkeit  $\text{Prob}(F=f_0 | S=s_0)$  heißt A-posteriori-Wahrscheinlichkeit, denn sie ist diejenige Wahrscheinlichkeit, die man mit Hilfe der durch den Messwert  $S=s_0$  erhaltenen Zusatzinformation („nach dem Experiment“) berechnet. Im Gegensatz dazu beschreibt die A-priori-Wahrscheinlichkeit  $\text{Prob}(F=f_0)$  diejenige Wahrscheinlichkeit, die „vor allen Experimenten“ gilt.

Für die Bewertung des Diagnoseergebnisses ist nicht der Wert der A-posteriori-Wahrscheinlichkeit  $\text{Prob}(F=f_0 | S=s_0)$  allein maßgebend, sondern die Veränderung von  $\text{Prob}(F=f_0)$  zu  $\text{Prob}(F=f_0 | S=s_0)$ , die die Messgröße bewirkt. Diese Veränderung ist typischerweise weit weniger von den konkreten Werten der verwendeten A-priori-Wahrscheinlichkeitsverteilung abhängig als die Werte der A-posteriori-Wahrscheinlichkeitsverteilung selbst. Beispielsweise ist  $\text{Prob}(F=f_0 | S=s_0)$  deutlich größer als  $\text{Prob}(F=f_0)$ , wenn die Wahrscheinlichkeit  $\text{Prob}(S=s_0 | F=f_0)$  für das Auftreten des Messwertes  $s_0$  im fehlerfreien Zustand groß ist.

### Beispiel 16.2 Diagnose eines Motorkühlsystems

Pkw-Motoren sind mit einem Thermostaten (Temperaturregler) ausgestattet, der die Motortemperatur weitgehend unabhängig von der Belastung des Motors auf einem konstanten Sollwert hält. Aus Sicherheitsgründen überwacht man die Motortemperatur außerdem mit einem Sensor, der bei einer Grenzwertüberschreitung seines Temperaturwertes einen Alarm auslöst. Da der Fehlerfall typischerweise sehr selten vorkommt (z. B.  $\text{Prob}(F=f_1) = 0,001$ ) und demzufolge erst nach langer Motorlaufzeit eine Alarmmeldung auftritt, ist die Zuverlässigkeit des Überwachungssensors wichtig für die Wahrscheinlichkeit, mit der im Falle einer Alarmmeldung das Kühlsystem tatsächlich fehlerhaft ist. Da auch der

Sensor bis zur ersten Alarmmeldung fehlerhaft sein kann und z. B. auf Grund von Verschmutzungen oder Kabelbruch die Grenzwertüberschreitung nicht signalisiert, wird mit den bedingten Wahrscheinlichkeiten  $\text{Prob}(S=s_1 | F=f_1) = 0,995$  und  $\text{Prob}(S=s_0 | F=f_0) = 0,9995$  gerechnet.

Aus diesen Angaben folgen weitere Werte der Wahrscheinlichkeitsverteilungen:

$$\begin{aligned}\text{Prob}(F=f_0) &= 0,999 \\ \text{Prob}(S=s_0 | F=f_1) &= 0,005 \\ \text{Prob}(S=s_1 | F=f_0) &= 0,0005.\end{aligned}$$

Gesucht ist die Wahrscheinlichkeit, mit der ein Fehler im Kühlsystem aufgetreten ist, wenn der Überwachungssensor einen Alarm auslöst.

Aus Gl. (16.17) erhält man

$$\begin{aligned}\text{Prob}(F=f_1 | S=s_1) &= \frac{\text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)}{\text{Prob}(S=s_1 | F=f_0) \cdot \text{Prob}(F=f_0) + \text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)} \\ &= \frac{0,995 \cdot 0,001}{0,0005 \cdot 0,999 + 0,995 \cdot 0,001} \\ &= \frac{0,000995}{0,0014945} \\ &= 0,666.\end{aligned}$$

Das Kühlsystem ist also im Falle einer Alarmmeldung nur mit einer Wahrscheinlichkeit von etwa 67% fehlerhaft. Durch die Alarmmeldung ist die Wahrscheinlichkeit für den Fehlerfall allerdings sehr stark gestiegen, wie das Verhältnis der A-posteriori-Wahrscheinlichkeit zur A-priori-Wahrscheinlichkeit zeigt:

$$\frac{\text{Prob}(F=f_1 | S=s_1)}{\text{Prob}(F=f_1)} = \frac{0,666}{0,001} = 666.$$

Um die Abhängigkeit des Diagnoseergebnisses von der A-priori-Wahrscheinlichkeit zu zeigen, wird jetzt mit folgenden veränderten Werten gerechnet

$$\begin{aligned}\text{Prob}(F=f_0) &= 0,9999 \\ \text{Prob}(F=f_1) &= 0,0001,\end{aligned}$$

bei der der Fehlerfall a-priori um eine Zehnerpotenz unwahrscheinlicher ist als bisher. Dann erhält man

$$\begin{aligned}\text{Prob}(F=f_1 | S=s_1) &= \frac{\text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)}{\text{Prob}(S=s_1 | F=f_0) \cdot \text{Prob}(F=f_0) + \text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)} \\ &= \frac{0,995 \cdot 0,0001}{0,0005 \cdot 0,9999 + 0,995 \cdot 0,0001} \\ &= \frac{0,0000995}{0,00059945} \\ &= 0,166.\end{aligned}$$

Da die A-priori-Wahrscheinlichkeit für einen Fehler im Kühlsystem jetzt deutlich kleiner ist, ist auch die Wahrscheinlichkeit dafür, dass dieser Fehler die Alarmmeldung ausgelöst hat, gesunken. Die A-posteriori-Wahrscheinlichkeit ist jedoch 1660 mal so groß wie die A-priori-Wahrscheinlichkeit

$$\frac{\text{Prob}(F=f_1 | S=s_1)}{\text{Prob}(F=f_1)} = \frac{0,166}{0,0001} = 1660,$$

also noch stärker angestiegen als bei der ersten Rechnung.

Wenn das Messgerät zuverlässiger wird, so wird (erwartungsgemäß) auch das Diagnoseergebnis besser. Zum Vergleich zur ersten Rechnung wird die Diagnose jetzt mit folgenden Werten wiederholt:

$$\text{Prob}(S=s_0 | F=f_0) = 0,9995$$

$$\text{Prob}(S=s_0 | F=f_1) = 0,0005$$

$$\text{Prob}(S=s_1 | F=f_0) = 0,00005$$

$$\text{Prob}(S=s_1 | F=f_1) = 0,99995.$$

Man erhält

$$\begin{aligned} & \text{Prob}(F=f_1 | S=s_1) \\ &= \frac{\text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)}{\text{Prob}(S=s_1 | F=f_0) \cdot \text{Prob}(F=f_0) + \text{Prob}(S=s_1 | F=f_1) \cdot \text{Prob}(F=f_1)} \\ &= \frac{0,9995 \cdot 0,001}{0,00005 \cdot 0,999 + 0,9995 \cdot 0,001} \\ &= \frac{0,0009995}{0,00104945} \\ &= 0,952, \end{aligned}$$

also aufgrund der verbesserten Zuverlässigkeit des Sensors eine fast eindeutige Anzeige des Fehlerzustands durch die Alarmmeldung.  $\square$

#### **Aufgabe 16.4** Überwachung eines automatischen Qualitätskontrollsystems

Die automatische Qualitätskontrolle beinhaltet schwierige Mess- und Bewertungsaufgaben, weil Qualitätsmängel in einem Produkt häufig nicht einer direkten Messung zugänglich sind. So muss bei einer Flaschenabfüllung der Füllstand in den verschlossenen Flaschen überprüft werden, was nur indirekt – beispielsweise durch die Messung des Flaschengewichts – erfolgen kann. Bei der Kontrolle von Textilien auf Webfehler ist die Messaufgabe noch wesentlich schwieriger. Aus diesem Grunde ist das Ergebnis der Qualitätskontrolle mit Unsicherheiten verbunden, die durch Wahrscheinlichkeitsverteilungen beschrieben werden können.

Es wird jetzt ein Qualitätskontrollsystem betrachtet, bei dem der Messwert  $W = 1$  ein qualitativ einwandfreies Produkt und der Messwert  $W = 0$  Qualitätsmängel anzeigt, wobei der Produktzustand mit  $Z = 1$  bzw.  $Z = 0$  bezeichnet wird. Die A-priori-Wahrscheinlichkeit für einen Qualitätsmangel betrage 0,0001. Es sollen drei Qualitätskontrollsysteme verglichen werden, die sich in der bedingten Wahrscheinlichkeit, dass das Kontrollsystem einen Fehler unter der Bedingung anzeigt, dass das Produkt fehlerhaft ist, um jeweils eine Zehnerpotenz unterscheiden:

$$\text{Prob}(W = 1 | Z = 1) = \text{Prob}(W = 0 | Z = 0) = 0,001 \text{ oder } 0,0001 \text{ oder } 0,00001.$$

Vergleichen Sie die Wahrscheinlichkeiten für das Nichterkennen eines Produktionsfehlers bzw. für die fehlerhafte Anzeige eines Fehlers durch die drei Kontrollsysteme.  $\square$

### 16.3.3 Diagnosealgorithmus

Dieser Algorithmus nutzt die für Bayesnetze erläuterte Diagnosemethode für stochastische Automaten, indem die Diagnoseaufgabe durch Anwendung des Beobachtungsalgorithmus 15.2 auf den Automaten (16.12) gelöst wird. Dabei wird zunächst die Verbundwahrscheinlichkeit

$$\begin{aligned} \text{Prob}(Z(k_e) = z, F(k_e) = f \mid V(0) = v_0, V(1) = v_1, \dots, V(k_e) = v_k, \\ W(0) = w_0, W(1) = w_1, \dots, W(k_e) = w_k), \end{aligned}$$

die vereinfacht durch

$$\text{Prob}(Z(k_e) = z, F(k_e) = f \mid V(0 \dots k_e), W(0 \dots k_e))$$

abgekürzt wird, berechnet und daraus als Randverteilung die Wahrscheinlichkeit für die Fehler bestimmt:

$$\begin{aligned} \text{Prob}(F(k_e) = f \mid V(0 \dots k_e), W(0 \dots k_e)) \\ = \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k_e) = z, F(k_e) = f \mid V(0 \dots k_e), W(0 \dots k_e)). \end{aligned}$$

Für die Formulierung des Diagnosealgorithmus werden wieder drei Wahrscheinlichkeitsverteilungen gebraucht, für die die folgenden Abkürzungen eingeführt werden:

$$\begin{aligned} p'(z', f', k_e) &= \text{Prob}(Z(k_e) = z', F(k_e) = f' \mid V(0 \dots k_e - 1), W(0 \dots k_e - 1)) \\ p(z, f, k_e) &= \text{Prob}(Z(k_e) = z, F(k_e) = f \mid V(0 \dots k_e), W(0 \dots k_e)). \end{aligned}$$

Aus den Gln. (15.14) erhält man dafür die rekursive Beziehung

$$p(z, f, k_e) = \frac{\sum_{z', f'} L(z', \bar{w} \mid z, \bar{v}, \bar{f}) \cdot G_F(f' \mid \bar{f}) \cdot p'(z', f', k_e - 1)}{\sum_{z, z', f, f'} L(z', \bar{w} \mid z, \bar{v}, f) \cdot G_F(f' \mid f) \cdot p'(z, f, k_e - 1)} \quad (16.19)$$

$$p'(z', f', k_e) = \frac{\sum_{z, f} L(z', \bar{w} \mid z, \bar{v}, f) \cdot G_F(\bar{f}' \mid f) \cdot p'(z, f, k_e - 1)}{\sum_{z, z', f, f'} L(z', \bar{w} \mid z, \bar{v}, f) \cdot G_F(f' \mid f) \cdot p'(z, f, k_e - 1)}. \quad (16.20)$$

Für die Fehlerwahrscheinlichkeit, für die die Abkürzung

$$p(f, k_e) = \text{Prob}(F(k_e) = f \mid V(0 \dots k_e), W(0 \dots k_e))$$

verwendet wird, erhält man dann

$$p(f, k_e) = \sum_{z \in \mathcal{Z}} \text{Prob}(Z(k_e) = z, f(k_e) = \bar{f} \mid k_e) = \sum_{z \in \mathcal{Z}} p(f, z, k_e). \quad (16.21)$$

---

**Algorithmus 16.4** *Diagnose stochastischer Automaten*


---

**Gegeben:** Automat mit der Verhaltensrelation  $L(z', w \mid z, v, f)$   
 Fehlermodell mit Übergangswahrscheinlichkeit  $G_F(f' \mid f)$   
 A-priori-Wahrscheinlichkeitsverteilung  $p_0(z) = \text{Prob}(Z(0) = z)$   
 A-priori-Fehlerwahrscheinlichkeitsverteilung  $p_F(f) = \text{Prob}(F(0) = f)$ .

**Initialisierung:**  $p'(z, f, -1) := p_0(z) \cdot p_F(f)$  für  $z \in \mathcal{Z}, \bar{f} \in \mathcal{F}$   
 $k_e = 0$

**Schleife:**

1. Warten Sie auf die nächsten Messwerte  $\bar{v}, \bar{w}$ .
2. Bestimmen Sie die Wahrscheinlichkeitsverteilungen  
 $p(z, f, k_e)$  für alle  $z, f$  mit Gl. (16.19)  
 $p'(z', f', k_e)$  für alle  $z', \bar{f}'$  mit Gl. (16.20).
3. Bestimmen Sie die Fehlerwahrscheinlichkeit  $p(f, k_e)$  mit Gl. (16.21)  
 sowie die Menge der Fehlerkandidaten  $\mathcal{F}(k_e) = \{f \mid p(f, k_e) > 0\}$ .
4.  $k_e := k_e + 1$   
 Setzen Sie den Algorithmus mit Schritt 1 fort.

**Ergebnis:**  $\mathcal{F}(k_e)$  und Fehlerwahrscheinlichkeitsverteilung  $p(f, k_e)$  für  $k_e = 0, 1, \dots$

---

Der Algorithmus erfordert die Vorgabe einer A-priori-Fehlerwahrscheinlichkeitsverteilung, bei der man beispielsweise von der Annahme ausgehen kann, dass das System fehlerfrei arbeitet

$$\text{Prob}(F(0) = f) = \begin{cases} 1 & \text{für } f = f_0 \\ 0 & \text{sonst} \end{cases}$$

oder eine Gleichverteilung über alle Fehler ansetzen kann

$$\text{Prob}(F(0) = f) = \frac{1}{q}, \quad f \in \mathcal{F},$$

bei der  $q$  die Anzahl der betrachteten Fehler ist. Obwohl der Fehler durch die erste Annahme auf  $f_0$  festgelegt ist, kann er für Zeithorizonte  $k_e > 0$  erkannt werden, weil das Fehlermodell eine zeitliche Veränderung des Fehlers  $f(k)$  zulässt. Wenn man jedoch mit einem konstanten Fehler arbeitet

$$G_F(f' \mid f) = \begin{cases} 1 & \text{für } f' = f \\ 0 & \text{sonst,} \end{cases}$$

dann müssen alle Fehler mit einer positiven A-priori-Wahrscheinlichkeit  $\text{Prob}(F(0) = f)$  angesetzt werden. Wenn Informationen über den Anfangszustand  $z(0)$  vorliegen, können diese bei der Initialisierung des Algorithmus durch eine entsprechende Festlegung von  $p_0(z) = \text{Prob}(Z(0) = z)$  berücksichtigt werden.



**Diagnoseergebnis.** Der Algorithmus liefert folgende Aussagen für die Fehlerdiagnose:

- **Fehlererkennung:** Wenn  $f_0 \notin \mathcal{F}(k_e)$  gilt, wobei  $f_0$  den fehlerfreien Zustand bezeichnet, so ist erwiesen, dass der Automat fehlerbehaftet ist.
- **Fehleridentifikation:** Der zur Zeit  $k_e$  vorhandene Fehler gehört zur Menge der Fehlerkandidaten

$$f \in \mathcal{F}(k_e).$$

Wenn  $\mathcal{F}(k_e) = \{f_i\}$  eine einelementige Menge ist, so ist erwiesen, dass der Fehler  $f = f_i$  aufgetreten ist (vorausgesetzt, dass  $f$  zum betrachteten Fehlerraum  $\mathcal{F}$  gehört). Wenn  $\mathcal{F}(k_e)$  mehrere Elemente enthält, so gibt  $p(f, k_e)$  die Wahrscheinlichkeit dafür an, dass der Fehler  $f$  aufgetreten ist.

### Beispiel 16.3 Diagnose eines stochastischen Automaten

Der in Abb. 16.9 gezeigte Automat mit dem Zustand  $\tilde{Z} = (z, f)^T$  ist für die beiden Fehlerfälle  $f = 0$  und  $f = 1$  angegeben. Der Anfangszustand  $z_0 = 1$  sei bekannt, weshalb für die Anfangszustandswahrscheinlichkeitsverteilung die Beziehung

$$\text{Prob}(Z(0) = z) = \begin{cases} 1 & \text{für } z = 1 \\ 0 & \text{sonst} \end{cases}$$

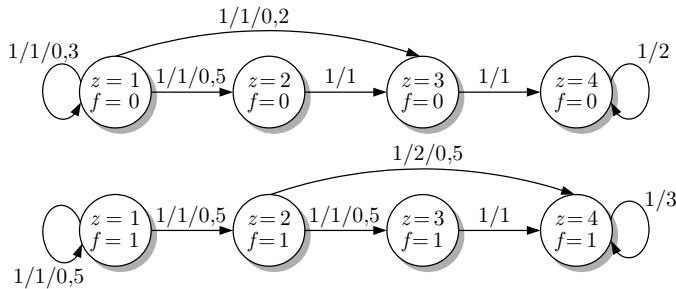
gilt. Für den Fehler wird a-priori eine Gleichverteilung angenommen:

$$\text{Prob}(F(0) = f) = 0,5 \quad \text{für } f = 0, 1.$$

Das heißt, dass der Algorithmus mit der A-priori-Wahrscheinlichkeit

$$\text{Prob} \left( \begin{pmatrix} Z(0) \\ F(0) \end{pmatrix} = \begin{pmatrix} z \\ f \end{pmatrix} \right) = \begin{cases} 0,5 & \text{für } z = 1 \\ 0 & \text{sonst} \end{cases}$$

arbeitet.



**Abb. 16.9:** Beschreibung des fehlerfreien und des fehlerbehafteten Systems

**Tabelle 16.1.** Wahrscheinlichkeitsverteilung des Fehlers  $\text{Prob}(f(k_e) | k_e)$ .

	$k_e = 0$ $V(0...0) = (1)$ $W(0...0) = (1)$	$k_e = 1$ $V(0...1) = (1, 1)$ $W(0...1) = (1, 1)$	$k_e = 2$ $V(0...2) = (1, 1, 1)$ $W(0...2) = (1, 1, 1)$
$f$	$p(f, 0)$	$p(f, 1)$	$p(f, 2)$
0	0,5	0,5714	0,5614
1	0,5	0,4286	0,4386

	$k_e = 3$ $V(0...3) = (1, 1, 1, 1)$ $W(0...3) = (1, 1, 1, 3)$	$k_e = 4$ $V(0...3) = (1, 1, 1, 1, 1)$ $W(0...3) = (1, 1, 1, 3, 3)$
$f$	$p(f, 3)$	$p(f, 4)$
0	0	0
1	1	1

Tabelle 16.1 zeigt das Diagnoseergebnis für das E/A-Paar

$$V(0...3) = (1, 1, 1, 1)$$

$$W(0...3) = (1, 1, 1, 3, 3).$$

Für den Anfangszustand  $z_0 = 1$  liefert das erste E/A-Paar ( $V(0) = 1, W(0) = 1$ ) keine Informationen über den Fehler, weil der Automat bei der Eingabe  $\bar{v} = 1$  für beide Fehler die Ausgabe  $\bar{w} = 1$  mit der Wahrscheinlichkeit 1 erzeugt. Zum nächsten Zeitpunkt  $k_e = 1$  kann das System in folgenden Zuständen sein:

$$\begin{pmatrix} z \\ f \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 3 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix} \text{ oder } \begin{pmatrix} 2 \\ 2 \end{pmatrix}.$$

Das E/A-Paar ( $V(1) = 1, W(1) = 1$ ) schließt den Übergang

$$\begin{pmatrix} 2 \\ 2 \end{pmatrix} \rightarrow \begin{pmatrix} 4 \\ 2 \end{pmatrix}$$

aus. Deshalb ist der Fehler  $f = 1$  nach der zweiten Messung weniger wahrscheinlich als der Fehler  $f = 0$  (Abb. 16.9). Nach der Messung  $W(3) = 3$  ist der Fehler  $f = 1$  eindeutig gefunden.  $\square$

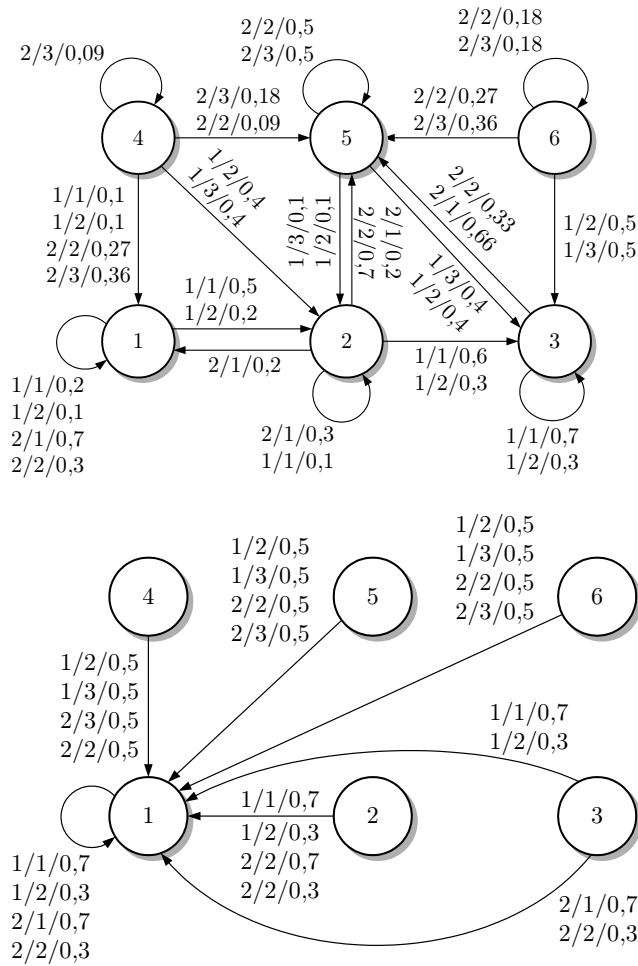
### Beispiel 16.1 (Forts.) Diagnose eines Batchprozesses

Für die Verbesserung des Diagnoseergebnisses für den Batchprozess werden die in Abb. 16.3 auf S. 532 gezeigten Automaten um Übergangswahrscheinlichkeiten erweitert, woraus die in Abb. 16.10 gezeigten Modelle entstehen.

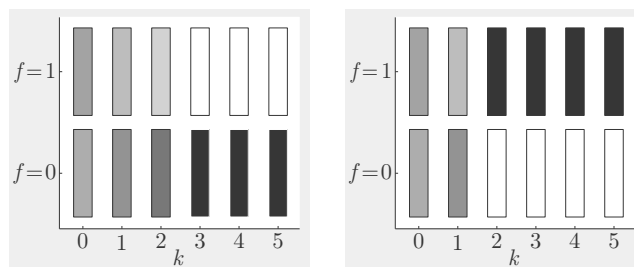
Für konstante Eingabe  $V = (2, 2, 2, 2, 2, 2)$  und die Ausgabefolge

$$W(0...5) = (3, 2, 2, 3, 3, 3) \quad (16.22)$$

erhält man das in Abb. 16.11 (links) dargestellte Ergebnis, bei dem die Fehlerwahrscheinlichkeiten durch Grautöne der betreffenden Rechtecke dargestellt sind. Der Fehler  $f = 1$  wird zunehmend unwahrscheinlicher, bis er nach der Messung zum Zeitpunkt  $k_e = 3$  ganz ausgeschlossen werden kann.



**Abb. 16.10:** Stochastischer Automat für den Fehler  $f = 0$  (oben) und den Fehler  $f = 1$  (unten)



**Abb. 16.11:** Diagnoseergebnis für die Ausgabefolge (16.22) (links) bzw. (16.23) (rechts)

Abbildung 16.11 (rechts) bezieht sich auf ein zweites Experiment, das mit dem fehlerbehafteten System durchgeführt wurde. Wiederum ist die Eingabe konstant ( $v = 2$ ), aber die Ausgabe

$$W(0\dots 5) = (3, 2, 1, 1, 1, 2) \quad (16.23)$$

zeigt einen abnehmenden Ausfluss aus dem rechten Behälter. Das Diagnoseergebnis weist nach drei Schritten eindeutig auf den Fehler hin.  $\square$

**Beziehung zwischen dem Diagnosealgorithmus und dem Bayesnetz.** Der beschriebene Diagnosealgorithmus verwendet die anhand von Abb. 16.8 beschriebene Grundidee der Diagnose von Systemen, deren Ursache-Wirkungsbeziehungen wahrscheinlichkeitstheoretisch beschrieben sind. Der einzige Unterschied liegt darin, dass sich jetzt der Zustand des betrachteten Systems verändert, wobei sich im Zustand  $(z, f)^T$  sowohl der Systemzustand als auch der Fehlerzustand ändern kann.

Abbildung 16.12 zeigt, dass man bei jedem Zeitschritt jetzt zunächst die Zustandsänderung betrachten muss. Die Ausgabe hängt dann von dem zum Zeitpunkt  $k + 1$  angenommenen Zustand ab. Das erweiterte Bayesnetz ist für ein autonomes System gezeichnet unter der Bedingung, dass es sich um einen Moore-Automat handelt, bei dem die Ausgabe nur vom aktuellen Zustand abhängt. Mit der Wahrscheinlichkeit

$$\text{Prob} \left( \tilde{Z}(k) = \begin{pmatrix} f_1 \\ z_1 \end{pmatrix} \middle| \tilde{Z}(k-1) = \begin{pmatrix} f_0 \\ z_0 \end{pmatrix} \right)$$

geht der Zustand des Automaten beim Zeitschritt  $k - 1 \rightarrow k$  von  $\tilde{Z}(k-1) = (f_0, z_0)^T$  in  $\tilde{Z}(k) = (f_1, z_1)^T$  über. In diesem Zustand wird mit der Wahrscheinlichkeit

$$\text{Prob} \left( W(k) = w_0 \middle| \tilde{Z}(k) = \begin{pmatrix} f_1 \\ z_1 \end{pmatrix} \right)$$

zum Zeitpunkt  $k$  die Ausgabe  $w_0$  erzeugt. Diese beiden Wahrscheinlichkeiten sind an zwei Kanten des gezeigten Bayesnetzes angetragen. Die anderen Kanten und ihre Bewertungen muss man dementsprechend interpretieren.

Mit dieser Darstellung ergeben sich die erweiterten Beziehungen, die in diesem Abschnitt abgeleitet wurden, in derselben Weise wie die in Gl. (16.17) angegebene, nur dass sie hier um die dynamischen Zustandsübergänge erweitert werden mussten und deshalb komplizierter sind. Die Abbildung lässt aber erkennen, wie sich das Grundprinzip im Diagnosealgorithmus wiederfindet.

#### **Aufgabe 16.5** *Diagnose eines Batchprozesses*

Berechnen Sie die Fehlerwahrscheinlichkeitsverteilung für den im Beispiel 16.1 betrachteten Batchprozess anhand der in Abb. 16.10 gezeigten Automaten. Verwenden Sie dieselben E/A-Paare wie im Beispiel und vergleichen Sie Ihre Ergebnisse mit den im Beispiel angegebenen.  $\square$

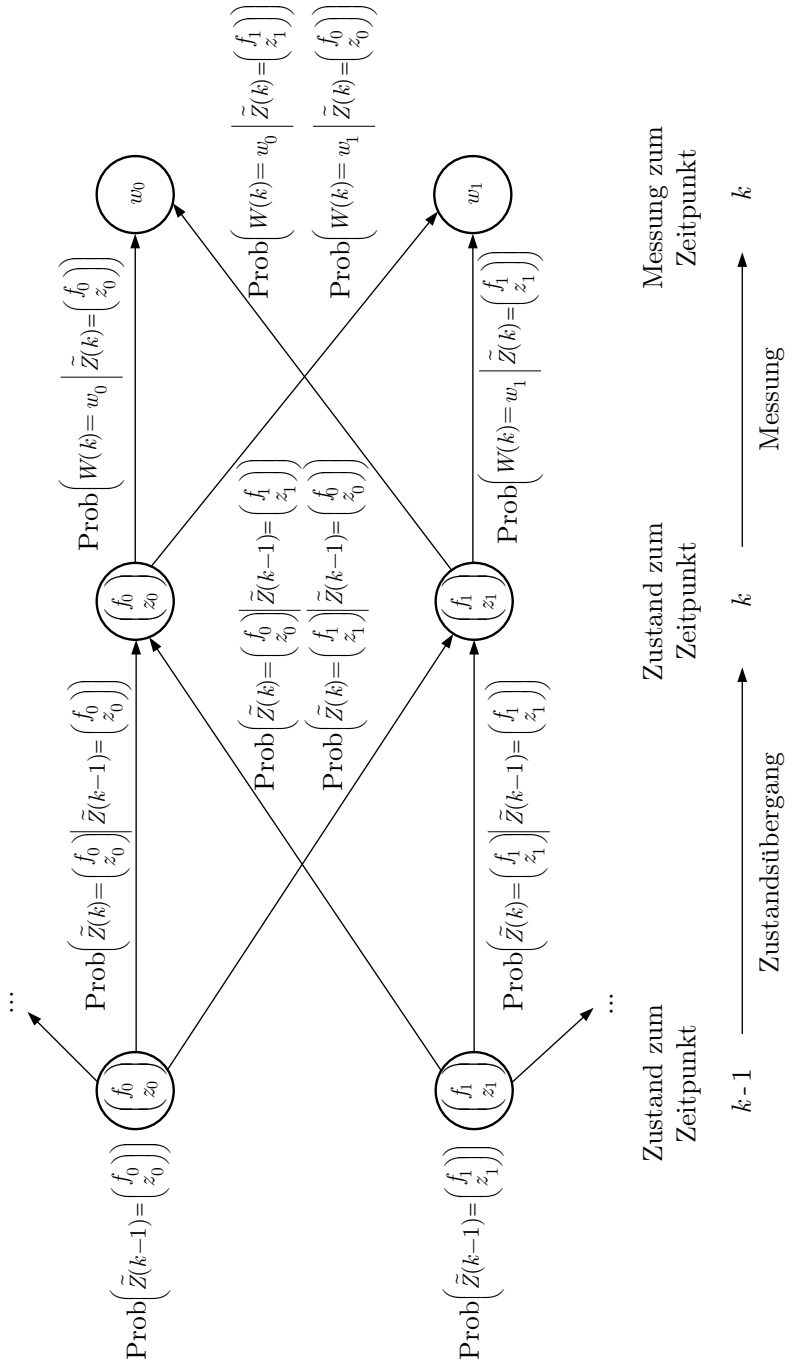
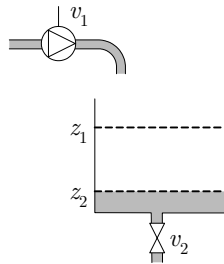


Abb. 16.12: Darstellung des zu diagnostizierenden Automaten als Bayesnetz

**Aufgabe 16.6\*** Modellierung eines Batchreaktors für die Fehlerdiagnose

Der in Abb. 16.13 gezeigte Reaktor wird innerhalb eines Batchprozesses eingesetzt, bei dem er mit einer Flüssigkeit gefüllt und später wieder entleert wird. Für das Verhalten des Reaktors wird deshalb nur zwischen einem leeren und einem vollen Reaktor unterschieden, wobei sich der Füllstand im ersten Fall unterhalb des unteren Schwellwertes und im zweiten Fall oberhalb des oberen Schwellwertes befindet. Diese beiden Zustände werden mit  $z_1$  und  $z_2$  bezeichnet. Das Füllen des Reaktors mit Hilfe einer Pumpe erfolgt nach der Eingabe  $v_1$ , die das Anschalten der Pumpe bewirkt, das Entleeren nach dem Öffnen des Ventils, das durch die Eingabe  $v_2$  ausgelöst wird. Eine hier nicht betrachtete Sicherheitsabschaltung der Pumpe verhindert, dass der Reaktor überläuft.


**Abb. 16.13:** Batchreaktor

1. Beschreiben Sie das diskrete Verhalten des Reaktors durch einen deterministischen Automaten.
2. Auf Grund des Fehlers  $f_1$  fällt die Pumpe aus. Erweitern Sie Ihr Modell so, dass es den Reaktor im fehlerfreien Betrieb und im fehlerbehafteten Betrieb beschreibt.
3. Der fehlerfreie Reaktor bleibt mit 99%-iger Wahrscheinlichkeit bei allen Zustandsübergängen fehlerfrei und geht mit 1%-iger Wahrscheinlichkeit in den Fehlerzustand über. Erweitern Sie Ihr Modell zu einem stochastischen Automaten, der diese Informationen wiedergibt.  $\square$

**Literaturhinweise**

Erste Arbeiten zur Diagnose ereignisdiskreter Systeme sind [61, 88, 97]. Die hier angegebenen Methoden, die auch mehrdeutige Lösungen für die Diagnoseaufgaben zulassen und für stochastische Automaten erweitert sind, wurden in [73] veröffentlicht.

Beispiel 16.1 ist [13] entnommen.

# 17

## Ausblick: Überwachung und Steuerung hybrider dynamischer Systeme

*Die bisher getrennt behandelten Methoden für die Automatisierung kontinuierlicher und diskreter Systeme müssen bei hybriden Prozessen miteinander kombiniert werden. Dieses Kapitel gibt einen Ausblick auf dieses aktuelle Forschungsthema.*

### 17.1 Automatisierung kontinuierlicher und ereignisdiskreter Systeme

Die in diesem Buch eingeführten Methoden der Automatisierungstechnik zeigen, dass es bezüglich der Aufgabenstellungen und der prinzipiellen Lösungswege wenige Unterschiede zwischen der Automatisierung wertkontinuierlicher und ereignisdiskreter Systeme gibt, dass sich jedoch die für die Lösung verwendeten mathematischen und systemtheoretischen Methoden für beide Systemklassen grundlegend unterscheiden. Die Ähnlichkeiten der Aufgabenstellungen werden an den im Kap. 1 gezeigten Blockschaltbildern offensichtlich.

Bei der Modellierung beider Systemklassen ist das Zustandsraumkonzept das verbindende Element. Es führt auf die sehr ähnlichen Darstellungsformen

$$\Sigma_{\text{NL}} = (\mathcal{X}, \mathcal{U}, \mathcal{Y}, g, h, x_0)$$

und

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, G, H, z_0),$$

die sich jedoch bezüglich ihrer mathematischen Repräsentation wesentlich unterscheiden. Der Grund liegt in den unterschiedlichen Eigenschaften beider Systemklassen, die für die kontinuierlichen Systeme eine Beschreibung durch algebraische Gleichungen und Differentialgleichungen zulassen, während man für ereignisdiskrete Systeme keine geschlossenen Ausdrücke für die Zustandsübergänge erhält und diese deshalb in umfangreichen Tabellen oder Grafiken darstellen muss. Diese Unterschiede in den Modellformen führen auf erhebliche Unterschiede in den Lösungswegen für alle Automatisierungsaufgaben.

Viele Beispiele dieses Buches zeigen, dass die Frage, ob ein technischer Prozess als kontinuierliches oder als ereignisdiskretes System behandelt werden muss, nicht nur von den Eigenschaften des Prozesses, sondern auch von den Automatisierungszielen abhängt. Wenn es um die Temperatur- oder Konzentrationsregelung eines Reaktors geht, muss der Reaktor als ein kontinuierliches System betrachtet werden, denn es kommt auf den quantitativ genauen Verlauf der betrachteten Größen an. Wenn derselbe Reaktor jedoch ein Element eines Batchprozesses ist, der gefüllt, erhitzt und wieder entleert wird, so ist es zweckmäßig, ihn als ereignisdiskretes System zu behandeln, bei dem nur zwischen dem gefüllten und dem entleerten sowie dem kalten und dem heißen Zustand unterschieden wird und die Vorgänge des Füllens, Erhitzens bzw. Entleerens als Schaltvorgang verkürzt dargestellt werden.

Der Übergang von der kontinuierlichen zur diskreten Betrachtungsebene ist stets mit einem Abstraktionsschritt verbunden, bei dem man bewusst auf Informationen verzichtet, um die wichtigsten Elemente und Eigenschaften eines Prozesses und einer Automatisierungsaufgabe gegenüber unwichtigen Elementen bzw. Eigenschaften herauszustellen. Bei der Behandlung des Reaktors als ereignisdiskretes System bedeutet diese Abstraktion, dass man dem kontinuierlichen Füllstand nur noch die diskreten Werte „voll“ und „leer“ zuweist. Bei der ereignisdiskreten Betrachtung wird also absichtlich auf Informationen verzichtet, die für die betrachtete Aufgabenstellung unwichtig sind, die jedoch bei anderen Aufgaben wie der Füllstandsregelung wichtig sein können. Hier zeigt sich ein allgemeines ingenieuertechnisches Prinzip, dass man bei der Modellbildung nicht nach der genauestmöglichen, sondern immer nach der der betrachteten Aufgabe angepassten Beschreibung eines Prozesses sucht.

Die Abstraktion kann auch andersherum von diskreten zu kontinuierlichen Werten führen. Ein Brenner, der aus technologischen Gründen nur angeschaltet oder abgeschaltet werden kann, ist hierfür ein gutes Beispiel. Über einen längeren Zeitraum gemittelt liefert der Brenner eine bestimmte durchschnittliche Heizleistung, also eine kontinuierliche Größe. Durch die Abstraktion von vielen diskreten Sprüngen der Leistung zu einer sich stetig ändernden durchschnittlichen Leistung wird das Modell vereinfacht. Ähnlich geht man vor, wenn man den Autobahnverkehr mit einer kontinuierlichen Verkehrsdichte als Zustandsgröße beschreibt oder die ganzzahlige Anzahl von Datenpaketen an den Knoten eines Kommunikationsnetzes durch einen reellwertigen Mittelwert ersetzt.

Ziel der Abstraktion ist in beiden Fällen, ein einheitlich kontinuierliches oder ereignisdiskretes Modell zu erhalten, auf das die für die jeweilige Systemklasse entwickelten Methoden angewendet werden können.



## 17.2 Hybride dynamische Systeme

In der automatisierungstechnischen Praxis treten kontinuierliche und ereignisdiskrete Elemente jedoch auch in solchen Kombinationen auf, die es verbieten, durch eine Abstraktion den einen oder anderen Teil zu eliminieren und zu einer einheitlichen Modellform überzugehen. Bevor auf die damit verbundenen Schwierigkeiten bei der Lösung von Automatisierungsaufgaben eingegangen wird, soll die Verknüpfung beider Systemklassen an einem Beispiel erläutert werden.

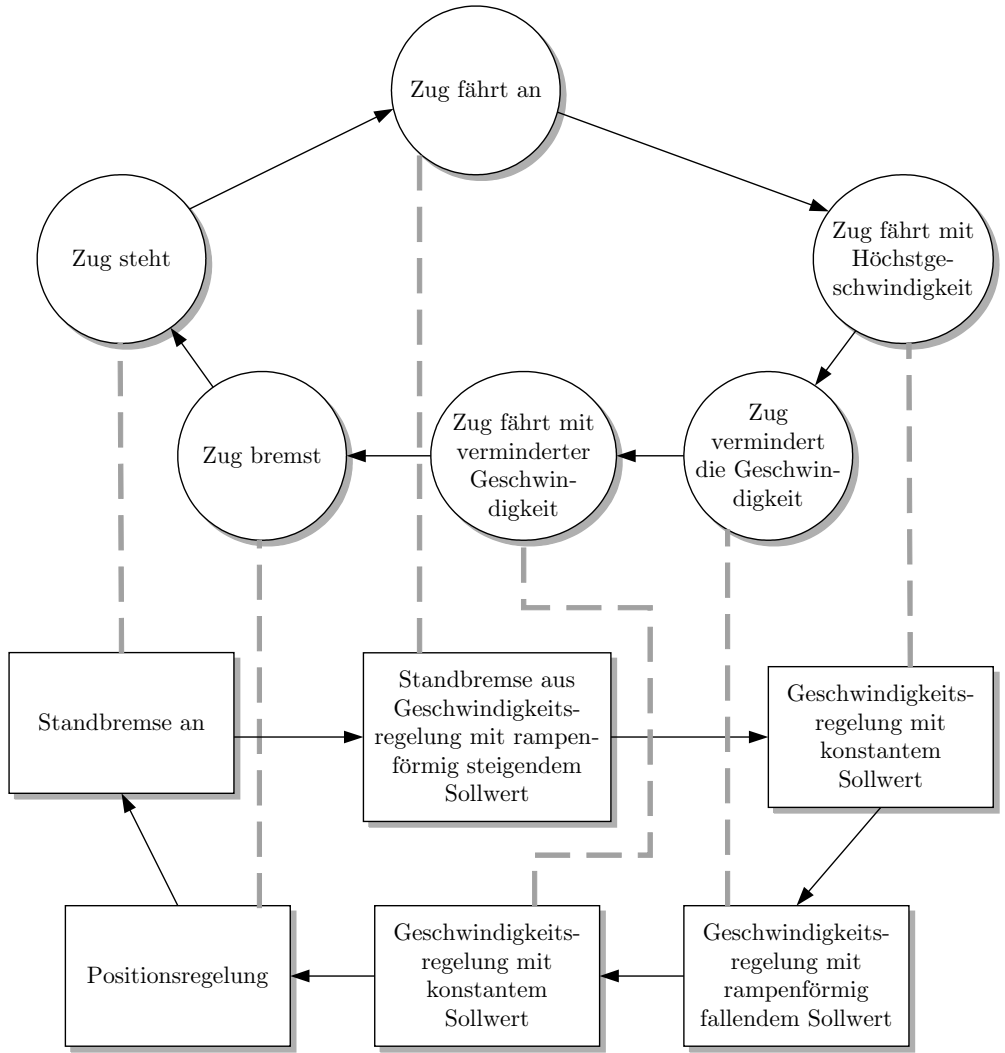
### Beispiel 17.1 Diskrete und kontinuierliche Elemente der U-Bahn-Steuerung

In Aufgabe 13.5 wurde eine automatisch fahrende U-Bahn betrachtet, wobei nur zwischen den diskreten Zuständen „Zug steht“, „Zug fährt an“ und „Zug bremst“ unterschieden wurde. Diese Betrachtungsweise ist zweckmäßig, wenn man auf einer höheren Betrachtungsebene die einzelnen Phasen der U-Bahn-Bewegung untersuchen und steuern will. Bei der Realisierung des vollautomatischen Betriebs muss man jedoch durch unterlagerte Regelungen erreichen, dass die Beschleunigung des Zuges auf die Höchstgeschwindigkeit in einer vorgegebenen Art und Weise erfolgt. Man muss dann die U-Bahn innerhalb des diskreten Zustands „Zug fährt an“ als ein kontinuierliches System betrachten, dessen Position, Geschwindigkeit und Beschleunigung durch ein Zustandsraummodell beschrieben wird.

Kontinuierliche und diskrete Betrachtungsweisen müssen also wie in Abb. 17.1 miteinander verknüpft werden. In der Abbildung stellen die Kreise die diskreten Zustände und die Vierecke die U-Bahn als kontinuierliches System dar. Zu jedem diskreten Zustand gehört ein Betriebszustand, der sich durch diskrete Eingänge wie z. B. „Standbremse an“ auszeichnet, aber gleichzeitig kontinuierliche Regelungen aktiviert wie beispielsweise „Geschwindigkeitsregelung mit konstantem Sollwert“. Typisch dabei ist, dass die diskrete Betrachtungsweise der höheren Abstraktionsebene und die kontinuierliche Betrachtungsweise der niederen Abstraktionsebene zugeordnet ist. Die diskrete Steuerung schaltet die U-Bahn zwischen unterschiedlichen diskreten Zuständen und aktiviert damit auf der unteren Ebene Regelungen mit bestimmten Führungsgrößen. Neben der hier gezeigten zyklischen Folge der sechs diskreten Zustände kann die diskrete Steuerung auch andere Zustandsfolgen erzeugen, beispielsweise ein mehrfaches Durchlaufen eines Abbrems- und Beschleunigungszyklus beim Durchfahren eines kurvenreichen Streckenabschnitts. □

Systeme, die aus einer Verknüpfung von kontinuierlichen und diskreten Elementen bestehen, werden als *hybride Systeme*<sup>1</sup> bezeichnet. Von denen in den letzten Jahren entwickelten vielfältigen Modellformen sei hier nur auf die in Abb. 17.2 gezeigte Kombination eines ereignisdiskreten und eines kontinuierlichen Teilsystems eingegangen, die alle wichtigen Elemente eines

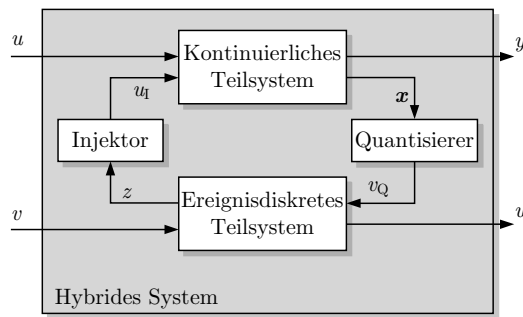
<sup>1</sup> Da in der Vergangenheit der Begriff des hybriden Systems für wertkontinuierliche Systeme verwendet wurde, die aus zeitkontinuierlichen und zeitdiskreten Komponenten bestehen, sei hier betont, dass sich die Begriffe kontinuierlich und diskret bei den hier betrachteten hybriden Systemen wie überall in diesem Buch auf den Wertebereich der Signale beziehen.



**Abb. 17.1:** Diskrete und kontinuierliche Steuerungen für die U-Bahn in Lille

hybriden Systems enthält. Auf Grund des unterschiedlichen Charakters der Signale wird zwischen folgenden Größen unterschieden:

- $v$  – diskrete Eingangsgröße
- $u$  – kontinuierliche Eingangsgröße
- $v_Q$  – diskrete Koppeleingangsgröße
- $u_I$  – kontinuierliche Koppeleingangsgröße
- $w$  – diskrete Ausgangsgröße
- $y$  – kontinuierliche Ausgangsgröße.
- $z$  – diskreter Systemzustand
- $x$  – kontinuierlicher Systemzustand.



**Abb. 17.2:** Hybrides dynamisches System

Um diese Signale unterschiedlichen Typs miteinander verbinden zu können, sind zwei Schnittstellen notwendig, die die wertkontinuierlichen in die wertdiskreten Größen abbilden und umgekehrt. Die beiden entsprechenden Blöcke sind in Abb. 17.2 als *Quantisierer* und *Injektor* bezeichnet. Der Quantisierer bestimmt aus dem kontinuierlichen Systemzustand  $x$  den aktuellen Wert der diskreten Eingangsgröße  $v_Q$  des diskreten Teilsystems. Dies beinhaltet eine Quantisierung, weil einer Menge unterschiedlicher Zustände  $x$  derselbe Wert  $v_Q$  der diskreten Größe zugeordnet ist. Als Beispiel denke man nur an einen Behälter mit einem in der Höhe  $h_A$  über dem Behälterboden angebrachten Auslauf. Es wird nur dann Flüssigkeit durch diesen Auslauf abfließen, wenn der Füllstand  $h(t)$  im Behälter höher als der Auslauf ist:

$$q(t) = \begin{cases} 0 & \text{wenn } h(t) < h_A \\ \sqrt{h(t) - h_A} & \text{wenn } h(t) \geq h_A. \end{cases}$$

Auf der rechten Seite dieser Beziehung für den Massenstrom  $q(t)$  durch den Auslauf sieht man deutlich die quantisierende Wirkung. Als Konsequenz wird sich der Füllstand in dem Behälter im ersten Fall erhöhen ( $\dot{h}(t) > 0$ ), während er im zweiten Fall konstant bleibt ( $\dot{h}(t) = 0$ ), weil die Flüssigkeit über den Auslauf aus dem Behälter fließt. Man erhält also unterschiedliche Modelle des kontinuierlichen Teilsystems, zwischen denen das diskrete Teilsystem schaltet.

Der Injektor nimmt die entgegengesetzte Transformation vor. Aus dem diskreten Zustand  $z(t)$  des diskreten Teilsystems wird eine kontinuierliche Eingangsgröße  $u_I(t)$  gebildet.

Die in Abb. 17.2 kann man beispielsweise direkt für Systeme anwenden, deren hybrider Charakter sich bei einer hierarchischen Zerlegung des Systems offenbart. Dann beschreibt – wie bei der U-Bahn im Beispiel 17.1 – das diskrete Teilsystem den Betriebsmodus und das kontinuierliche Teilsystem das kontinuierliche Verhalten innerhalb des betreffenden Betriebsmodus. Hybride Modelle zeigen also, dass sich viele diskrete Systeme im Detail betrachtet wie kontinuierliche Systeme verhalten.

**Hybride Automaten.** Hybride dynamische Systeme führen auf sehr komplexe Modelle und dementsprechend auf komplexe Analyse- und Steuerungsprobleme. Der Grund hierfür liegt in der Tatsache, dass die grundsätzlich unterschiedlichen mathematischen Werkzeuge zur Behandlung von kontinuierlichen und ereignisdiskreten Systemen bei hybriden Systemen miteinander verknüpft werden müssen. Deshalb wird gegenwärtig nach Modellformen gesucht, die einerseits dem Charakter der hybriden Systeme Rechnung tragen, andererseits auf möglichst überschaubare Analyse- und Entwurfsprobleme führen. Als Beispiel dafür soll hier kurz auf den hybriden Automaten eingegangen werden.

Der hybride Automat entsteht aus einer Kombination eines deterministischen Automaten mit kontinuierlichen Beschreibungsformen. Dabei gilt im  $i$ -ten diskreten Zustand  $z_i \in \mathcal{Z}$  die  $i$ -te Differentialgleichung

$$\dot{\mathbf{x}}(t) = \mathbf{f}_i(\mathbf{x}(t), \mathbf{u}(t)).$$

Wenn das System den  $i$ -ten diskreten Zustand zum Zeitpunkt  $t_k$  annimmt, gehört zu der Differentialgleichung die Anfangsbedingung

$$\mathbf{x}(t_k) = \mathbf{x}_k,$$

wobei  $\mathbf{x}_k$  von dem im vorherigen diskreten Zustand berechneten kontinuierlichen Zustand abhängt.

Die Änderung des diskreten Zustands wird durch eine der folgenden Ursachen bewirkt:

- **Autonomes Schalten:** Wenn eine Umschaltbedingung (*guard*)

$$h_{z'z}(\mathbf{x}(t_k), \mathbf{u}(t_k)) = 0$$

zur Zeit  $t_k$  erfüllt ist, ändert sich der diskrete Zustand zur Zeit  $t_k$  von  $z$  in  $z'$ .

- **Gesteuertes Schalten:** Eine diskrete Eingabe  $v(k)$ , die zur Zeit  $t_k$  auftritt, bewirkt das Umschalten vom Zustand  $z$  zum Zustand  $z'$ , der von der Eingabe mitbestimmt wird.

Die Modellform des hybriden Automaten lässt zu, dass sich beim Umschalten des diskreten Zustands der kontinuierliche Zustand sprunghaft ändert und der Zustand von seinem Wert  $\mathbf{x}(t_k^-)$  vor dem Umschalten auf einen neuen Wert  $\mathbf{x}(t_k^+)$  nach dem Umschalten springt:

$$\mathbf{x}(t_k^+) = \phi(\mathbf{x}(t_k^-)).$$

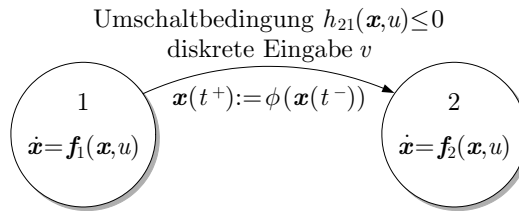
$t_k^-$  und  $t_k^+$  symbolisieren den Zeitpunkt kurz vor bzw. kurz nach dem Zustandssprung. Dabei können wiederum zwei Fälle unterschieden werden:

- **Autonomer Zustandssprung:** Der neue Zustand  $\mathbf{x}(t_k^+)$  wird nur von dem vor dem Umschalten angenommenen Zustand  $\mathbf{x}(t_k^-)$  und gegebenenfalls dem diskreten Zustandswechsel  $z \rightarrow z'$  bestimmt.

- **Gesteuerter Zustandssprung:** Der Zustandssprung wird durch eine diskrete Eingabe  $v(k)$  ausgelöst und der neue Wert  $x(t_k^+)$  hängt auch von dieser Eingabe ab.

Zustandssprünge treten in Modellen vor allem dann auf, wenn die zu beschreibenden physikalischen Phänomene sehr unterschiedliche Zeitkonstanten haben und das Modell die langsamen Vorgänge richtig wiedergeben soll. Dann kann es zweckmäßig sein, die schnellen Zustandswechsel als Zustandssprünge zu modellieren.

Andererseits kann man in Reglern Zustandssprünge programmieren, um ein stoßfreies Umschalten zwischen unterschiedlichen Betriebsmodi zu erzeugen. Wenn man im Antriebsstrang eines Fahrzeugs durch einen Wechsel des Gangs das Übersetzungsverhältnis von Motordrehzahl und Raddrehzahl verändert, muss die Geschwindigkeitsregelung auf diese sprungförmige Veränderung ihrer Regelstrecke gegebenenfalls durch einen Zustandssprung reagieren.



**Abb. 17.3:** Ausschnitt aus dem Automatengraphen eines hybriden Automaten

Grafisch können hybride Automaten wie in Abb. 17.3 dargestellt werden, wobei dem diskreten Zustand des Automaten eine Zustandsgleichung für die kontinuierlichen Bewegungen zugeordnet sind. Die Umschaltbedingung und die diskrete Eingabe, die die Richtung des Zustandswechsels beeinflussen, sind zusammen mit der Funktion, die den Zustandssprung beschreibt, die Kantenbewertungen.

### Beispiel 17.2 Darstellung einer Uhr als hybrides dynamisches System

Eine Uhr, die Bestandteil eines Steuerungsalgorithmus ist, ist ein einfaches Beispiel für ein hybrides System. Ist die Uhr im Ruhezustand, der als diskreter Zustand  $z_1$  bezeichnet wird, so gilt

$$\dot{x}(t) = f_1(x(t)) = 0, \quad x(0) = 0.$$

Zum Zeitpunkt  $t_1$  wird die Uhr durch eine diskrete Eingangsgröße gestartet, d. h., die Uhr geht in den diskreten Zustand  $z_2$  über, in dem die Differentialgleichung

$$\dot{x}(t) = f_2(x(t)) = 1, \quad x(t_1^+) = 0$$

gilt. Eine Umschaltbedingung

$$h_{12}(x(t)) = x(t) - \bar{x} \leq 0$$

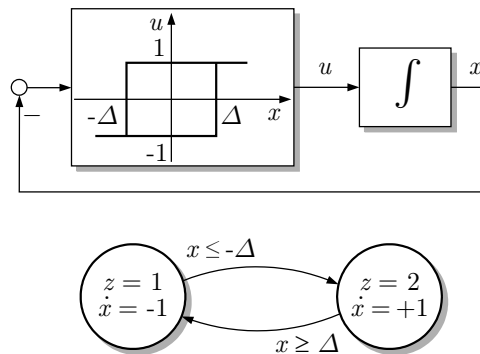
erkennt den Zeitpunkt, an dem die Uhr die Zeit  $\bar{x}$  anzeigt. Dies ist offenbar zur Zeit  $t_2 = t_1 + \bar{x}$  der Fall. Dann geht die Uhr in den diskreten Zustand  $z_1$  über, wobei sie eine diskrete Ausgangsgröße  $w$

erzeugt, die das Ablaufen der Zeitspanne  $\bar{x}$  anzeigt. Der kontinuierliche Zustand der Uhr wird beim Zustandsübergang  $z_2 \rightarrow z_1$  auf den Wert 0 zurückgesetzt:

$$x(t_2^+) = 0. \quad \square$$

**Beispiel 17.3** Beschreibung eines Zweipunktregelkreises durch einen hybriden Automaten

Abbildung 17.4 zeigt einen Zweipunktregelkreis, bei dem die Stellgröße  $u(t)$  über eine Hysterese von der Zustandsgröße der Regelstrecke abhängt. Der Regelkreis funktioniert im Wesentlichen so, dass bei einem zu kleinen Wert der Zustandsgröße die Stellgröße auf den Wert 1 gesetzt wird und bei zu großer Zustandsgröße den Wert  $-1$  erhält. Dabei muss eine Hysteresis verwendet werden, um zu verhindern, dass die Stellgröße sehr schnell umgeschaltet wird, was das Stellglied überlasten kann.



**Abb. 17.4:** Zweipunktregelkreis

Die Hysterese ordnet den Werten  $-\Delta \leq x \leq \Delta$  der Regelgröße zwei Werte der Stellgröße zu. Welcher Wert tatsächlich als Stellgröße verwendet wird, hängt von der Richtung ab, in der die Hysteresekennlinie durchlaufen wird. Diese Richtungsabhängigkeit kann nur dargestellt werden, indem man einen diskreten Zustand einführt. Der Regelkreis kann deshalb nur als hybrides System beschrieben werden.

Der in der Abbildung gezeigte hybride Automat hat zwei diskrete Zustände, die den beiden Ästen der Hysterese zugeordnet sind. Als Modell für die Regelstrecke wurde der Einfachheit halber ein Integrator gewählt, so dass für die kontinuierlichen Vorgänge eine sehr einfache Differentialgleichung entsteht.

Das Verhalten des Zweipunktregelkreises kann man dadurch simulieren, dass man sowohl den Anfangszustand  $x_0$  der kontinuierlichen Zustandsgröße als auch den diskreten Zustand  $z_0$  vorgibt. Daraus ergibt sich, welcher diskrete Zustand und welche Differentialgleichung „aktiviert“ sind. Wenn die Umschaltbedingung erfüllt ist, wechselt der diskrete Zustand (autonomes Schalten), was in diesem Beispiel keine abrupte Änderung der kontinuierlichen Zustandsgröße nach sich zieht ( $x(t_k^+) = x(t_k^-)$ ).  $\square$

Ein anderes Beispiel, bei dem die kontinuierliche Dynamik nicht aus einem einfachen Integrator besteht, ist ein Kraftfahrzeug. Durch das Getriebe wird die Übersetzung zwischen Motor und Fahrwerk verändert. Es ergeben sich dadurch unterschiedliche dynamische Eigenschaften für das Fahrzeug. Der Gangwechsel ist ein wichtiger Bestandteil einer automatischen Steuerung,

so dass das Fahrzeug als hybrides System betrachtet werden muss, vor allem dann, wenn es sich um fahrerlose Transportfahrzeuge handelt, bei denen die Automatisierungseinrichtung die diskrete Getriebestellung und die kontinuierliche Beschleunigung vorgeben muss.

**Aufgabe 17.1** *Darstellung eines Behältersystems als hybrider Automat*

Das in Abb. 3.18 auf S. 111 dargestellte Behältersystem hat eine hybride Dynamik, weil die Verbindungsleitung nur dann „eingeschaltet“ ist, wenn mindestens einer der beiden Behälter bis über diese Leitung gefüllt ist. Wenn die Behälter innerhalb eines Batchprozesses verwendet werden, dienen die Pumpe und die Ventile als diskrete Stellglieder, die nur an- und ausgeschaltet bzw. vollständig geöffnet und geschlossen werden ( $u_i \in \{0, 1\}$ ). Man kann dann die Gleichungen (A.14) für jeden auf diese Weise entstehenden Betriebszustand in ein Modell eines autonomen Systems (Systems ohne Eingang) umformen.

Stellen Sie das so erhaltene Modell als hybriden Automaten dar. Handelt es sich bei den Schaltvorgängen um autonomes oder gesteuertes Schalten?  $\square$

## Literaturhinweise

Hybride dynamische Systeme sind ein aktuelles Forschungsthema der Automatisierungstechnik. Erste zusammenfassende Darstellungen sind [25, 72, 102]. Gegenwärtig konzentriert sich die Forschung u. a. auf stückweise lineare Systeme, bei denen die kontinuierlichen Teilsysteme durch lineare Differentialgleichungen beschrieben sind [47].





# Literaturverzeichnis

1. Abel, D.: *Petri-Netze für Ingenieure*, Springer-Verlag, Berlin 1990.
2. Abel, D.; Lemmer, K. (Hrsg.): *Theorie ereignisdiskreter Systeme*, Oldenbourg-Verlag, München 1998.
3. Ackermann, J.: *Robuste Regelung*, Springer-Verlag, Berlin 1993.
4. Albert, J.; Ottmann, T.: *Automaten, Sprachen und Maschinen für Anwender*, Bibliographisches Institut, Mannheim 1983.
5. Arbib, M. A.: A common framework for automata theory and control theory, *J. SIAM Control*, Ser. A, **2** (1965), pp. 206–222.
6. Arbib, M. A.: Automata theory and control theory - a rapprochement, *Automatica* **3** (1966), pp. 161–189.
7. Baehr, H. D.: *Thermodynamik*, Springer-Verlag, Berlin 1996.
8. Baier, C.; Katoen, J.-P.: *Principles of Model Checking*, MIT Press: Cambridge 2008.
9. Beitz, W.; Grote, K.-H. (Hrsg.): *DUBBEL Taschenbuch für den Maschinenbau*, Springer-Verlag, Berlin 1997, (19. Auflage).
10. Berger, H.: *Automatisieren mit STEP 7*, Publicis Corporate Publishing 2006.
11. Besancon, G.: An overview on observer tools for nonlinear systems, in Besancon, G. (ed.): *Nonlinear Observers and Applications*, Springer-Verlag, Heidelberg 2007, pp. 1–33.
12. Bewersdorff, J.: *Glück, Logik und Bluff*, Vieweg, Braunschweig 2001.
13. Blanke, M.; Kinnaert, M.; Lunze, J.; Staroswiecki, M.: *Diagnosis and Fault-Tolerant Control*, Springer-Verlag, Heidelberg 2016.
14. Bochmann, D.: *Einführung in die strukturelle Automatentheorie*, Verlag Technik, Berlin 1975.
15. R. Bosch GmbH (Hrsg.): *Kraftfahrtechnisches Taschenbuch*, Vieweg, Braunschweig 1999.
16. Bukharaev, R. G.: *Theorie der stochastischen Automaten*, B.G. Teubner, Stuttgart 1995.
17. Carlyle, J. W.: State-calculable stochastic sequential machines, equivalences and events. Switching circuit theory and logic. *Design IEEE Conf. Re4c*. 16C13, New York 1965.
18. Cassandras, C. G.; Lafortune, S.: *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, Boston 1999.
19. Chiang, L. H.; Russell, E. L.; Braatz, R. D.: *Fault Detection and Diagnosis in Industrial Systems*, Springer-Verlag, London 2001.
20. Chow, E. Y.; Willsky, A. S.: Analytical redundancy and the design of robust failure detection systems, *IEEE Trans. AC-29* (1984), pp. 603–614.
21. Claus, V.: *Stochastische Automaten*, B.G. Teubner, Stuttgart 1971.
22. Davison, E. J.: Connectability and structural controllability of composite systems, *Automatica* **13** (1977), pp. 109–123.
23. DIN 19226: *Regelungstechnik und Steuerungstechnik; Begriffe und Benennungen*, zitiert nach Klein, M.: *Einführung in die DIN-Normen*, B. G. Teubner, Stuttgart 1985, S. 893.
24. Eichhorn, M.; Pfützenreuter, T.; Schmidt, M.; Walzl, B.: Modellierung, Regelung und Navigation von teilautonomen, ferngesteuerten Unterwasserfahrzeugen, *Automatisierungstechnik* **49** (2001), S. 373–380.

25. Engell, S.; Frehse, G.; Schnieder, E. (Eds.): *Modelling, Analysis and Synthesis of Hybrid Dynamical Systems*, Springer-Verlag, Heidelberg 2002.
26. Even, S.: *Graph Algorithms*, London 1979.
27. Föllinger, O.: *Optimierung dynamischer Systeme*, Oldenbourg-Verlag, München 1985.
28. Föllinger, O.: *Nichtlineare Regelungen*, Oldenbourg-Verlag, München 1989.
29. Fossen, T. I.: *Guidance and Control of Ocean Vehicles*, John Wiley & Sons, Chichester 1994.
30. Frank, P. M.: Fault diagnosis in dynamic systems via state estimation - a survey, in Tzafestas u. a. (eds.): *System Fault Diagnostics, Reliability and Related Knowledge-Based Approaches*, vol. 1, pp. 35–98, D. Reidel Publ. Co. 1987.
31. Frank, P. M.: Diagnoseverfahren in der Automatisierungstechnik, *Automatisierungstechnik*, **42** (1994), S. 47–64.
32. Franksen, O. I., Falster, P.; Evens, F. J.: *Qualitative Aspects of Large-Scale Systems*, Springer-Verlag, New York 1979.
33. Früh, K. F.; Maier, U.; Schaudel, D.: (Hrsg.): *Handbuch der Prozeßautomatisierung*, Oldenbourg-Verlag, München 2009 (4. Auflage).
34. Gevatter, H.-J. (Hrsg.) *Automatisierungstechnik* (Band 1: Mess- und Sensorsysteme, Band 2: Geräte, Band 3: Aktoren), Springer-Verlag, Berlin 2000.
35. Giua, A.: Petri net state estimators based on event observation, *IEEE Conf. on Decision and Control*, San Diego 1997, pp. 4086–4091.
36. Giua, A.; Seatzu, C.; Julvez, J.: Marking estimation of Petri nets with pairs of nondeterministic transitions, *Asian J. of Control* **6** (2004) No. 2, pp. 270–280.
37. Gössel, M.: *Angewandte Automatentheorie*, Akademie-Verlag, Berlin 1972.
38. Harbach, F.; Janschek, K.; Jumar, U.; Sawodny, O., Spohr, G.-U.: Herausfordernde Anwendungsgebiete der Automatisierungstechnik, *Automatisierungstechnik* **55** (2007) H. 5, S. 260–265.
39. Heidepriem, J.: *Prozessinformatik*, Oldenbourg-Industrieverlag, München 2000.
40. Hopcroft, J. E.; Ullman, J. D.: *Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie*, Addison-Wesley, Bonn 1988.
41. Isermann, R.: Process fault detection based on modelling and estimation methods - a survey, *Automatica* **20** (1984), pp. 387–404.
42. Isermann, R.: Modellgestützte Überwachung und Fehlerdiagnose technischer Systeme, *Automatisierungstechnische Praxis* **38** (1996), S. 9–20 u. 48–57.
43. Isermann, R.: *Identifikation dynamischer Systeme*, Springer-Verlag, Berlin 1988.
44. Iftar, A.; Davison, E. J.: Decentralized robust control for dynamic routing of large scale networks, *American Control Conf.*, San Diego 1990, pp. 441–446.
45. Isidori, A.: *Nonlinear Control Systems*, Springer-Verlag, Berlin 1996.
46. Jörns, C.; Litz, L.; Bergold, S.: Automatische Erzeugung von SPS-Programmen auf der Basis von Petri-Netzen, *Automatisierungstechnische Praxis* **37** (1995), S. 10–14.
47. Johansson, M.: *Piecewise Linear Control Systems*, Springer-Verlag, Heidelberg 2003.
48. Juen, G.; Krämer, W.; Schaffner, J.; Zeitz, M.: Regelung von Radioteleskopen, *Automatisierungstechnik* **43** (1995), S. 215–223.
49. Jünemann, R.; Beyer, A.: *Steuerung von Materialfluß- und Logistiksystemen*, Springer-Verlag, Berlin 1998.
50. Kalman, R. E.: Contributions to the theory of optimal control, *Boletin de la Sociedad Matematica Mexicana*, **5** (1960), 102–119.
51. Khalil, H. K.: *Nonlinear Systems*, Prentice-Hall, Upper Saddle River 1996.
52. Kiencke, U.; Nielsen, L.: *Automotive Control Systems*, Springer-Verlag, Berlin 2005.
53. Kiesewetter, W.; Klinkner, W.; Reichelt, W.; Steiner, M.: Der neue Brake Assist von Mercedes-Benz, *Automobiltechnische Zeitschrift* **99** (1997), 330–339.
54. König, R.; Quäck, L.: *Petri-Netze in der Steuerungstechnik*, Verlag Technik, Berlin 1988.

55. Kowalewski, S.: *Modulare diskrete Modellierung verfahrenstechnischer Anlagen zum systematischen Steuerungsentwurf*. Dissertation Univ. Dortmund, Schriftenreihe des Lehrstuhls für Anlagensteuerungstechnik, Band 1, Shaker-Verlag 1996.
56. Krapp, M.: *Digitale Automaten*, Verlag Technik, Berlin 1988.
57. Lauber, R.: *Prozeßautomatisierung*, Band 1, Springer-Verlag, Berlin 1989.
58. Lauber, R.; Göhner, P.: *Prozeßautomatisierung*, Band 2, Springer-Verlag, Berlin 1999.
59. Leonhardt, S.; Walter, M. (Hrsg.): *Medizintechnische Systeme*, Springer-Verlag, Heidelberg 2015.
60. Lin, C. T.: System structure and minimal structure controllability, *IEEE Trans.* **AC-22** (1977), pp. 855–862.
61. Lin, F.: Diagnosability of discrete-event systems and its applications, *Discrete Event Dynamic Systems: Theory and Applications* **4** (1994), pp. 197–212.
62. Lin, F.; Wonham W. M.: On observability of discrete-event systems, *Information Sciences* **44** (1988), pp. 173–198.
63. Listmann, K. D.; Wenzelburger, P.; Allgöwer, F.: Industrie 4.0 - (R)evolution ohne Regelungstechnik? *Automatisierungstechnik* **64** (2016), S. 507–520.
64. Litz, L.; Frey, G.: Methoden und Werkzeuge zum industriellen Steuerungsentwurf – Historie, Stand, Ausblick. *Automatisierungstechnik* **47** (1999), S. 145–156.
65. Litz, L.: *Grundlagen der Automatisierungstechnik*, Oldenbourg-Verlag, München 2004.
66. Luenberger, D. G.: Observing the state of a linear system, *IEEE Trans.* **MIL-8** (1994), pp. 163–171.
67. Lunze, J.: *Robust Multivariable Feedback Control*, Prentice-Hall, London 1988.
68. Lunze, J.: *Feedback Control of Large-Scale Systems*, Prentice-Hall, London 1992.
69. Lunze, J.: *Regelungstechnik*, Band 1, Springer-Verlag, Berlin 2020 (12. Auflage).
70. Lunze, J.: *Regelungstechnik*, Band 2, Springer-Verlag, Berlin 2020 (10. Auflage).
71. Lunze, J.: *Ereignisdiskrete Systeme*, de Gruyter Oldenbourg, Berlin 2017 (3. Auflage).
72. Lunze, J.; Lamnabhi-Lagarigue, F. (Eds.): *Handbook of Hybrid Systems Control: Theory, Tools, Applications*, Cambridge University Press, Cambridge 2009.
73. Lunze, J.; Schröder, J.: State observation and diagnosis of discrete-event systems described by stochastic automata, *Discrete Event Dynamic Systems: Theory and Applications* **11** (2001), pp. 319–369.
74. Maul, C.; Zeller, M.: Unterschiede und Gemeinsamkeiten von Leitsystemen in kontinuierlicher und diskreter Fertigung, *Automatisierungstechnische Praxis* (2008) H.9, S. 36–40.
75. Mitschke, M.; Wallentowitz, H.: *Dynamik der Kraftfahrzeuge*, Band A: Antrieb und Bremsung, Springer-Verlag, Berlin 2015 (5. Auflage).
76. Moody, J. O.; Antsaklis, P.: *Supervisory Control of Discrete Event Systems Using Petri Nets*, Kluwer Academic Publishers, Boston 1998.
77. Moore, E. F.: Gedanken-Experimente on sequential machines. In: Shannon, C. E.; McCarthy, J. (Eds.): *Automata Studies*, Princeton Univ. Press 1956, pp. 129–153.
78. Morari, M.; Zafriou, E.: *Robust Process Control*, Prentice-Hall, Englewood Cliffs 1989.
79. Petri, C. A.: *Kommunikation mit Automaten*, Dissertation, TH Darmstadt 1962.
80. Polke, M. (Hrsg.): *Prozeßleittechnik*, Oldenbourg-Verlag, München 1994.
81. Özveren, C. M.; Willsky, A. S.: Observability of discrete-event dynamic systems, *IEEE Trans.* **AC-35** (1990), pp. 797–806.
82. Papageorgiou, M.: *Optimierung*, Oldenbourg-Verlag, München 1996.
83. Ramadge, P. J.; Wonham, W. M.: The control of discrete event Systems, *Proc IEEE* **77** (1989), pp. 81–98.
84. Reinisch, K.: *Analyse und Synthese kontinuierlicher Steuerungssysteme*, Verlag Technik, Berlin 1979.
85. Reinschke, K. J.: *Multivariable Control - A Graph-Theoretic Approach*, Springer-Verlag, Berlin 1988.

86. Richter, J. H.: *Reconfigurable Control on Nonlinear Dynamical Systems: A Fault-Hiding Approach*, Springer-Verlag, Berlin 2011.
87. Roppenecker, G.: Fahrzeugdynamik: Grundlagen der Modellierung und Regelung, *Automatisierungstechnik* **42** (1994), H. 10, S. 429–441.
88. Sampath, M.; Sengupta, R.; Lafortune, S.; Sinnamohideen, K.; Teneketzis, D.: Diagnosability of discrete-event systems, *IEEE Trans. AC-40* (1995), pp. 1555–1575.
89. Sander, P.; Stucky, W.; Herschel, R.: *Automaten, Sprachen, Berechenbarkeit*, B.G. Teubner, Stuttgart 1995.
90. Schaffner, J.; Zeitz, M.: Entwurf nichtlinearer Beobachter, In: Engell, S. (Hrsg.): *Entwurf nichtlinearer Regelungen*, Oldenbourg-Verlag, München 1995.
91. Schmidt, G.; Ströhlein, T.: *Relationen und Graphen*, Springer-Verlag, Berlin 1989.
92. Schnieder, E.: *Prozessinformatik*, Friedr. Vieweg & Sohn Verlagsgesellschaft, Braunschweig 1993.
93. Schnieder, E.: *Methoden der Automatisierung*, Vieweg, Braunschweig 1999.
94. Schönfeld, R.: *Bewegungssteuerungen*, Springer-Verlag, Berlin 1998.
95. Schuler, H. (Hrsg.): *Prozeßführung*, Oldenbourg-Verlag, München 1999.
96. Seborg, D. E.; Edgar, T. F.; Mellichamp, D. A.: *Process Dynamics and Control*, John Wiley & Sons, New York 1989.
97. Srinivasan, V. S.; Jafari, M. A.: Fault detection/monitoring using timed Petri-nets, *IEEE Trans. SMC-23* (1993).
98. Starke, P. H.: *Abstrakte Automaten*, Deutscher Verlag der Wissenschaften, Berlin 1969.
99. Starke, P. H.: *Petri-Netze*, Deutscher Verlag der Wissenschaften, Berlin 1980.
100. Steffen, T.: *Control Reconfiguration of Dynamical Systems*, Springer-Verlag, Berlin 2005.
101. Töpfer, H.; Rudert, S.: *Einführung in die Automatisierungstechnik*, Verlag Technik, Berlin 1979.
102. van der Schaft, A.; Schumacher, H.: *An Introduction to Hybrid Dynamical Systems*, Springer-Verlag, London 2000.
103. Walther, H.; Nägler, G.: *Graphen, Algorithmen, Programme*, Fachbuchverlag, Leipzig 1987.
104. Wellenreuther, G.; Zastrow, G.: *Steuerungstechnik mit SPS*, Vieweg, Braunschweig 1998.
105. Wend, H.-D.: *Strukturelle Analyse linearer Regelungssysteme*, Oldenbourg-Verlag, München 1993.
106. Werner, J. (Hrsg.): *Kooperative und autonome Systeme der Medizintechnik*, Oldenbourg-Verlag, München 2005.
107. Wernstedt, J.: *Experimentelle Prozeßanalyse*, Verlag Technik, Berlin 1989.
108. Zander, H.-J.: *Logischer Entwurf binärer Systeme*, Verlag Technik, Berlin 1989.
109. Zander, H.-J.: Entwurf von Ablaufsteuerungen für ereignisdiskrete Prozesse auf der Basis geeigneter Steuerstreckenmodelle, *Automatisierungstechnik* **53** (2005), S. 140–150.
110. Zander, H.-J.: *Steuerung ereignisdiskreter Systeme*, Springer Fachmedien Wiesbaden 2015.
111. Ziegler, J. G.; Nichols, N. B.: Optimum settings for automatic controllers, *Trans. ASME* **64** (1942), pp. 759–768.

# Anhang 1

## Lösungen der Übungsaufgaben

### **Aufgabe 1.2** *Automatisierungstechnik im täglichen Leben*

Der Betrieb der verschiedenen Geräte ist schematisch in Abbildung A.1 dargestellt. Dabei sind manuelle Tätigkeiten wie z. B. die Eingabe von Automatisierungszielen in runde Flächen eingetragen, während die automatisch ablaufenden Prozesse durch Blöcke dargestellt sind. Nicht berücksichtigt wurden Störungen, vorzeitiges Beenden des Betriebs und Schleifen, die eine wiederholte Ausführung bestimmter Vorgänge ermöglichen.

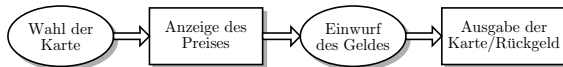
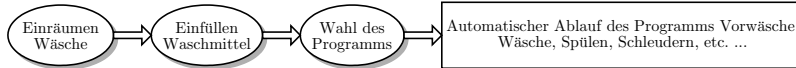
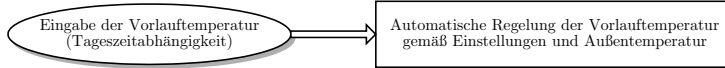
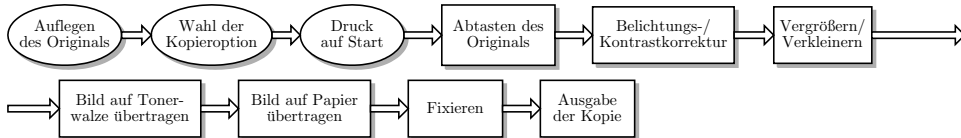
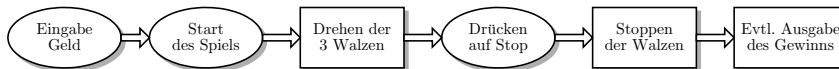
Um die Wirkung der Automatisierungstechnik zu erkennen, muss zwischen der Realisierung der Prozesse und deren Steuerung unterschieden werden. Die Berechnung des Fahrpreises durch den Fahrkartensautomaten oder das Fokussieren eines Bildes durch die Digitalkamera sind mechanische oder elektrische Prozesse. Die Automatisierung betrifft die Steuerung des Gerätes, also das rechtzeitige An- bzw. Abschalten dieser Prozesse.

Alle betrachteten Geräte verlieren ohne Automatisierung ihre Funktion. Ein Fahrkartensautomat, der nur aus Münzschlitz, Knöpfen, Anzeige, Rückgabefach und Fahrkartendrucker besteht, kann seine Aufgabe nicht erfüllen. Ohne Automatisierung kann er nicht auf Knopfdruck reagieren, geschweige denn komplexere Aufgaben wie die Geldrückgabe ausführen. Diese Aufgaben müssten dann durch einen Menschen ausgeführt werden, wie es früher an einem Fahrkartenschalter geschah.

An den Beispielen wird klar, dass der Benutzer automatisierter Prozesse nur noch die Zielstellungen vorgibt und die Automatisierungseinrichtung die Steuerung und Überwachung des Prozesses übernimmt (Abb. 1.9 auf S. 20).

Eine weitergehende Automatisierung erfordert in den meisten Fällen auch eine erweiterte Eingabe, denn das Automatisierungsziel muss nach wie vor vom Nutzer vorgegeben werden. So kann man den Fahrkartenverkauf für Kunden beschleunigen, wenn bekannt ist, dass der Kunde häufig eine bestimmte Strecke fährt. Dann kann diese Strecke auf einem Eingabemedium, z. B. der BahnCard, gespeichert werden und der Kauf einer Karte für diese Strecke erfolgt nach den gespeicherten Vorgaben.

Beim Kopierer erfordert eine weitergehende Automatisierung zunächst eine Mechanisierung des Einzuges des Originals, wenn dieses in gebundener Form vorliegt. So muss ein Buch Seite für Seite aufgeschlagen und richtig positioniert werden, was zunächst eine Aufgabe der Mechanisierung und für die Durchführung dann eine Aufgabe der Automatisierung ist.

**1. Fahrkartenautomat****Geldautomat****2. Digitalkamera****3. Waschmaschine****4. Heizungsanlage****5. Kopiermaschine****6. Spielautomat****Abb. A.1:** Funktionsablauf verschiedener Geräte des täglichen Lebens**Aufgabe 1.3 Automatiktoaster und andere automatische Geräte**

Der Automatiktoaster wirft den Toast selbsttätig nach einer bestimmten Zeit aus. Ob der Toast dann die gewünschte Bräune hat, wird dabei jedoch nicht kontrolliert. Da der Toaster lediglich die Toastzeit misst, die Bräunung aber von der Temperatur des Toasters und der Feuchtigkeit und Dicke der Toastscheibe abhängt, ist die Zeit nur eine von mehreren Bestimmungsgrößen für die Bräune der Toastscheiben. Erfahrungsge-  
mäß ändert sich die Bräunung der Scheiben im Betrieb, weil die Toastzeit beim kalten Toaster länger dauert als beim bereits vorgewärmten Toaster und dieser Einfluss nicht in der Zeitmessung berücksichtigt wird.

Der Bezeichnung „Automatiktoaster“ ist durch die umgangssprachliche Verwendung des Begriffs „automatisieren“ begründet, die die Bedeutung „es geht von selbst“ hat. Die Toastscheibe kommt zwar von selbst wieder aus dem Toaster heraus, aber es fehlt eine Steuerung, die für automatisierte Systeme typisch ist und die beim Toaster für die richtige Bräune sorgt.

Die anderen angegebenen Systeme werden durch eine Automatisierungseinrichtung gesteuert. Eine automatische Schranke schließt sich bei der Annäherung eines Zuges selbsttätig und öffnet sich nach der Zugdurchfahrt. Die Steuerung erhält dafür Informationen über die Annäherung eines Zuges und die Durchfahrt aller Wagen durch die Schranke. Ein Automatikgetriebe passt den Gang der Belastung des Fahrzeugs an, wofür Messgrößen über die aktuelle Drehzahl und den Gaspedalwinkel ausgewertet werden. Ein Kaffeeautomat stellt selbsttätig ein Getränk her, wobei er vorher auf den Einwurf des geforderten

Geldes achtet und gegebenenfalls Kleingeld zurückgibt. Die automatische Klimaanlage enthält eine Regelung, die die Leistung der Klimaanlage so steuert, dass die Raumtemperatur auf einem vorgegebenen Sollwert liegt.

Diesen Geräten ist gemeinsam, dass ein mechanisierter Prozess durch eine Automatisierungseinrichtung gesteuert wird.

### **Aufgabe 1.5** *Autofahren als Problem der Mehrebenensteuerung*

1. Auf der obersten Hierarchieebene plant der Fahrer seine Fahrtroute zum vorgegebenen Ziel. Das Ergebnis kann als eine Folge von Straßen dargestellt werden, die der Fahrer passieren will (Soltrajektorie).

Eine Hierarchieebene tiefer muss sich der Fahrer auf die gegebene Situation im Straßenverkehr einstellen. Er muss Straßenschilder und Verkehrsampeln beachten und auf den Verkehr achten. Der Verkehrssituation entsprechend ermittelt er Sollwerte für die Geschwindigkeit und die Richtung, die er unter Verwendung des Gaspedalwinkels bzw. des Lenkwinkels als Stellgröße zu erreichen versucht. Der Fahrer erfüllt dabei die Aufgabe eines Reglers: Wenn er zu langsam fährt, vergrößert er den Gaspedalwinkel, fährt er zu schnell verkleinert er diesen.

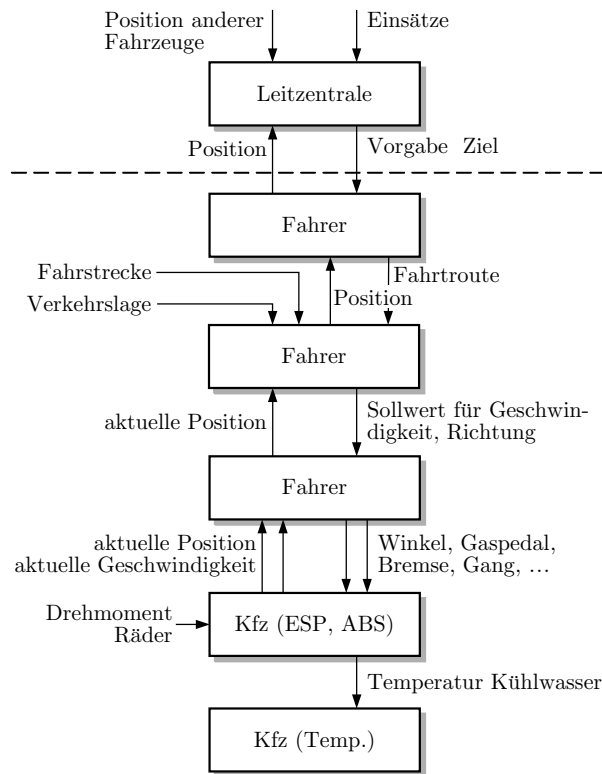
Auf die unteren Ebenen der Steuerung hat der Fahrer keinen direkten Einfluss mehr, da diese von der Elektronik des Fahrzeugs übernommen werden. Bei modernen Fahrzeugen wirkt sich zum Beispiel der Winkel des Gaspedals bzw. der Bremse nicht mehr direkt auf die Geschwindigkeit des Fahrzeugs aus. Steuerungen wie ESP (Elektronisches Stabilisierungsprogramm) und ABS (Antiblockiersystem) ändern den Befehl des Fahrers so ab, dass zu jedem Zeitpunkt größtmögliche Sicherheit gewährleistet ist.

Auf der untersten Hierarchieebene laufen Regelungsvorgänge ab, die nicht direkt die Bewegung des Fahrzeugs beeinflussen, sondern eine reibungslose Arbeit der übergeordneten Systeme garantieren. Ein Beispiel dafür ist der Regelkreis, der die Temperatur des Kühlwassers konstant hält (Abb. A.2).

2. Aus dem Blockschaltbild geht hervor, dass Steuerungsaufgaben der unteren Ebenen automatisch gelöst werden. Die Automatisierung ist hier notwendig, weil der Fahrer diese Aufgaben nicht mit derselben Güte lösen könnte (z. B. ABS oder Kühlerwassertemperaturregelung) bzw. weil die automatisierten Funktionen zu einer höheren Sicherheit des Fahrzeugs im Straßenverkehr führen. Die Praxis zeigt, dass jedesmal, wenn sich weitere Steuereinrichtungen im Fahrzeugbau durchgesetzt haben, die Unfallzahlen zurückgehen.

Fahrerassistenzsysteme wie der Abstandsregeltempomat, der Spurhalteassistent oder der Notbremsassistent übernehmen Aufgaben der höheren Ebene, die bis vor einigen Jahren allein vom Fahrer ausgeführt werden mussten. Die Einführung elektronischer Routenplaner entlastet den Fahrer auch von den Planungsaufgaben in der obersten Ebene der Steuerungshierarchie. Diese Beispiele zeigen, dass der Fahrer mit fortschreitender Entwicklung immer mehr von kurzfristigen Steuerungsaufgaben entlastet wird. Für das autonome Fahren sind allerdings noch viele weitere Automatisierungsschritte notwendig, von denen sich viele nicht nur mit dem eigenen Fahrzeug, sondern auch mit der Situationserkennung und Überwachung des Straßenverkehrs in der Umgebung beschäftigen. Schließlich muss unter Nutzung einer Fahrzeug-zu-Fahrzeug-Kommunikation erreicht werden, dass benachbarte Fahrzeuge kooperative Aufgaben wie das Einfädeln an einer Fahrbahnzusammenführung oder das Überqueren einer ampellosen Kreuzung unter Einhaltung aller Sicherheitsforderungen lösen.

3. Das Blockschaltbild zeigt, dass die Leitzentrale eine übergeordnete Aufgabe wahrnimmt. Abhängig von den aktuellen Einsätzen und der Position mehrerer Fahrzeuge gibt die Zentrale den Fahrern die Fahrziele vor. In jedem Fahrzeug setzt sich die Automatisierungshierarchie wie in der Abbildung gezeigt nach unten fort.



**Abb. A.2:** Automatisierungsebenen beim Autofahren

### Aufgabe 2.1 Blockschaltbild des Antriebsstrangs eines Kraftfahrzeugs

- Abbildung A.3 zeigt das gesuchte Blockschaltbild. Das durch den Gaspedalwinkel, die Stellung der Kupplung und die Gangwahl festgelegte Antriebsmoment wird um das durch den Fahrwiderstand entstehende Moment sowie um das Bremsmoment verkleinert. Das daraus resultierende Moment auf die Räder verkleinert oder vergrößert die Fahrzeuggeschwindigkeit.

Will man im Blockschaltbild erfassen, dass der Fahrer seine Vorgaben für das Gaspedal, die Kupplung und den Gang in Abhängigkeit von der Verkehrssituation und der Geschwindigkeit wählt, so muss man eine Rückführung der Geschwindigkeit auf den den Fahrer beschreibenden Block vorsehen und als weitere Eingangsgröße die Verkehrssituation eintragen.

Das Blockschaltbild zeigt eine Reihenschaltung von fünf Teilsystemen. Eine Rückführung entsteht, wenn man durch ein zusätzliches Teilsystem darstellen will, dass der Fahrwiderstand von der Geschwindigkeit abhängig ist.



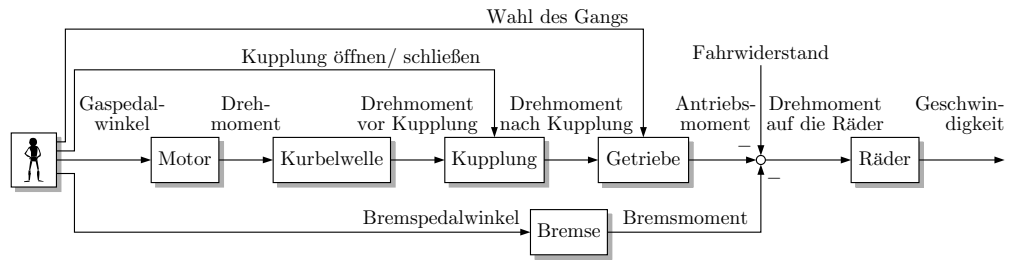


Abb. A.3: Blockschaltbild des Antriebsstrangs

2. Die folgende Tabelle zeigt die Art der im Blockschaltbild verwendeten Signale:

Wertkontinuierliche Signale	Wertdiskrete Signale
Gaspedalwinkel Drehmoment Antriebsmoment Fahrwiderstand Drehmoment auf die Räder Geschwindigkeit Bremspedalwinkel Bremsmoment	Kupplung (offen/geschlossen) Wahl des Gangs (1-6, Rückwärtsgang)

3. Die Geschwindigkeitsregelung verwendet die aktuelle Fahrzeuggeschwindigkeit als Regelgröße, vergleicht diese mit einer Sollgeschwindigkeit und verändert den Gaspedalwinkel (oder das Vergaser-ventil bzw. die Einspritzmenge) so, dass die Regelabweichung kleiner wird. Der Fahrer gibt jetzt den Geschwindigkeitssollwert vor (Abb. A.5).

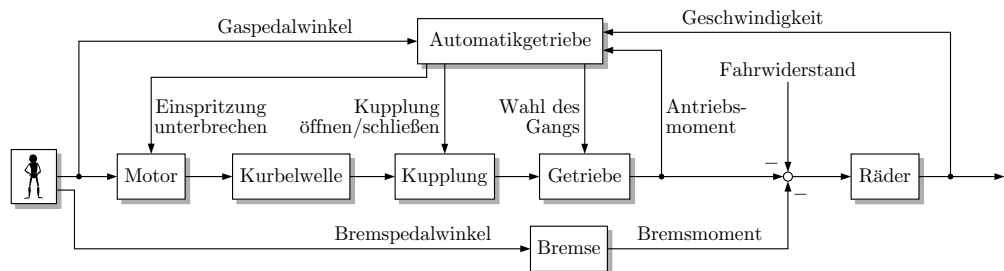


Abb. A.4: Blockschaltbild mit Automatikgetriebe

4. Automatikgetriebe beeinflussen den Motor (Einspritzung des Treibstoffes), die Kupplung und das Getriebe in Abhängigkeit von der aktuellen Geschwindigkeit, dem Antriebsmoment und dem Gaspedalwinkel, der den „Fahrerwunsch“ (z. B. „Beschleunigung“ oder „konstante Geschwindigkeit“) erkennen lässt. Im Blockschaltbild (Abb. A.4) sieht man, dass das zunächst aus einer Reihenschaltung bestehende System dadurch eine Rückkopplungsstruktur erhält.

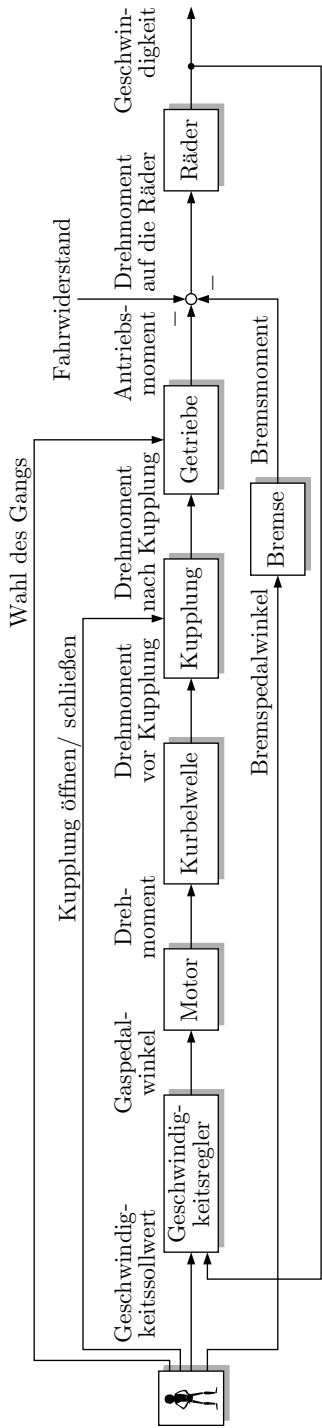


Abb. A.5: Blockschaltbild mit Geschwindigkeitsregler

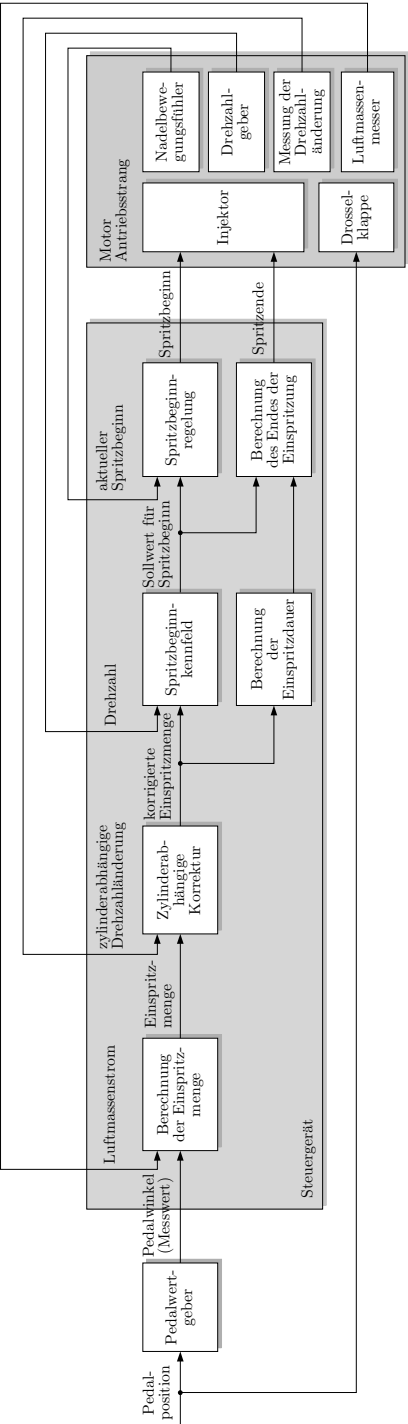


Abb. A.6: Blockschaltbild einer Motorsteuerung

**Diskussion.** Das Denken in Blockschaltbildern ist für die Automatisierungstechnik typisch. Bei allen Überwachungs- und Steuerungsaufgaben muss zunächst geklärt werden, auf welche Eingangs- und Ausgangssignale sich die Automatisierungseinrichtung stützen kann und welche strukturellen Eigenschaften das betrachtete physikalische System hat. Die in dieser Aufgabe behandelten Beispiele zeigen, dass man das Blockschaltbild auch mit oberflächlichen Kenntnissen über den Systemaufbau zeichnen kann und dass daraus wichtige Schlussfolgerungen für die Automatisierungsaufgabe und das automatisierte System abgeleitet werden können.

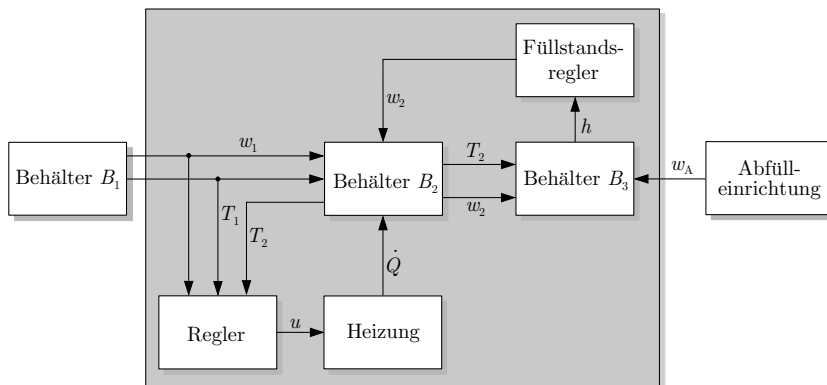
### Aufgabe 2.2 Blockschaltbild einer Motorsteuerung

Abbildung A.6 auf S. 573 zeigt das Blockschaltbild, das aus zwei großen Blöcken besteht, die das Steuergerät und den Motor einschließlich Antriebsstrang darstellen. Eingangsgrößen des Motors sind die Zeitpunkte des Beginns und des Endes des Einspritzens sowie die Drosselklappenstellung. Für die Motorsteuerung sind vier Messgrößen wichtig: der mit dem Nadelbewegungsfühler gemessene aktuelle Einspritzbeginn, die Motordrehzahl, die zylinderabhängige Änderung der Drehzahl sowie der gemessene Luftmassenstrom.

Im Steuergerät wird, ausgehend vom gemessenen Pedalwinkel als Beschreibung des gewünschten Drehmomentes, die Einspritzmenge berechnet, zylinderabhängig korrigiert und in die Zeitpunkte des Spritzbeginns und des Spritzendes umgerechnet.

### Aufgabe 2.3 Analyse der erweiterten Abfüllanlage

Der Füllstandsregler gibt in Abhängigkeit vom aktuellen Füllstand  $h$  die pro Zeiteinheit aus dem Behälter  $B_2$  in den Behälter  $B_3$  laufende Flüssigkeitsmenge vor. Dadurch muss jetzt zwischen dem Flüssigkeitsstrom  $w_1$  von  $B_1$  in  $B_2$  und  $w_2$  von  $B_2$  in  $B_3$  unterschieden werden. Der aktuelle Füllstand, der von der Abfülleinrichtung beeinflusst wird, wirkt jetzt auf den Behälter  $B_2$  (Abb. A.7).



**Abb. A.7:** Kopplungsstruktur der erweiterten Abfüllanlage

Diese neue Rückkopplung führt dazu, dass jetzt die Behälter  $B_2$  und  $B_3$ , beide Regler und die Heizung stark zusammenhängend sind. Damit wirkt beispielsweise die Abfüllmenge  $w_A$  auch auf die Heizung des zweiten Behälters ein.

**Diskussion.** Dieses Beispiel zeigt, dass Zwischenspeicher wie der ungeregelte Behälter  $B_3$  Teilanlagen entkoppeln. Da sich in der bisherigen Verwendung des Behälters  $B_3$  der Füllstand nach der Produktion und der Abfüllmenge eingestellt hat und keinen Vorgaben unterlag, waren die Behälter  $B_2$  und  $B_3$  mit der Abfülleinrichtung nur in jeweils einer Richtung verkoppelt. Beim Entwurf des Temperaturreglers musste deshalb nur der Behälter  $B_2$  zusammen mit der Heizung betrachtet werden. Wenn jetzt der Behälter  $B_3$  einen geregelten Füllstand haben soll, so übertragen sich Störungen in der Abfülleinrichtung über den Behälter  $B_3$  auf den Behälter  $B_2$  und beeinflussen dort den Aufwärmprozess. Um den Temperaturregler entwerfen zu können, muss man ein Modell der gekoppelten Behälter  $B_2$  und  $B_3$  und der Heizung aufstellen und man muss den Füllstandsregler kennen.

#### Aufgabe 2.4 Kopplungsanalyse eines Elektroenergieversorgungsnetzes

1. Alle Kraftwerke und Verbraucher sind stark zusammenhängend. Der Grund liegt in der Rückwirkung der Frequenz  $f$  auf die Verbraucher und die Kraftwerke.
2. Wird die Primärregelung entfernt, sind die Kraftwerke nicht mehr stark zusammenhängend, da nun die Frequenz  $f$  nicht mehr zu den Kraftwerken zurückgeführt wird. Weiterhin stark zusammenhängend sind allerdings die Verbraucher.
3. Das Netz wird jetzt wieder mit Primärregelung betrachtet. Aus dem Blockschaltbild A.8 und Aufgabenteil 1 folgt, dass alle Blöcke stark zusammenhängend sind.

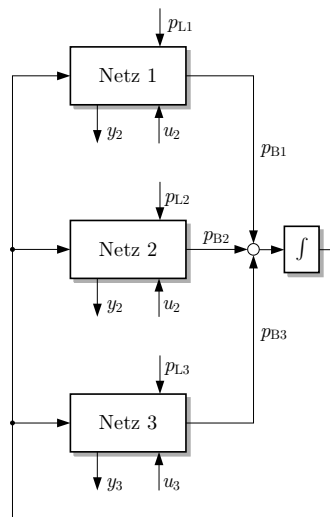


Abb. A.8: Elektroenergiesystem mit drei Teilnetzen

4. Wenn man annimmt, dass die Netzfrequenz konstant ist, entfällt im Blockschaltbild die Rückkopplung über die Frequenz. Damit sind die drei Teilnetze nicht mehr stark zusammenhängend und können getrennt voneinander modelliert, analysiert und geregelt werden.

5. Die Annahme des starren Netzes geht von einer konstanten Frequenz aus. Für die Sekundärregelung wirkt die Netzfrequenz jedoch als Regelgröße. Man kann deshalb nicht mit der Annahme des starren Netzes arbeiten.

### Aufgabe 2.5 Steuerung einer Abfüllanlage

1. Ein Beispiel für die hier betrachtete Abfüllanlage ist der Kaffeeautomat in der Kantine. Der Abfüllvorgang funktioniert so, dass der Kaffeeautomat nach einem Knopfdruck eine bestimmte Menge des bereits gekochten Kaffees in den darunter gestellten Becher abgibt. Die Menge des abgegebenen Kaffees wird über die Zeitdauer  $t_e$  bestimmt, über die das Ventil des Automaten geöffnet ist (siehe Abb. A.9 (oben)). Für die Stellgröße gilt

$$u(t) = \begin{cases} \bar{u} & \text{für } 0 \leq t \leq t_e \\ 0 & \text{sonst.} \end{cases} \quad (\text{A.1})$$

Das Volumen des abgefüllten Kaffees im Becher beträgt demnach

$$V(t_e) = \int_0^{t_e} u(t) dt + V_0 = \bar{u} \cdot t_e + V_0,$$

wobei  $\bar{u}$  den als konstant angenommenen Volumenstrom bei geöffnetem Ventil bezeichnet und  $V_0$  das Kaffeevolumen bezeichnet, das zu Beginn des Abfüllvorgangs im Becher ist (also üblicherweise  $V_0 = 0$ ).

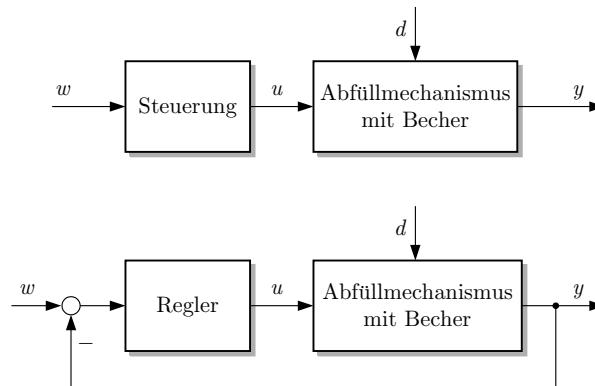


Abb. A.9: Blockschaltbild eines Kaffeeautomaten

Modellinformationen über den Abfüllvorgang stecken in den Größen  $\bar{u}$  und  $V$ , aus denen die Abfüllzeit  $t_e = \frac{V}{\bar{u}}$  so festgelegt wird, dass der Becher bis zu einer gewünschten Höhe, also mit einem vorgegebenen Volumen  $v_e$  gefüllt wird ( $V(t_e) \stackrel{!}{=} V_e$ ).

Wenn man keinen Standardbecher oder einen bereits teilweise gefüllten Becher verwendet, kann dies nicht von der Steuerung berücksichtigt werden, da keinerlei Information über die Größe des gerade verwendeten Bechers oder über den aktuellen Füllstand zur Steuerung gelangt. Der Becher läuft also über oder ist nicht ausreichend gefüllt. Auch wenn eine Störung  $d(t)$  den Volumenstrom durch das Abfüllventil verändert, wird der Becher im Zeitintervall  $[0, t_e]$  nicht in gewünschter Weise gefüllt.

2. Eine Abfüllanlage mit Regelung misst die aktuelle Füllhöhe  $y(t)$  und schaltet den Abfüllvorgang ab, wenn der Sollwert  $w$  erreicht ist. An Stelle von Gl. (A.1) gilt jetzt

$$u(t) = \begin{cases} \bar{u} & \text{für } y(t) \leq w \\ 0 & \text{sonst.} \end{cases}$$

Damit ändert sich das Blockschaltbild aus Abb. A.9 (oben) in die darunter gezeigte Struktur. Die Abfüllanlage reagiert jetzt auf veränderte Bechergröße, bereits im Becher vorhandenen Kaffee oder den durch eine Störung veränderten Durchfluss des Kaffees während der Abfüllung.

### Aufgabe 2.6 Blockschaltbild einer Raumtemperaturregelung

Das Blockschaltbild ist in Abb. A.10 zu sehen. Es handelt sich um eine Steuerung im geschlossenen Wirkungskreis.

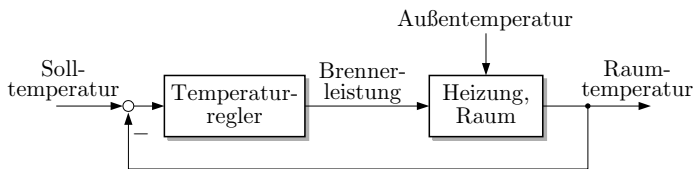


Abb. A.10: Blockschaltbild der Raumtemperaturregelung

Die Regelgröße ist die Raumtemperatur. Um den Istwert der Raumtemperatur der vorgegebenen Solltemperatur anzupassen, verstellt der Regler die Brennerleistung. Als Regelstrecke wirken folglich alle Teilsysteme, die gemeinsam die dynamische Kopplung zwischen der Brennerleistung und der Raumtemperatur herstellen, nämlich der Heizkessel, die Heizungsrohre, der Heizkörper und der Raum. Die Raumtemperatur wird ferner durch die Außentemperatur beeinflusst, die als Störung auf die Regelstrecke wirkt.

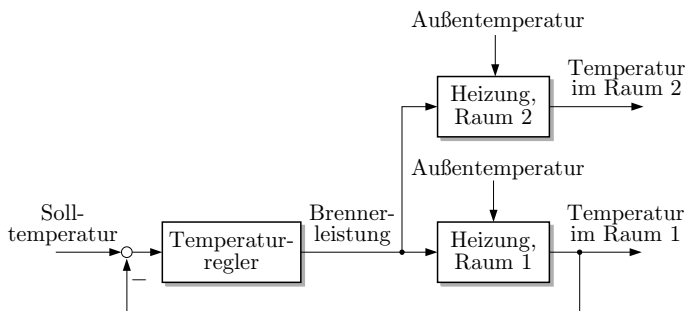
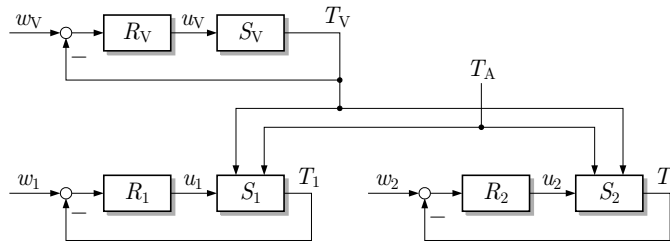


Abb. A.11: Erweitertes Blockschaltbild der Raumtemperaturregelung

Die Temperatur von Raum 2 hängt über die Vorlauftemperatur von der Brennerleistung und von der Außentemperatur ab (Abb. A.11). Die Temperaturregelung für Raum 1 hat eine Wirkung auf diese Raumtemperatur, aber man kann weder von einer Regelung noch von einer Steuerung sprechen, da die Beeinflussung der Temperatur von Raum 2 nicht zielgerichtet erfolgt. Da der Raum 2 andere Eigenschaften als der Raum 1 hat, unterscheidet sich die Temperatur in beiden Räumen.



**Abb. A.12:** Erweitertes Blockschaltbild der Raumtemperaturregelung

Bei getrennter Raumtemperaturregelung werden die Temperaturen  $T_1$  und  $T_2$  durch zwei Regler  $R_1$  und  $R_2$  den Sollwerten  $w_1$  bzw.  $w_2$  angepasst, wobei die Ventilstellungen  $u_1$  bzw.  $u_2$  der beiden Heizkörper als Stellgrößen dienen. Das Verhalten beider Regelkreise wird durch die Vorlauftemperatur  $T_V$  und die Außentemperatur  $T_A$  beeinflusst, die für diese Regelkreise zwei Störgrößen darstellen (Abb. A.12).

Der Vorlauftemperaturregler  $R_V$  verwendet als Stellgröße  $u_V$  die Brennerleistung, um die Vorlauftemperatur  $T_V$  auf den Sollwert  $w_V$  zu bringen.

**Diskussion.** Streng genommen sind die in Abb. A.12 gezeigten drei Regelkreise nicht nur in einer Richtung untereinander verkoppelt, weil die Raumtemperaturregelkreise die Vorlauftemperatur beeinflussen. So bewirkt eine niedrige Außentemperatur eine große Wärmeentnahme aus dem Heizkreislauf und damit ein Absinken der Vorlauftemperatur. Diese Kopplung kann man in das angegebene Blockschaltbild zusätzlich eintragen.

Bei vielen Heizungen wird die Vorlauftemperatur in Abhängigkeit von der Außentemperatur festgelegt, was im Blockschaltbild durch einen zusätzlichen Block mit der Eingangsgröße  $T_A$  und der Ausgangsgröße  $w_V$  berücksichtigt werden kann.

### Aufgabe 3.1 Zustandsraummodell einer elektrischen Schaltung

1. Für die dynamische Ordnung der Schaltung sind nur die beiden Kondensatoren maßgebend, weil sie – im Gegensatz zu Widerständen – Energie speichern können. Die dynamische Ordnung ist also gleich zwei. Geeignete Zustandsvariablen sind beispielsweise die beiden Kondensatorspannungen  $u_{C_1}(t)$  und  $u_{C_2}(t)$ :

$$\mathbf{x}(t) = \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix}.$$



2. Es wird entsprechend Algorithmus 3.1 vorgegangen:

1. **Systemzerlegung:** Das System besteht aus drei Widerständen und zwei Kondensatoren, deren physikalische Wirkungsweise durch algebraische bzw. Differentialgleichungen beschrieben werden. Diese Bauelemente sind die im Folgenden betrachteten Komponenten.

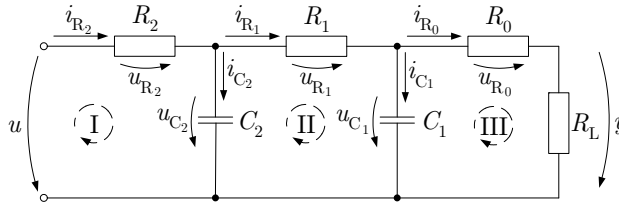


Abb. A.13: Elektrische Schaltung mit den im Modell verwendeten Spannungen und Strömen

2. **Beschreibung der Komponenten:** Die elektrischen Bauelemente sind durch folgende Gleichungen beschrieben:

$$y(t) = R_L i_{R_0}(t) \quad (\text{A.2})$$

$$u_{R_0}(t) = R_0 i_{R_0}(t) \quad (\text{A.3})$$

$$u_{R_1}(t) = R_1 i_{R_1}(t) \quad (\text{A.4})$$

$$u_{R_2}(t) = R_2 i_{R_2}(t) \quad (\text{A.5})$$

$$i_{C_1}(t) = C_1 \frac{du_{C_1}(t)}{dt}, \quad u_{C_1}(0) = u_{C_1 0} \quad (\text{A.6})$$

$$i_{C_2}(t) = C_2 \frac{du_{C_2}(t)}{dt}, \quad u_{C_2}(0) = u_{C_2 0} \quad (\text{A.7})$$

(für die Bezeichnungen siehe Abb. A.13).

3. **Beschreibung der Kopplungen:** Die Kopplungen zwischen den Bauelementen werden durch die Kirchhoffschen Gesetze beschrieben. In den beiden Knotenpunkten gilt der Knotenpunktsatz:

$$i_{R_2}(t) - i_{C_2}(t) - i_{R_1}(t) = 0 \quad (\text{A.8})$$

$$i_{R_1}(t) - i_{C_1}(t) - i_{R_0}(t) = 0. \quad (\text{A.9})$$

Für die in Abb. A.13 eingetragenen Maschen gelten folgende Beziehungen:

$$u_{C_2}(t) - u(t) + u_{R_2}(t) = 0 \quad (\text{A.10})$$

$$u_{C_1}(t) - u_{C_2}(t) + u_{R_1}(t) = 0 \quad (\text{A.11})$$

$$y(t) - u_{C_1}(t) + u_{R_0}(t) = 0. \quad (\text{A.12})$$

4. **Aufstellung des Zustandsraummodells:** Die Gln. (A.2) – (A.12) beschreiben die gegebene elektrische Schaltung vollständig. Die folgenden Umformungen dienen dazu, aus diesen Gleichungen ein Zustandsraummodell abzuleiten. Dafür müssen alle diejenigen Signale eliminiert werden, die

nicht Zustandsvariable, Eingangsgröße oder Ausgangsgröße sind, also  $u_{R_0}$ ,  $u_{R_1}$ ,  $u_{R_2}$ ,  $i_{R_0}$ ,  $i_{R_1}$ ,  $i_{R_2}$ ,  $i_{C_1}$  und  $i_{C_2}$ .

Aus den Gln. (A.3) – (A.5) für die Widerstände und den drei Maschengleichungen (A.10) – (A.12) können zunächst die Ströme durch die drei Widerstände ausgerechnet werden. Aus den Gln. (A.5) und (A.10) erhält man

$$R_2 i_{R_2}(t) = u(t) - u_{C_2}(t)$$

und

$$i_{R_2}(t) = \frac{u(t) - u_{C_2}(t)}{R_2},$$

aus (A.4) und (A.11)

$$i_{R_1}(t) = \frac{u_{C_2}(t) - u_{C_1}(t)}{R_1}$$

und aus (A.2), (A.3) und (A.12)

$$i_{R_0}(t) = \frac{u_{C_1}(t)}{R_0 + R_L}.$$

Durch Einsetzen dieser Ströme in die Knotenpunktgleichungen (A.8) und (A.9) und unter Verwendung der Kondensatorgleichungen (A.6) bzw. (A.7) erhält man die Zustandsgleichungen:

$$\begin{aligned} i_{C_2}(t) &= i_{R_2}(t) - i_{R_1}(t) \\ C_2 \frac{du_{C_2}(t)}{dt} &= \frac{u(t) - u_{C_2}(t)}{R_2} - \frac{u_{C_2}(t) - u_{C_1}(t)}{R_1} \\ \frac{du_{C_2}(t)}{dt} &= \frac{u(t) - u_{C_2}(t)}{C_2 R_2} - \frac{u_{C_2}(t) - u_{C_1}(t)}{C_2 R_1} \\ &= -\left(\frac{1}{C_2 R_2} + \frac{1}{C_2 R_1}\right) u_{C_2}(t) + \frac{1}{C_2 R_1} u_{C_1}(t) + \frac{1}{C_2 R_2} u(t) \end{aligned}$$

und

$$\begin{aligned} i_{C_1}(t) &= i_{R_1}(t) - i_{R_0}(t) \\ C_1 \frac{du_{C_1}(t)}{dt} &= \frac{u_{C_2}(t) - u_{C_1}(t)}{R_1} - \frac{u_{C_1}(t)}{R_L + R_0} \\ \frac{du_{C_1}(t)}{dt} &= \frac{u_{C_2}(t) - u_{C_1}(t)}{C_1 R_1} - \frac{u_{C_1}(t)}{C_1 (R_L + R_0)} \\ &= -\left(\frac{1}{C_1 R_1} + \frac{1}{C_1 (R_L + R_0)}\right) u_{C_1}(t) + \frac{1}{C_1 R_1} u_{C_2}(t). \end{aligned}$$

Daraus folgt das Zustandsraummodell

$$\text{Schaltung : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{u}_{C_1}(t) \\ \dot{u}_{C_2}(t) \end{pmatrix} = \begin{pmatrix} -\left(\frac{1}{R_1 C_1} + \frac{1}{C_1(R_L + R_0)}\right) & \frac{1}{R_1 C_1} \\ \frac{1}{R_1 C_2} & -\left(\frac{1}{R_2 C_2} + \frac{1}{R_1 C_2}\right) \end{pmatrix} \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} \\ + \begin{pmatrix} 0 \\ \frac{1}{R_2 C_2} \end{pmatrix} u(t) \\ y(t) = \begin{pmatrix} \frac{R_L}{R_L + R_0} & 0 \end{pmatrix} \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix}, \end{array} \right.$$

dessen Ausgangsgleichung man aus Gl. (A.2) erhält. Der Anfangszustand

$$\mathbf{x}_0 = \begin{pmatrix} u_{C_1 0} \\ u_{C_2 0} \end{pmatrix}$$

ist durch die als Anfangswerte der Differentialgleichungen (A.6) und (A.7) eingeführten Kondensatorspannungen zur Zeit  $t = 0$  festgelegt.

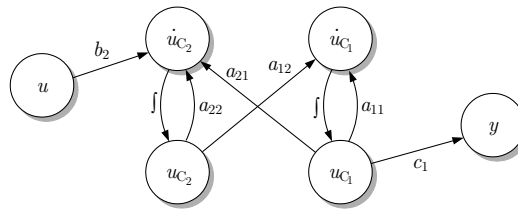


Abb. A.14: Signalflussgraph der elektrischen Schaltung

3. Aus dem Zustandsraummodell erhält man den in Abb. A.14 gezeigten Signalflussgraphen. Die Kantenengewichte sind durch die Elemente der Systemmatrix  $\mathbf{A}$  sowie des Eingangsvektors  $\mathbf{b}$  und des Ausgangsvektors  $\mathbf{c}^T$  gegeben, die ihrerseits von den Parametern  $R_L$ ,  $R_0$ ,  $R_1$ ,  $R_2$ ,  $C_1$  und  $C_2$  abhängen. Wie man aus dem Zustandsraummodell ablesen kann, gilt dabei beispielsweise  $a_{12} = \frac{1}{R_1 C_1}$ . Zwischen  $u(t)$  und  $\dot{x}_1(t)$  sowie zwischen  $u_{C_2}(t)$  und  $y(t)$  gibt es keine Kanten, weil das erste Element im Vektor  $\mathbf{b}$  bzw. das zweite Element im Vektor  $\mathbf{c}^T$  gleich null ist.

Wenn man wie in Abb. A.14 die zur Zustandsvariablen  $u_{C_2}(t)$  gehörenden Knoten links und die zur Zustandsvariablen  $u_{C_1}(t)$  gehörenden Knoten rechts anordnet, so sieht man genau wie in der Schaltung, dass die Eingangsspannung zunächst die Spannung des Kondensators  $C_2$  und über diesen die Spannung des Kondensators  $C_1$  beeinflusst. Dynamisch beeinflussen sich beide Kondensatoren wechselseitig, denn die Auf- und Entladevorgänge können auf Grund der elektrischen Kopplung zwischen beiden Bauelementen nicht unabhängig voneinander ablaufen.

4. Alle im Zustandsraummodell auftretenden Signale  $u(t)$ ,  $y(t)$ ,  $u_{C_1}(t)$  und  $u_{C_2}(t)$  sind Spannungen, die in Volt gemessen werden. Die Zeit hat die Einheit Sekunden. Die Elemente der Systemmatrix  $\mathbf{A}$  und des Vektors  $\mathbf{b}$  müssen deshalb die Maßeinheit  $\frac{1}{s}$  haben, denn auf der linken Seite der Zustandsgleichung steht die zeitliche Ableitung der Kondensatorspannungen. Diese Maßeinheit erhält man tatsächlich, wenn man das im Nenner aller Elemente auftretende Produkt aus Kapazität und ohmschem

Widerstand berechnet, wobei für die Maßeinheiten die Beziehung  $\frac{\Delta V}{V_{As}} = s^{-1}$  gilt. Die Elemente des Vektors  $c^T$  sind einheitenlos, denn auf beiden Seiten der Ausgabegleichung stehen Spannungen.

### Aufgabe 3.2 Zustandsraummodell des invertierten Pendels

Nach dem Drallsatz ist der Drehimpuls gleich dem resultierenden Moment der Kräfte an einem Massensystem. Beim Pendel treten die in Abb. A.15 eingetragenen Kräfte und Momente auf. Die Schwerkraft  $F_G = mg$  führt auf das Moment

$$M_G(t) = F_G s(t),$$

wobei man den senkrechten Abstand  $s$  des Massenschwerpunkts vom Drehpunkt anhand der Beziehung

$$s(t) = \frac{l}{2} \sin \phi(t) \approx \frac{l}{2} \phi(t)$$

ermitteln kann. Diese Näherung gilt für kleine Winkel. Die Eingangsgröße  $u(t)$  ist das vom Motor ausgeübte Drehmoment. Folglich ist das Pendel durch die Differentialgleichung

$$\begin{aligned} J\ddot{\phi}(t) &= M_G(t) + u(t) \\ &= \frac{mgl}{2}\phi(t) + u(t) \\ \ddot{\phi}(t) &= \frac{3}{2}\frac{g}{l}\phi(t) + \frac{3}{ml^2}u(t), \quad \dot{\phi}(0) = \dot{\phi}_0 \end{aligned}$$

beschrieben, wobei  $\phi_0$  den Winkel und  $\dot{\phi}_0$  die Winkelgeschwindigkeit zur Zeit  $t = 0$  beschreiben.

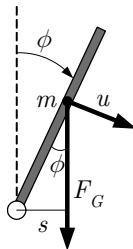


Abb. A.15: Kräfte am invertierten Pendel

Um zu einem Zustandsraummodell zu kommen, muss ein geeigneter Systemzustand gewählt werden. Da die Differentialgleichung von zweiter Ordnung ist, braucht man zwei Zustandsvariablen. Wählt man  $x_1(t) = \phi(t)$  und  $x_2(t) = \dot{\phi}(t)$ , so ergibt sich die Zustandsgleichung

$$\begin{pmatrix} \dot{\phi}(t) \\ \ddot{\phi}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{3}{ml^2} \end{pmatrix} u(t).$$

Wird der Pendelwinkel  $\phi(t)$  als Messgröße verwendet, so muss noch die Ausgabegleichung

$$\phi(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}$$

aufgeschrieben werden. In der Standardform geschrieben, heißt das Modell dann

$$\text{Pendel : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ \frac{3}{ml^2} \end{pmatrix} u(t), & \mathbf{x}(0) = \begin{pmatrix} \phi_0 \\ \dot{\phi}_0 \end{pmatrix} \\ y(t) = (1 \ 0) \mathbf{x}(t). \end{cases} \quad (\text{A.13})$$

### Aufgabe 3.3 Zustandsraummodell eines Radioteleskops

Das Modell erhält man aus der Grundgleichung einer Drehbewegung

$$J\ddot{\phi}(t) = u(t) - M_A(t) + M_G(t) - M_R(t), \quad \phi(0) = \phi_0, \quad \dot{\phi}(0) = \dot{\phi}_0$$

mit den folgenden Bezeichnungen:

- $J$  – Trägheitsmoment der Antenne,
- $u(t)$  – Antriebsmoment in  $\phi$ -Richtung,
- $M_A(t)$  – Moment, das die Masse  $m_A$  entgegen der positiven Richtung von  $\phi(t)$  erzeugt,
- $M_G(t)$  – Moment, das der Balastausleger mit der Masse  $m_G$  in  $\phi$ -Richtung erzeugt,
- $M_R(t)$  – Reibmoment,
- $\phi_0, \dot{\phi}_0$  – Anfangswerte des Winkels und der Winkelgeschwindigkeit.

Der in der Stellung  $\phi(t)$  des Teleskops wirkende Hebelarm für die Masse  $m_A$  hat die Länge  $l_A \cos \phi(t)$ , so dass sich das Moment  $M_A(t)$  aus

$$M_A(t) = g m_A l_A \cos \phi(t)$$

ergibt, wobei  $g$  die Erdbeschleunigung darstellt. Analog dazu erhält man für das durch den Balastausleger erzeugte Moment die Beziehung

$$M_G(t) = g m_G l_G \cos \phi(t).$$

Für das Reibmoment wird entsprechend der Aufgabenstellung der Ansatz

$$M_R(t) = k_R \dot{\phi}(t)$$

mit dem Proportionalitätsfaktor  $k_R$  gemacht. Damit erhält man die nichtlineare Differentialgleichung zweiter Ordnung

$$\text{Radioteleskop: } J\ddot{\phi}(t) + k_R \dot{\phi}(t) + g m_A l_A \cos \phi(t) - g m_G l_G \cos \phi(t) = u(t).$$

Für die Umformung der Differentialgleichung in ein Zustandsraummodell müssen zwei Zustandsvariablen gewählt werden. Dass es *zwei* Variablen sind, ergibt sich aus der Tatsache, dass die höchste vorkommende Ableitung von  $\phi(t)$  die Ordnung zwei hat. Es soll hier mit dem Zustand

$$\mathbf{x}(t) = \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}$$

gearbeitet werden. Aus dieser Definition des Zustandsvektors ergibt sich die Ausgangsgleichung

$$y(t) = x_1(t)$$

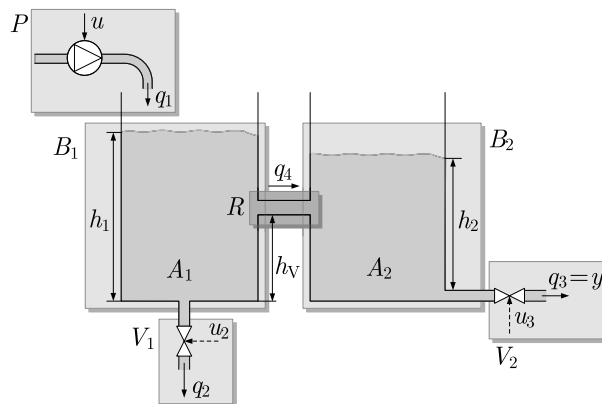
und das Zustandsraummodell

$$\text{Radioteleskop : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} x_2(t) \\ \frac{g(m_G l_G - m_A l_A)}{J} \cos x_1(t) - \frac{k_R}{J} x_2(t) + \frac{1}{J} u(t) \end{pmatrix} \\ \begin{pmatrix} \phi(0) \\ \dot{\phi}(0) \end{pmatrix} = \begin{pmatrix} \phi_0 \\ \dot{\phi}_0 \end{pmatrix} \\ y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}. \end{array} \right.$$

### Aufgabe 3.4 Kontinuierliches Modell eines Behältersystems

Die Anwendung von Algorithmus 3.2 auf S. 92 führt auf die folgenden vier Lösungsschritte:

- 1. Systemzerlegung:** Das Behältersystem wird entsprechend Abb. A.16 in sechs Komponenten zerlegt, wobei die Komponente  $R$  das Verbindungsrohr bezeichnet. Bei der Zerlegung müssen die in der Abbildung angegebenen Koppelsignale eingeführt werden.



**Abb. A.16:** Komponenten des Behältersystems mit den verwendeten Bezeichnungen

- 2. Beschreibung der Komponenten:** In diesem Schritt wird jede Komponente für sich betrachtet und es werden die physikalischen Wirkprinzipien durch die folgenden Gleichungen beschrieben.

- Pumpe  $P$ : Der Volumenstrom ist proportional zur Eingangsgröße  $u(t)$ :

$$q_1(t) = s_1 u(t),$$

wobei  $s_1$  einen Proportionalitätsfaktor darstellt, der von den Pumpenparametern abhängt.

- Behälter  $B_1$ : Der Füllstand  $h_1(t)$  verändert sich in Abhängigkeit von den drei in der Abbildung eingetragenen Flüssen  $q_1(t)$ ,  $q_2(t)$  und  $q_4(t)$  entsprechend der folgenden Differentialgleichung

$$\dot{h}_1(t) = \frac{1}{A_1} (q_1(t) - q_2(t) - q_4(t)), \quad h_1(0) = h_{10}.$$

$h_{10}$  bezeichnet den Füllstand zur Zeit  $t = 0$ . Der Druck  $p_1(t)$  am Boden des Behälters berechnet sich nach dem Toricelli-Gesetz entsprechend

$$p_1(t) = \sqrt{2g h_1(t)}.$$

Da die Füllhöhe über dem Verbindungsrohr den Wert  $h_1(t) - h_V$  hat, erhält man für den Wasserdruck am Verbindungsrohr auf der Seite des linken Behälters

$$p_{21}(t) = \begin{cases} \sqrt{2g(h_1(t) - h_V)} & \text{für } h_1(t) \geq h_V \\ 0 & \text{für } h_1(t) < h_V. \end{cases}$$

- Behälter  $B_2$ : Es gelten die Gleichungen

$$\begin{aligned} \dot{h}_2(t) &= \frac{1}{A_2} (q_4(t) - q_3(t)), \quad h_2(0) = h_{20} \\ p_2(t) &= \sqrt{2g h_2(t)}. \\ p_{12}(t) &= \begin{cases} \sqrt{2g(h_2(t) - h_V)} & \text{für } h_2(t) \geq h_V \\ 0 & \text{für } h_2(t) < h_V. \end{cases} \end{aligned}$$

- Verbindungsrohr  $R$ : Der Durchfluss  $q_4(t)$  durch das Rohr hängt vom Druckabfall  $\Delta p(t)$  über dem Rohr ab:

$$q_4(t) = s_4 \Delta p(t).$$

- Ventil  $V_1$ : Mit dem Proportionalitätsfaktor  $s_2$  ist der Durchfluss durch das Ventil proportional zum Vordruck  $p_{V_1}(t)$ :

$$q_2(t) = s_2 p_{V_1}(t).$$

- Ventil  $V_2$ : Es gilt

$$q_3(t) = s_3 p_{V_2}(t).$$

- 3. Beschreibung der Kopplungen:** Da in den Modellen der Komponenten für einige Koppelsignale bereits dieselben Bezeichnungen verwendet wurden, sind einige Kopplungen implizit in den Komponentenmodellen berücksichtigt, beispielsweise die Tatsache, dass der Zufluss  $q_1(t)$  des linken Behälters gleich dem Ausfluss aus der Pumpe ist. In gleicher Weise bezeichnet  $q_4(t)$  den Abfluss vom linken Behälter in das Verbindungsrohr und gleichzeitig den Fluss durch das Verbindungsrohr und den entsprechenden Zufluss zum rechten Behälter. Für andere Signale muss die Kopplung durch die folgenden Beziehungen explizit beschrieben werden:

$$\begin{aligned} p_{V_1}(t) &= p_1(t) \\ p_{V_2}(t) &= p_2(t) \\ \Delta p(t) &= p_{21}(t) - p_{12}(t). \end{aligned}$$

- 4. Aufstellung des Zustandsraummodells:** Die oben angegebenen Gleichungen beschreiben das Behältersystem vollständig. Die folgenden Umformungen haben das Ziel, aus diesen Gleichungen ein Zustandsraummodell abzuleiten.

Da die beiden Behälter die einzigen Speicherelemente des Systems sind, hat das Modell die dynamische Ordnung  $n = 2$ . In den o. a. Gleichungen treten mit  $h_1(t)$  und  $h_2(t)$  zwei Ableitungen auf, die es nahelegen, die beiden Füllstände als Zustandsvariablen zu wählen:  $\mathbf{x}(t) = (h_1(t) \ h_2(t))^T$ .

Durch Einsetzen der Koppelbeziehungen erhält man das folgende Modell, in dem der Durchfluss  $q_4(t)$  durch das Verbindungsrohr entsprechend Gl. (A.15) von den beiden Füllständen  $h_1(t)$  und  $h_2(t)$  abhängig ist.

$$\text{Behälter : } \begin{cases} \begin{pmatrix} \dot{h}_1(t) \\ \dot{h}_2(t) \end{pmatrix} = \begin{pmatrix} \frac{1}{A_1} (s_1 u(t) - s_2 \sqrt{2gh_1(t)} - q_4(t)) \\ \frac{1}{A_2} (q_4(t) - s_3 \sqrt{2gh_2(t)}) \end{pmatrix}, \begin{pmatrix} h_1(0) \\ h_2(0) \end{pmatrix} = \begin{pmatrix} h_{10} \\ h_{20} \end{pmatrix} \\ y(t) = s_3 \sqrt{2gh_2(t)} \end{cases} \quad (\text{A.14})$$

mit

$$q_4(t) = \begin{cases} 0 & \text{wenn } h_1(t) < h_V, h_2(t) < h_V \\ s_4 \sqrt{2g(h_1(t) - h_V)} & \text{wenn } h_1(t) \geq h_V, h_2(t) < h_V \\ -s_4 \sqrt{2g(h_2(t) - h_V)} & \text{wenn } h_1(t) < h_V, h_2(t) \geq h_V \\ s_4 \operatorname{sgn}(h_1(t) - h_2(t)) (\sqrt{2gh_1(t)} - \sqrt{2gh_2(t)}) & \text{wenn } h_1(t) \geq h_V, h_2(t) \geq h_V. \end{cases} \quad (\text{A.15})$$

**Diskussion.** Wenn man die beiden Ventile  $V_1$  und  $V_2$  ansteuert, ändern sich deren Ventilkonstanten  $s_2$  und  $s_3$  in Abhängigkeit von den neuen Eingangsgrößen und müssen durch die Produkte  $s_2 u_2(t)$  bzw.  $s_3 u_3(t)$  mit  $u_i(t) \in [0, 1]$ , ( $i = 2, 3$ ) ersetzt werden. Das System hat dann insgesamt drei Eingangsgrößen und die nichtlinearen Zusammenhänge werden noch komplizierter, weil Produkte zwischen den neuen Eingangsgrößen und den Zustandsvariablen auftreten.

### Aufgabe 3.5 Linearisierung des Modells eines Unterwasserfahrzeugs

Wie im Beispiel 3.8 gezeigt wurde, müssen die Arbeitspunktwerte  $\bar{u}$  und  $\bar{x}$  die Beziehungen

$$\begin{aligned} \bar{u} &= 0 \\ \bar{x}_3 &= 0 \\ \bar{x}_2 &= \frac{m - \bar{x}_1 \rho_0}{\bar{x}_1 \rho_0^2 (\kappa_W - \kappa_D) g} \end{aligned}$$

erfüllen.  $\bar{x}_1 = V_0$  ist das durch die Kolbenstellung vorgebbare Volumen des Drifters in der Tiefe  $z = 0$ . Für die in das lineare Modell eingehenden Ableitungen erhält man

$$\begin{aligned} \frac{\partial g_1}{\partial x_1} &= 0, \quad \frac{\partial g_1}{\partial x_2} = 0, \quad \frac{\partial g_1}{\partial x_3} = 0 \\ \frac{\partial g_2}{\partial x_1} &= 0, \quad \frac{\partial g_2}{\partial x_2} = 0, \quad \frac{\partial g_2}{\partial x_3} = 1 \end{aligned}$$

$$\begin{aligned} \frac{\partial g_3}{\partial x_1} &= -\frac{\rho_0 g}{m} (1 + (\kappa_W - \kappa_D) g \rho_0 \bar{x}_2) = -\frac{g}{\bar{x}_1} \\ \frac{\partial g_3}{\partial x_2} &= -\frac{\rho_0^2 g^2}{m} (\kappa_W - \kappa_D) \bar{x}_1 \\ \frac{\partial g_3}{\partial x_3} &= -\frac{k_w}{m} \end{aligned}$$

sowie

$$\frac{\partial g_1}{\partial u} = k, \quad \frac{\partial g_2}{\partial u} = 0, \quad \frac{\partial g_3}{\partial u} = 0.$$

Damit erhält man das linearisierte Modell in der Form



$$\begin{pmatrix} \frac{d \delta x_1(t)}{dt} \\ \frac{d \delta x_2(t)}{dt} \\ \frac{d \delta x_3(t)}{dt} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ -\frac{g}{\bar{x}_1} & -\frac{\rho_0^2 g^2}{m}(\kappa_W - \kappa_D)\bar{x}_1 & -\frac{k_{yv}}{m} \end{pmatrix} \begin{pmatrix} \delta x_1(t) \\ \delta x_2(t) \\ \delta x_3(t) \end{pmatrix} + \begin{pmatrix} k \\ 0 \\ 0 \end{pmatrix} \delta u(t) \quad (\text{A.16})$$

$$\delta \mathbf{x}(0) = \mathbf{x}_0 - \bar{\mathbf{x}}. \quad (\text{A.17})$$

Die Ausgangsgleichung ist bereits linear:

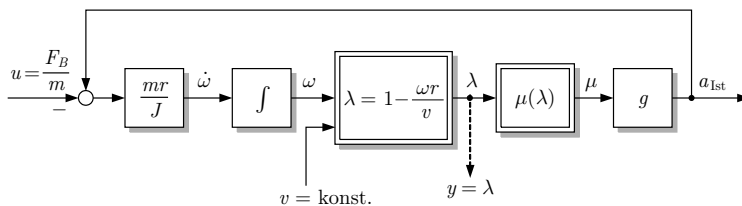
$$\delta y(t) = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \delta x_1(t) \\ \delta x_2(t) \\ \delta x_3(t) \end{pmatrix}. \quad (\text{A.18})$$

### Aufgabe 3.6 Linearisierte Beschreibung des Abbremsvorgangs eines Fahrzeugs

1. Durch die Annahme einer konstanten Geschwindigkeit  $v(t) = \bar{v}$  entfällt die in Abb. 3.16 eingetragene Rückführung von  $v$  auf den linken nichtlinearen Block. Das Blockschaltbild vereinfacht sich zu der in Abb. A.17 angegebenen Form. Es beschreibt den Zusammenhang zwischen der vom Fahrer gewünschten Sollbeschleunigung  $u(t) = \frac{F_B(t)}{m}$  und der Istbeschleunigung  $a_{\text{Ist}}(t)$ . Das Modell soll für die Bestimmung des Schlupfes verwendet werden, so dass jetzt  $y(t) = \lambda(t)$  gilt.

Auf Grund der Annahme, dass die Geschwindigkeit konstant ist, kann das Modell nur für Kurzzeituntersuchungen eingesetzt werden, also beispielsweise zur Beantwortung der Frage, wie schnell die Istbeschleunigung  $a_{\text{Ist}}$  der Sollbeschleunigung  $u(t)$  folgt bzw. wie sich der Schlupf  $\lambda(t)$  bei einer gegebenen Sollbeschleunigung verhält. In Aufgabe 6.4 wird das Modell eingesetzt, um die Stabilität des im Blockschaltbild gezeigten rückgeführten Systems zu untersuchen, wofür das linearisierte Modell ausreichend genau ist.

Das Modell eignet sich jedoch nicht, um den Verlauf der Beschleunigung für einen vollständigen Abbremsvorgang zu berechnen, weil dabei die Geschwindigkeitsänderung berücksichtigt werden muss.



**Abb. A.17:** Vereinfachtes Blockschaltbild, das den Zusammenhang der Sollbeschleunigung  $u(t)$ , der Istbeschleunigung  $a_{\text{Ist}}(t)$  und der Ausgangsgröße  $y(t)$  darstellt

2. Das Modell ist durch die Gln. (3.37) und (3.40) sowie die  $\mu(\lambda)$ -Kennlinie, die wie im Beispiel 3.9 mit  $f_2(\lambda)$  bezeichnet wird, beschrieben:

$$\lambda(t) = \frac{\bar{v} - r\omega(t)}{\bar{v}} = f_1(\bar{v}, \omega(t))$$

$$\dot{\omega}(t) = \frac{mr}{J} \left( g\mu(\lambda(t)) - \frac{F_B(t)}{m} \right).$$

Ineinander eingesetzt erhält man das nichtlineare Zustandsraummodell

$$\text{Fahrzeug : } \begin{cases} \dot{\omega}(t) = \frac{mr}{J} (f_2(f_1(\bar{v}, \omega(t))) g - u(t)), & \omega(0) = \omega_0 \\ y(t) = f_1(\bar{v}, \omega(t)), \end{cases} \quad (\text{A.19})$$

bei dem wie üblich die Eingangsgröße mit  $u(t)$  und die Ausgangsgröße mit  $y(t)$  bezeichnet sind. Wenn man sich nicht für den Schlupf, sondern für die Istbeschleunigung interessiert, kann man die Ausgabegleichung durch

$$a_{\text{Ist}}(t) = -f_2(f_1(\bar{v}, \omega(t))) g$$

ersetzen.

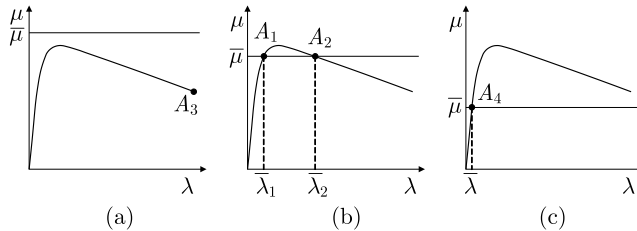
3. Entsprechend Gl. (3.29) muss für den durch  $\bar{u}$ ,  $\bar{\omega}$  und  $\bar{y}$  beschriebenen Arbeitspunkt die Beziehung

$$0 = \frac{mr}{J} (f_2(f_1(\bar{v}, \bar{\omega})) g - \bar{u})$$

gelten. Führt man die Abkürzung  $\bar{\mu} = f_2(f_1(\bar{v}, \bar{\omega}))$  ein, so muss im Arbeitspunkt die Beziehung

$$\bar{\mu} = \frac{\bar{u}}{g} \quad (\text{A.20})$$

erfüllt sein. Das heißt,  $\bar{\mu}$  ist die Kraftschlussbeanspruchung, die sich bei der Sollbeschleunigung  $\bar{u}$  einstellt.



**Abb. A.18:** Arbeitspunkte auf der  $\mu(\lambda)$ -Kennlinie

Wie Abb. A.18 zeigt, ist  $\bar{\mu}$  und folglich  $\bar{u}$  beschränkt. Ist nämlich  $\bar{\mu}$  größer als der Haftbeiwert  $\mu_H$  des Straßenbelags, so kann die Gl. (A.20) nicht erfüllt werden. Es stellt sich der im Abbildungsteil (a) eingetragene Arbeitspunkt  $A_3$  ein, bei dem das Rad blockiert. Die beiden rechten Teile der Abbildung zeigen Fälle, bei denen die Gl. (A.20) erfüllt ist, und zwar in zwei bzw. einem Arbeitspunkt. Wenn  $\bar{\mu} = \mu_H$  gilt, fallen die im Teil (b) gezeigten Arbeitspunkte  $A_1$  und  $A_2$  zu einem am Maximum der Kennlinie liegenden Arbeitspunkt zusammen. In allen Fällen beschreiben die in der Abbildung eingetragenen Größen

$$\bar{\lambda} = \frac{\bar{v} - r\bar{\omega}}{\bar{v}} = f_1(\bar{v}, \bar{\omega}) \quad (\text{A.21})$$

den im Arbeitspunkt wirkenden Schlupf  $\bar{\lambda}$ .

Um den zur Eingangsgröße  $\bar{u}$  gehörenden Arbeitspunkt auszurechnen, bestimmt man zunächst aus Gl. (A.20) die zu dieser Eingangsgröße gehörende Kraftschlussbeanspruchung  $\bar{\mu}$ , ermittelt dann aus der  $\mu(\lambda)$ -Kennlinie den dazu gehörenden Schlupf  $\bar{\lambda}$  und bestimmt schließlich aus der nach  $\bar{\omega}$  umgestellten Beziehung (A.21)

$$\bar{\omega} = \frac{(1 - \bar{\lambda})\bar{v}}{r} \quad (\text{A.22})$$

die Winkelgeschwindigkeit des Rades.

4. Entsprechend der Gln. (3.42) und (3.43) treten die Ableitungen der rechten Seite nach  $\omega$  und  $u$  in der linearisierten Zustandsgleichung (3.46) auf. Man erhält

$$\begin{aligned} \frac{d\delta\omega(t)}{dt} &= \frac{mrg}{J} \left( \frac{df_2(\lambda)}{d\lambda} \right)_{\lambda=\bar{\lambda}} \left( \frac{df_1(\bar{v}, \omega)}{d\omega} \right)_{\omega=\bar{\omega}} \delta\omega(t) - \frac{mr}{J} \delta u(t) \\ &= \frac{mrg}{J} k_{\mu\lambda} \left( -\frac{r}{\bar{v}} \right) \delta\omega(t) - \frac{mr}{J} \delta u(t) \end{aligned}$$

und

$$\frac{d\delta\omega(t)}{dt} = -\frac{mr^2g}{J\bar{v}} k_{\mu\lambda} \delta\omega(t) - \frac{mr}{J} \delta u(t), \quad \delta\omega(0) = \omega_0 - \bar{\omega}, \quad (\text{A.23})$$

wobei  $k_{\mu\lambda}$  die Steigung der  $\mu(\lambda)$ -Kennlinie im Arbeitspunkt beschreibt:

$$k_{\mu\lambda} = \left( \frac{df_2(\lambda)}{d\lambda} \right)_{\lambda=\bar{\lambda}}. \quad (\text{A.24})$$

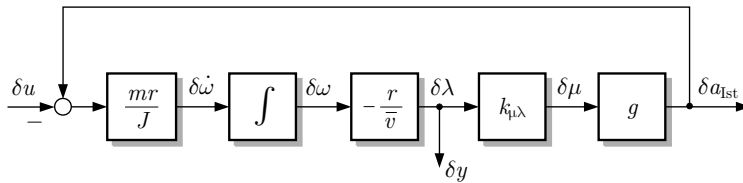
Zur Kennzeichnung der in diesen Gleichungen vorkommenden Abweichungen vom Arbeitspunkt wurden die Deltas beibehalten. Als linearisierte Ausgangsgleichung ergibt sich aus Gl. (A.19)

$$\delta y(t) = -\frac{r}{\bar{v}} \delta\omega(t) \quad (\text{A.25})$$

und für die Istbeschleunigung

$$\delta a_{\text{Ist}}(t) = \frac{rg}{\bar{v}} k_{\mu\lambda} \delta\omega(t).$$

Das linearisierte Zustandsraummodell besteht aus der Zustandsgleichung (A.23) und der Ausgangsgleichung (A.25).



**Abb. A.19:** Blockschaltbild des linearisierten Modells

5. Abbildung A.19 zeigt das Blockschaltbild des linearisierten Modells. Es ist offensichtlich, dass bei der Linearisierung die nichtlinearen durch lineare Blöcke ersetzt wurden. Dabei ist zu beachten, dass die Übertragungseigenschaften der linearisierten Blöcke vom Arbeitspunkt abhängen.

**Aufgabe 3.7** Modell eines Wärmeübertragers

1. Für den Arbeitspunkt des affinen Systems (3.47) erhält man aus  $\dot{\mathbf{x}} = \mathbf{0}$  die Beziehungen

$$\begin{aligned}\bar{\mathbf{x}} &= -\tilde{\mathbf{A}}^{-1} \mathbf{e} \\ \bar{y} &= -\tilde{\mathbf{c}}^T \tilde{\mathbf{A}}^{-1} \mathbf{e},\end{aligned}$$

so dass im linearisierten Modell die Größen

$$\begin{aligned}\delta \mathbf{x}(t) &= \mathbf{x}(t) + \tilde{\mathbf{A}}^{-1} \mathbf{e} \\ \delta y(t) &= y(t) + \tilde{\mathbf{c}}^T \tilde{\mathbf{A}}^{-1} \mathbf{e}\end{aligned}$$

stehen. Entsprechend Gl. (3.42) erhält man mit  $\mathbf{g}(\mathbf{x}(t)) = \tilde{\mathbf{A}}\mathbf{x}(t) + \mathbf{e}$  die Beziehung

$$\mathbf{A} = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \tilde{\mathbf{A}}$$

und aus Gl. (3.44) mit  $h(\mathbf{x}(t)) = \tilde{\mathbf{c}}^T \mathbf{x}(t)$

$$\mathbf{c}^T = \frac{\partial h}{\partial \mathbf{x}} = \tilde{\mathbf{c}}^T.$$

Das heißt, im linearisierten Modell

$$\Sigma : \begin{cases} \frac{d\delta \mathbf{x}(t)}{dt} = \tilde{\mathbf{A}} \delta \mathbf{x}(t) \\ \delta y(t) = \tilde{\mathbf{c}}^T \delta \mathbf{x}(t) \end{cases}$$

stehen dieselbe Matrix  $\tilde{\mathbf{A}}$  und derselbe Vektor  $\tilde{\mathbf{c}}$  wie im affinen System. Allerdings kommt der Summand  $\mathbf{e}$  nicht mehr vor, weshalb das Modell jetzt linear ist. Bei affinen Systemen bedeutet die Linearisierung nur eine Verschiebung des Arbeitspunkts, wodurch der affine Anteil  $+\mathbf{e}$  aus dem Modell verschwindet.

Die Rechnung zeigt, dass affine Systeme nichtlinear sind, weil eine Linearisierung eine Veränderung des Modells mit sich bringt. Deshalb tritt in der allgemeinen Form (3.17) des Zustandsraummodells kein Summand  $+\mathbf{e}$  auf.

2. Für den Wärmeübertrager erhält man ein affines Modell mit dem Zustandsvektor

$$\mathbf{x}(t) = \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix}$$

folgendermaßen:

$$\text{Wärmeübertrager : } \begin{cases} \begin{pmatrix} \dot{T}_1(t) \\ \dot{T}_2(t) \end{pmatrix} = \begin{pmatrix} -\frac{kA+w_1c_1}{m_1c_1} & +\frac{kA}{m_1c_1} \\ +\frac{kA}{m_2c_2} & -\frac{kA+w_2c_2}{m_2c_2} \end{pmatrix} \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix} + \begin{pmatrix} \frac{w_1}{m_1} T_{10} \\ \frac{w_2}{m_2} T_{20} \end{pmatrix} \\ y(t) = (1 \quad 0) \begin{pmatrix} T_1(t) \\ T_2(t) \end{pmatrix}. \end{cases}$$

Der Anfangszustand ist durch die Temperaturen  $T_{10}$  und  $T_{20}$  zur Zeit  $t = 0$  festgelegt:

$$\mathbf{x}(0) = \begin{pmatrix} T_{10} \\ T_{20} \end{pmatrix}.$$

## 3. Der Arbeitspunkt

$$\begin{pmatrix} \bar{T}_1 \\ \bar{T}_2 \end{pmatrix} = - \begin{pmatrix} -\frac{kA+w_1c_1}{m_1c_1} & +\frac{kA}{m_1c_1} \\ +\frac{kA}{m_2c_2} & -\frac{kA+w_2c_2}{m_2c_2} \end{pmatrix}^{-1} \cdot \begin{pmatrix} \frac{w_1}{m_1} T_{10} \\ \frac{w_2}{m_2} T_{20} \end{pmatrix}$$

gibt die Temperaturen an, die sich stationär im Wärmeübertrager einstellen. Das linearisierte Modell betrifft die Größen

$$\delta T_1(t) = T_1(t) - \bar{T}_1$$

$$\delta T_2(t) = T_2(t) - \bar{T}_2$$

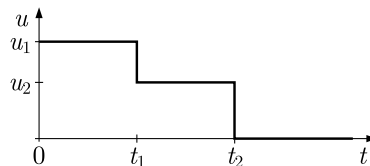
und lautet

$$\text{Wärmeübertrager : } \begin{cases} \frac{d}{dt} \begin{pmatrix} \delta T_1(t) \\ \delta T_2(t) \end{pmatrix} = \begin{pmatrix} -\frac{kA+w_1c_1}{m_1c_1} & +\frac{kA}{m_1c_1} \\ +\frac{kA}{m_2c_2} & -\frac{kA+w_2c_2}{m_2c_2} \end{pmatrix} \begin{pmatrix} \delta T_1(t) \\ \delta T_2(t) \end{pmatrix} \\ \begin{pmatrix} \delta T_1(t) \\ \delta T_2(t) \end{pmatrix} = \begin{pmatrix} T_{10} - \bar{T}_1 \\ T_{20} - \bar{T}_2 \end{pmatrix} \\ \delta y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \delta T_1(t) \\ \delta T_2(t) \end{pmatrix}. \end{cases}$$

**Aufgabe 4.2** Temperaturverhalten eines Rührkesselreaktors
1. Der zeitliche Verlauf der Eingangsgröße  $u(t)$ 

$$u(t) = \begin{cases} u_1 & \text{für } 0 \leq t < t_1 \\ u_2 & \text{für } t_1 \leq t < t_2 \\ 0 & \text{für } t_2 \leq t \end{cases}$$

ist in Abb. A.20 dargestellt. Der Temperaturverlauf  $T(t)$  muss für die drei Zeitintervalle getrennt berechnet werden. Da der Anfangswert  $T(0)$  verschwindet, tritt in der Bewegungsgleichung nur der Summand für die erzwungene Bewegung auf.

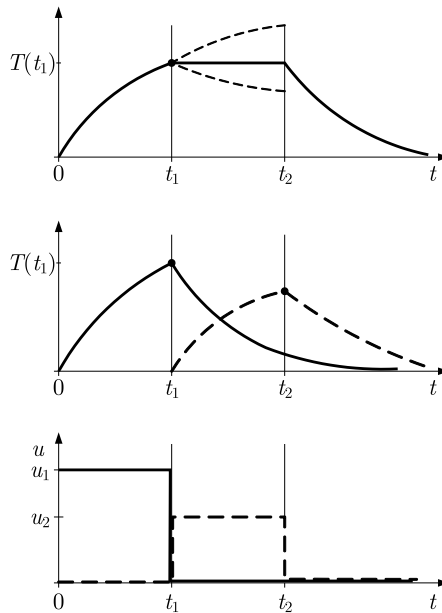


**Abb. A.20:** Verlauf der Eingangsgröße

Für  $0 \leq t < t_1$  erhält man für das Modell (4.23) aus der Bewegungsgleichung (4.5) die Beziehung

$$\begin{aligned}
 T(t) &= \int_0^t e^{a(t-\tau)} b u_1 \, d\tau \\
 &= b u_1 e^{at} \int_0^t e^{-a\tau} \, d\tau \\
 &= -\frac{b u_1}{a} e^{at} e^{-a\tau} \Big|_0^t \\
 &= -\frac{b u_1}{a} (1 - e^{at}) \quad \text{für } 0 \leq t < t_1.
 \end{aligned}$$

Weil  $a$  negativ ist, klingt die e-Funktion ab, so dass sich die Temperatur asymptotisch dem Wert  $-\frac{b u_1}{a}$  nähert. Der Temperaturverlauf entspricht der Übergangsfunktion eines Systems erster Ordnung (vgl. oberen Teil von Abb. A.21 für das Zeitintervall  $0 \leq t < t_1$ ).



**Abb. A.21:** Temperaturverlauf für den Rührkesselreaktor (oben); Zerlegung des Temperaturverlaufs (Mitte); Eingangsgrößen (unten)

Für das Zeitintervall  $t_1 \leq t < t_2$  muss das Integral in der Bewegungsgleichung in zwei Anteile zerlegt werden, um die für die beiden Zeitintervalle  $0 \leq t < t_1$  und  $t_1 \leq t$  geltenden konstanten Eingangsgrößen einsetzen zu können:

$$\begin{aligned}
 T(t) &= \int_0^t e^{a(t-\tau)} b u(\tau) \, d\tau \\
 &= \int_0^{t_1} e^{a(t-\tau)} b u_1 \, d\tau + \int_{t_1}^t e^{a(t-\tau)} b u_2 \, d\tau
 \end{aligned}$$

$$\begin{aligned}
&= -\frac{bu_1}{a} e^{at} e^{-a\tau} \Big|_0^{t_1} - \frac{bu_2}{a} e^{at} e^{-a\tau} \Big|_{t_1}^t \\
&= -\frac{bu_1}{a} (1 - e^{at_1}) e^{a(t-t_1)} - \frac{bu_2}{a} (1 - e^{a(t-t_1)}) \quad \text{für } t_1 \leq t < t_2. \quad (\text{A.26})
\end{aligned}$$

Für das dritte Zeitintervall  $t_2 \leq t$  muss man das Integral zunächst in drei Summanden zerlegen. Der dritte Summand verschwindet jedoch, weil die Eingangsgröße für das in diesem Integral stehende Zeitintervall gleich null ist:

$$\begin{aligned}
T(t) &= \int_0^t e^{a(t-\tau)} bu(\tau) d\tau \\
&= \int_0^{t_1} e^{a(t-\tau)} bu_1 d\tau + \int_{t_1}^{t_2} e^{a(t-\tau)} bu_2 d\tau + \int_{t_2}^t e^{a(t-\tau)} b \cdot 0 d\tau \\
&= -\frac{bu_1}{a} e^{at} e^{-a\tau} \Big|_0^{t_1} - \frac{bu_2}{a} e^{at} e^{-a\tau} \Big|_{t_1}^{t_2} \\
&= -\frac{bu_1}{a} (1 - e^{at_1}) e^{a(t-t_1)} - \frac{bu_2}{a} (1 - e^{a(t_2-t_1)}) e^{a(t-t_2)} \quad \text{für } t \geq t_2. \quad (\text{A.27})
\end{aligned}$$

2. Damit die Temperatur im Zeitintervall  $t_1 \leq t < t_2$  konstant bleibt, muss die aus Gl. (A.26) berechenbare zeitliche Ableitung der Temperatur verschwinden

$$\frac{dT(t)}{dt} = -a \frac{bu_1}{a} (1 - e^{at_1}) e^{a(t-t_1)} + a \frac{bu_2}{a} e^{a(t-t_1)} = 0,$$

was auf

$$u_2 = u_1 (1 - e^{at_1})$$

führt. Für diese Eingangsgröße wird die Verkleinerung des ersten Summanden von Gl. (A.26) durch die Vergrößerung des zweiten Summanden gerade ausgeglichen und die Temperatur hat den im oberen Teil der Abb. A.21 durch die dicke Linie gezeigten Verlauf. Ist  $u_2$  größer, so steigt die Temperatur in diesem Zeitintervall an, ist  $u_2$  kleiner, so fällt die Temperatur, wie es durch die gestrichelten Linien angedeutet ist.

3. Aus Gl. (A.26) erhält man für die Temperatur zur Zeit  $t_1$  die Beziehung

$$T_1 = T(t_1) = -\frac{bu_1}{a} (1 - e^{at_1}).$$

Der jeweils erste Summand in den Gln. (A.26) und (A.27) beschreibt also die Eigenbewegung, die zur Zeit  $t_1$  in der Temperatur  $T_1$  beginnt:

$$-\frac{bu_1}{a} (1 - e^{at_1}) e^{a(t-t_1)} = T_1 e^{a(t-t_1)}.$$

Zerlegt man die Eingangsgröße  $u(t)$  in die zwei Anteile, die im unteren Teil der Abb. A.21 mit der durchgezogenen bzw. der gestrichelten Linie dargestellt sind, so entspricht diese Bewegung dem durch die durchgezogene Eingangsgröße hervorgerufenen Anteil an der Bewegung im Zeitintervall  $t \geq t_1$ . Dieser ist im mittleren Teil der Abbildung mit einer durchgezogenen Linie gezeichnet.

Der zweite Summand in Gl. (A.27) kann ebenfalls als eine Eigenbewegung gedeutet werden und zwar als eine solche Bewegung im Zeitintervall  $t \geq t_2$ , die zum Zeitpunkt  $t_2$  bei der Temperatur

$$T_2 = -\frac{bu_2}{a} (1 - e^{a(t_2-t_1)})$$

beginnt. Diese Bewegung wird durch den gestrichelten Anteil der im Zeitintervall  $t_1 \leq t < t_2$  nicht verschwindenden Eingangsgröße hervorgerufen und durch die gestrichelte Linie im mittleren Abbildungsteil dargestellt.

Zusammengefasst zeigt diese Analyse, dass die Bewegung für das Zeitintervall  $t > t_2$ , in dem die Eingangsgröße verschwindet, als Überlagerung zweier Eigenbewegungen gedeutet werden kann. Diese beiden Eigenbewegungen beginnen bei den Temperaturen  $T_1$  bzw.  $T_2$  und klingen nach einer e-Funktion ab.  $T_1$  und  $T_2$  beschreiben den Temperaturanstieg in der Flüssigkeit, den die beiden Anteile der Eingangsgröße bewirkt haben.

#### Aufgabe 4.4 Kennwertermittlung eines Wärmeübertragers

1. Um auf die Übergangsfunktion zu kommen, müssen die Experimentdaten in drei Schritten verändert werden:

1. Es wird eine neue Zeitachse eingeführt, deren Zeitpunkt  $t = 0$  gleich dem Zeitpunkt ist, an dem die Eingangsgröße sprunghaft verändert wurde, hier also der Zeitpunkt 255 min.
2. Das Modell soll das Verhalten um den Arbeitspunkt beschreiben, der bis zur Veränderung der Eingangsgröße bei  $57^\circ\text{C}$  lag. Also werden alle Temperaturen um  $57^\circ\text{C}$  verkleinert, so dass die neue Temperaturachse bei  $57^\circ\text{C}$  beginnt. Die ersten beiden Veränderungen führen auf das im oberen Teil der Abb. A.22 eingetragene Koordinatensystem.
3. Die Ausgangsgröße muss auf die Sprunghöhe der Eingangsgröße normiert, hier also durch 10 dividiert werden.

Außerdem muss das Messrauschen durch ein Filter (oder näherungsweise durch Konstruktion der Kurve nach „Augenmaß“) beseitigt werden. Das Ergebnis ist im unteren Teil der Abb. A.22 zu sehen.

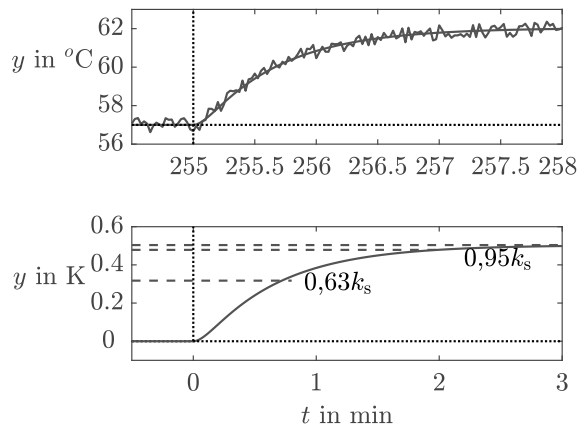
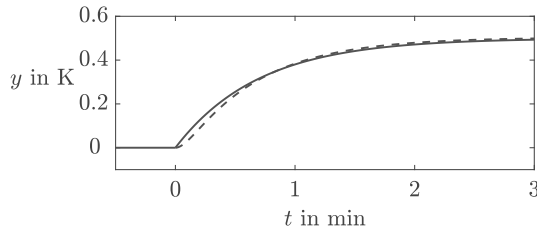


Abb. A.22: Analyse der Experimentdaten

2. Aus der unteren Kurve in Abb. A.22 liest man  $k_s = 0,5$  ab. Das heißt, dass eine Temperaturerhöhung oder -erniedrigung der Kühlflüssigkeit um ein Kelvin auf eine Temperaturänderung in der gekühlten Flüssigkeit von  $0,5$  Kelvin führt.



Für die Zeitkonstante liest man mit der 63%-Regel den Wert 0,7 min, mit der 95%-Regel den Wert  $\frac{19}{3} \approx 0,63$  min ab. Die „gefilterte“ Kurve hat zum Zeitpunkt  $t = 0$  einen sehr flachen Anstieg, so dass sich dort sehr schlecht eine Tangente an die Kurve zeichnen lässt. Die Tangentenmethode führt auf  $T \approx 1$ . Diese Ergebnisse besagen, dass sich eine Temperaturänderung der Kühlflüssigkeit nach  $3 \cdot T \approx 1,89 \dots 3$  min vollständig in einer Temperaturänderung der gekühlten Flüssigkeit bemerkbar macht.



**Abb. A.23:** Vergleich von Modell (—) und Experiment (---)

Abbildung A.23 vergleicht die Übergangsfunktionen des Modells mit  $k_s = 0,5$  und  $T = 0,7$  min mit der gefilterten Messkurve. Es ist eine sehr gute Übereinstimmung festzustellen.

3. Das Modell (17.2) ist ein Modell zweiter Ordnung, während hier ein Modell erster Ordnung bestimmt wurde. Die dynamische Ordnung 2 bei der theoretischen Modellbildung ergibt sich aus den zwei Bilanzräumen, in denen nach dem Energieerhaltungssatz die zugeführte Wärme gleich der Summe der abgeführten und der gespeicherten Wärme sein muss. Diese beiden Bilanzräume sind die Rohrleitung mit der gekühlten Flüssigkeit und der Teil des Wärmeübertragers, durch den das Kühlmittel fließt.
4. Der Vergleich von Modell und Experiment in Abb. A.23 zeigt, dass das Verhalten des Wärmeübertragers trotzdem sehr gut durch ein Modell erster Ordnung dargestellt werden kann. Der Grund dafür liegt in der Tatsache, dass der Wärmeübertrager zwei Zeitkonstanten mit sehr unterschiedlicher Größe hat, die für die hier verwendeten Parameter bei  $T_1 = 0,63$  min und  $T_2 = 0,08$  min liegen. Die kleine Zeitkonstante beeinflusst die Übergangsfunktion nur in einem kurzen Zeitabschnitt, der etwa zwischen 0 und  $3 \cdot T_2 \approx 0,25$  Minuten liegt. Der wesentliche Teil des Übergangsverhaltens wird durch die Zeitkonstante  $T_1$  beeinflusst, die etwa mit der ermittelten Zeitkonstante übereinstimmt. Diese Zeitkonstante entspricht etwa der Zeit, nach der die sich im Wärmeübertrager befindenden Flüssigkeiten einmal ausgetauscht sind.

### Aufgabe 5.1 Berechnung des aktuellen Zustands aus $n$ Messwerten

Berechnet werden soll der Zustand

$$\mathbf{x}_e = \underbrace{e^{\mathbf{A}t_e} \mathbf{x}_0}_{\mathbf{x}_{\text{frei}}(t_e)} + \underbrace{\int_0^{t_e} e^{\mathbf{A}(t-\tau)} \mathbf{b}u(\tau) d\tau}_{\mathbf{x}_{\text{erzw}}(t_e)}.$$

Der zweite Summand kann aus dem bekannten Verlauf der Eingangsgröße berechnet werden. Für den ersten Summanden wird Gl. (5.18) verwendet, womit man auf die Beziehung

$$\mathbf{x}_{\text{frei}}(t_e) = \underbrace{e^{\mathbf{A}t_e} \mathbf{M}^{-1}}_{\tilde{\mathbf{M}}^{-1}} \begin{pmatrix} y_{\text{frei}}(t_1) \\ y_{\text{frei}}(t_2) \\ \vdots \\ y_{\text{frei}}(t_n) \end{pmatrix}$$

kommt und für die Matrix  $\tilde{\mathbf{M}}$  der folgende Ausdruck entsteht:

$$\begin{aligned} \tilde{\mathbf{M}} &= \mathbf{M} \left( e^{\mathbf{A}t_e} \right)^{-1} \\ &= \begin{pmatrix} \mathbf{c}^T e^{\mathbf{A}t_1} \\ \mathbf{c}^T e^{\mathbf{A}t_2} \\ \vdots \\ \mathbf{c}^T e^{\mathbf{A}t_n} \end{pmatrix} e^{-\mathbf{A}t_e} \\ &= \begin{pmatrix} \mathbf{c}^T e^{\mathbf{A}(t_1 - t_e)} \\ \mathbf{c}^T e^{\mathbf{A}(t_2 - t_e)} \\ \vdots \\ \mathbf{c}^T e^{\mathbf{A}(t_n - t_e)} \end{pmatrix}. \end{aligned}$$

Nach dem Einsetzen dieser Größen erhält man den gesuchten Zustand  $\mathbf{x}_e$ .

### Aufgabe 5.3 Vollständige Steuerbarkeit und strukturelle Steuerbarkeit

Da die angegebenen Parameter von null verschieden sein können, ist die Klasse  $\mathcal{S}(\mathbf{S}_A, \mathbf{S}_b)$  von Systemen durch die Strukturmatrizen

$$\mathbf{S}_A = \begin{pmatrix} * & * \\ 0 & 0 \end{pmatrix} \quad \text{und} \quad \mathbf{S}_b = \begin{pmatrix} 0 \\ * \end{pmatrix}$$

festgelegt. Entsprechend Definition 5.3 ist diese Systemklasse strukturell steuerbar, wenn es ein System

$$\tilde{\Sigma} = \left( \begin{pmatrix} a_{11} & a_{12} \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 \\ b_2 \end{pmatrix} \right) \in \mathcal{S}$$

gibt, das vollständig steuerbar ist. Das System  $\tilde{\Sigma}$  muss also das Steuerbarkeitskriterium (5.4) erfüllen:

$$\text{Rang} \begin{pmatrix} 0 & a_{12}b_2 \\ b_2 & 0 \end{pmatrix} = 2.$$

Dieses Kriterium ist erfüllt, wenn

$$\begin{aligned} \det \begin{pmatrix} 0 & a_{12}b_2 \\ b_2 & 0 \end{pmatrix} &\neq 0 \\ a_{12}b_2^2 &\neq 0 \end{aligned}$$

gilt, d. h., wenn die Parameter  $a_{21}$  und  $b_2$  von null verschieden sind. Diese Bedingung hat zwei Konsequenzen. Erstens ist damit gezeigt, dass die hier betrachtete Systemklasse  $\mathcal{S}$  strukturell beobachtbar ist,

denn die angegebene Bedingung ist für viele Systeme aus der Klasse  $\mathcal{S}$  erfüllt. Zweitens zeigt diese Beziehung, dass die Systeme  $\Sigma$ , die nicht vollständig steuerbar sind, obwohl sie zu einer strukturell steuerbaren Systemklasse gehören, auf den Ebenen im Raum der drei Parameter  $a_{11}$ ,  $a_{12}$  und  $b_2$  liegen, die durch die Beziehungen

$$a_{12} = 0 \quad \text{oder} \quad b_2 = 0$$

beschrieben sind. Dies sind die Ebenen, die durch die  $a_{11}$ -Achse zusammen mit der  $b_2$ -Achse bzw. durch die  $a_{11}$ -Achse und der  $a_{21}$ -Achse aufgespannt werden. Die strukturelle Eigenschaft der Steuerbarkeit, die für die Klasse  $\mathcal{S}$  gilt, schlägt sich also in der vollständigen Steuerbarkeit aller Systeme mit Ausnahme derjenigen Systeme  $\Sigma$  nieder, deren Parameter auf diesen Ebenen liegen. „Fast alle“ Systeme  $\Sigma$  sind deshalb vollständig steuerbar.

Zu demselben Ergebnis kommt man mit den beiden notwendigen und hinreichenden Bedingungen für die strukturelle Steuerbarkeit aus Satz 5.3, denn der Strukturgraph enthält einen Pfad  $u \rightarrow x_2 \rightarrow x_1$  und ist damit eingangsverbunden und die Rangbedingung ist erfüllt:

$$\begin{pmatrix} * & \bullet & 0 \\ 0 & 0 & \bullet \end{pmatrix} = 2.$$

#### Aufgabe 5.4 Steuerbarkeit und Beobachtbarkeit einer RC-Schaltung

1. Das Zustandsraummodell erhält man aus den Bauelementgleichungen

$$i_1(t) = C_1 \dot{u}_{C_1}(t)$$

$$i_2(t) = C_2 \dot{u}_{C_2}(t)$$

$$u_{R_1}(t) = R_1 i_1(t)$$

$$u_{R_2}(t) = R_2 i_2(t)$$

und den Kopplungsbedingungen zwischen den Bauelementen, die aus dem Maschensatz folgen:

$$u(t) = u_{R_1}(t) + u_{C_1}(t)$$

$$u(t) = u_{R_2}(t) + u_{C_2}(t)$$

Durch Zusammenfassen der Gleichungen erhält man

$$\dot{u}_{C_1}(t) = -\frac{1}{R_1 C_1} u_{C_1}(t) + \frac{1}{R_1 C_1} u(t)$$

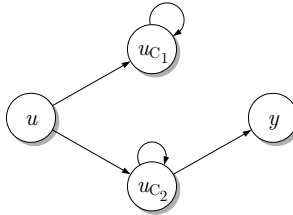
$$\dot{u}_{C_2}(t) = -\frac{1}{R_2 C_2} u_{C_2}(t) + \frac{1}{R_2 C_2} u(t)$$

und daraus nach Einführung der Zeitkonstanten  $T_1 = R_1 C_1$  und  $T_2 = R_2 C_2$  das Zustandsraummodell

$$\text{RC-Schaltung : } \begin{cases} \begin{pmatrix} \dot{u}_{C_1}(t) \\ \dot{u}_{C_2}(t) \end{pmatrix} = \begin{pmatrix} -\frac{1}{T_1} & 0 \\ 0 & -\frac{1}{T_2} \end{pmatrix} \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{T_1} \\ \frac{1}{T_2} \end{pmatrix} u(t) \\ \begin{pmatrix} u_{C_1}(0) \\ u_{C_2}(0) \end{pmatrix} = \begin{pmatrix} u_{C_{10}} \\ u_{C_{20}} \end{pmatrix} \\ y(t) = (0 \ 1) \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix}, \end{cases}$$

in dem die beiden Kondensatorspannungen als Zustandsgrößen auftreten.

2. Da in der Systemmatrix nur die Diagonalelemente besetzt sind, zerfällt das System in die Parallelschaltung zweier Teilsysteme erster Ordnung, von denen nur die Zustandsvariable des zweiten Teilsystems gemessen wird. Weil die beiden Zustandsvariablen nicht miteinander gekoppelt sind, kann man vermuten, dass das System nicht vollständig beobachtbar ist. Allerdings wird es steuerbar sein, weil die Eingangsgröße beide Zustandsvariablen beeinflusst.



**Abb. A.24:** Strukturgraph der RC-Schaltung

3. Abbildung A.24 zeigt den Strukturgraphen der RC-Schaltung. Das System ist nicht ausgangsverbunden, also auch nicht strukturell beobachtbar. Damit ist es auch nicht vollständig beobachtbar, wie aus der Zusammensetzung des Systems als Parallelschaltung schon vermutet wurde.

Das System ist eingangsverbunden und erfüllt die für die strukturelle Steuerbarkeit notwendige Rangbedingung

$$s - \text{Rang}([A] \ [b]) = s - \text{Rang} \begin{pmatrix} \bullet & 0 & * \\ 0 & \bullet & * \end{pmatrix} = 2,$$

wie die zwei hervorgehobenen Matrixelemente zeigen. Um die vollständige Steuerbarkeit zu überprüfen, wird die Steuerbarkeitsmatrix aufgestellt:

$$S_S = (b \quad Ab) = \begin{pmatrix} \frac{1}{T_1} & -\frac{1}{T_1^2} \\ \frac{1}{T_2} & -\frac{1}{T_2^2} \end{pmatrix}.$$

Es gilt

$$\det S_S = \frac{1}{T_1 T_2} \left( \frac{1}{T_1} - \frac{1}{T_2} \right).$$

Die Steuerbarkeitsbedingung ist also genau dann erfüllt, wenn

$$T_1 \neq T_2$$

ist. Das heißt, das strukturell steuerbare System ist für alle diejenigen Parameterwerte auch vollständig steuerbar, für die die beiden Zeitkonstanten verschiedene Werte haben. Nur für Parameter, für die  $T_1 = T_2 = T$  gilt, ist das System zwar strukturell steuerbar, aber nicht vollständig steuerbar. Dann gilt

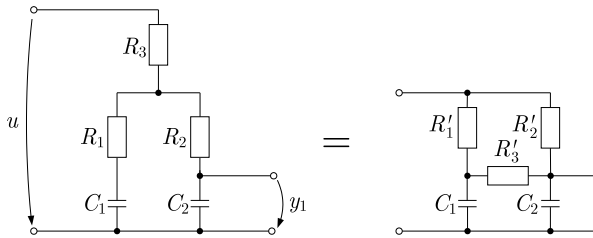
$$\begin{aligned} u_{C1}(t) &= e^{-\frac{t}{T}} u_{C10} + \frac{1}{T} \int_0^t e^{-\frac{t-\tau}{T}} u(\tau) d\tau \\ u_{C2}(t) &= e^{-\frac{t}{T}} u_{C20} + \frac{1}{T} \int_0^t e^{-\frac{t-\tau}{T}} u(\tau) d\tau \end{aligned}$$

woraus

$$u_{C_1}(t) = u_{C_2}(t) + (u_{C_{10}} - u_{C_{20}}) e^{-\frac{t}{T}}$$

folgt. Das heißt, unabhängig von der Wahl der Eingangsspannung  $u(t)$  unterscheiden sich die beiden Kondensatorspannungen immer um eine von  $u_{C_{10}} - u_{C_{20}}$  abhängige Differenz. Die Kondensatorspannungen können also nicht auf beliebig vorgegebene Werte gebracht werden.

Die Tatsache, dass die Schaltung für keine Parameterwerte vollständig beobachtbar ist und für  $T_1 = T_2$  auch nicht vollständig steuerbar ist, hängt übrigens nicht von den im Zustandsraummodell verwendeten Zustandsgrößen ab. Wenn man beispielsweise an Stelle der Kondensatorspannungen die Ströme in den beiden Zweigen als Zustandsvariable verwenden würde, erhält man dieselben Aussagen.



**Abb. A.25:** Veränderte RC-Schaltung

4. Die Schaltung ist nicht beobachtbar, weil die beiden parallel geschalteten RC-Elemente durch die Vorgabe der Spannung  $u$  vollkommen unabhängig voneinander aufgeladen und entladen werden können, so dass die Kenntnis der Kondensatorspannung  $u_{C_2}$  keine Informationen über die Kondensatorspannung  $u_{C_1}$  enthält.

Wenn man die Schaltung so verändern will, dass sie vollständig beobachtbar wird, muss man die beiden Kondensatorspannungen miteinander koppeln. Abbildung A.25 zeigt zwei Möglichkeiten, die durch eine Stern-Dreieck-Umformung ineinander überführt werden können. Im Folgenden wird deshalb nur die rechte Schaltung untersucht.

Die Bauelemente sind mit den üblichen Gleichungen  $i_R(t) = \frac{u_R(t)}{R'}$  und  $\dot{u}_C(t) = \frac{i_C(t)}{C}$  mit den entsprechenden Indizes beschrieben. Die drei Maschen der Schaltung ergeben die folgenden Kopplungen der für die Bauelementebeschreibung verwendeten Signale:

$$\begin{aligned} u(t) &= u_{R_1}(t) + u_{C_1}(t) \\ u(t) &= u_{R_2}(t) + u_{C_2}(t) \\ u_{R_2}(t) &= u_{R_1}(t) + u_{R_3}(t) \\ i(t) &= i_{R_1}(t) + i_{R_2}(t) \\ i_{R_1}(t) &= i_{C_1}(t) + i_{R_3}(t) \\ i_{C_2}(t) &= i_{R_2}(t) + i_{R_3}(t). \end{aligned}$$

Daraus erhält man

$$\begin{aligned} \dot{u}_{C_1}(t) &= \frac{i_{C_1}(t)}{C_1} \\ &= \frac{i_{R_1}(t) - i_{R_3}(t)}{C_1} \end{aligned}$$

$$\begin{aligned}
&= \frac{u_{R_1}(t)}{R'_1 C_1} - \frac{u_{R_3}(t)}{R'_3 C_1} \\
&= \frac{u(t) - u_{C_1}(t)}{R'_1 C_1} - \frac{u_{R_2}(t) - u_{R_1}(t)}{R'_3 C_1} \\
&= \frac{u(t) - u_{C_1}(t)}{R'_1 C_1} - \frac{u(t) - u_{C_2}(t)}{R'_3 C_1} + \frac{u(t) - u_{C_1}(t)}{R'_3 C_1} \\
&= \left( -\frac{1}{R'_1 C_1} - \frac{1}{R'_3 C_1} \right) u_{C_1}(t) + \frac{1}{R'_3 C_1} u_{C_2}(t) + \frac{1}{R'_1 C_1} u(t)
\end{aligned}$$

und auf demselben Weg

$$\dot{u}_{C_2}(t) = \frac{1}{R'_3 C_1} u_{C_1}(t) + \left( -\frac{1}{R'_2 C_2} - \frac{1}{R'_3 C_2} \right) u_{C_2}(t) + \frac{1}{R'_2 C_2} u(t).$$

Als Zustandsraummodell erhält man daraus

$$\text{rechte Schaltung : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{u}_{C_1}(t) \\ \dot{u}_{C_2}(t) \end{pmatrix} = \begin{pmatrix} -\left(\frac{1}{R'_1 C_1} + \frac{1}{R'_3 C_1}\right) & \frac{1}{R'_3 C_1} \\ \frac{1}{R'_3 C_2} & -\left(\frac{1}{R'_2 C_2} + \frac{1}{R'_3 C_2}\right) \end{pmatrix} \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} \\ \quad + \begin{pmatrix} \frac{1}{R'_1 C_1} \\ \frac{1}{R'_2 C_2} \end{pmatrix} u(t) \\ y(t) = (0 \ 1) \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} \end{array} \right.$$

Für die linke Schaltung bekommt man folgendes Zustandsraummodell

$$\text{linke Schaltung : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{u}_{C_1}(t) \\ \dot{u}_{C_2}(t) \end{pmatrix} = \begin{pmatrix} -\frac{R_2+R_3}{R^2 C_1} & \frac{R_3}{R^2 C_1} \\ \frac{R_3}{R^2 C_2} & -\frac{R_1+R_3}{R^2 C_2} \end{pmatrix} \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} + \begin{pmatrix} \frac{R_2}{R^2 C_1} \\ \frac{R_1}{R^2 C_2} \end{pmatrix} u(t) \\ y(t) = (0 \ 1) \begin{pmatrix} u_{C_1}(t) \\ u_{C_2}(t) \end{pmatrix} \end{array} \right.$$

in dem die Abkürzung

$$\bar{R}^2 = R_1 R_2 + R_1 R_3 + R_2 R_3$$

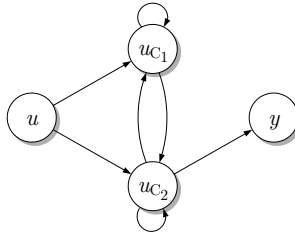
verwendet wird.

In beiden Modellen sind die in der Nebendiagonalen der Systemmatrix stehenden Elemente genau dann verschieden von null, wenn der Widerstand  $R_3$  bzw.  $R'_3$  nicht verschwindet. Beide RC-Glieder sind unter dieser Bedingung also miteinander verkoppelt, was vermuten lässt, dass die Schaltung jetzt vollständig beobachtbar ist, weil man durch Messung der Kondensatorspannung  $u_{C_2}(t)$  auch Informationen über die Kondensatorspannung  $u_{C_1}(t)$  erhält.

Abbildung A.26 zeigt den Strukturgraphen der veränderten RC-Schaltung. Die Systemklasse, zu der die veränderte RC-Schaltung gehört, ist sowohl eingangsverbunden als auch ausgangsverbunden und erfüllt auch die beiden Rangbedingungen an die Strukturmatrizen. Sie ist also strukturell steuerbar und strukturell beobachtbar.

Für die Beobachtbarkeitsmatrix erhält man

$$S_B = \begin{pmatrix} \mathbf{c}^T \\ \mathbf{c}^T \mathbf{A} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{R_3}{R^2 C_2} & -\frac{R_1+R_3}{R^2 C_2} \end{pmatrix},$$



**Abb. A.26:** Strukturgraph der veränderten RC-Schaltung

so dass die Rangbedingung genau dann erfüllt ist, wenn  $R_3 \neq 0$  gilt. Wenn  $R_3$  vorhanden ist, so ist die Schaltung also nicht nur strukturell beobachtbar, sondern auch vollständig beobachtbar. Dieses Ergebnis gilt übrigens für beliebige Zeitkonstanten  $T_1$  und  $T_2$ .

Wie man durch eine Analyse der Steuerbarkeitsmatrix erkennen kann, ist auch das veränderte System genau dann vollständig steuerbar, wenn  $T_1 \neq T_2$  gilt. Der Widerstand  $R_3$  ändert also nichts an der Steuerbarkeit.

### Aufgabe 5.5 Konstruktion eines strukturell nicht steuerbaren Systems

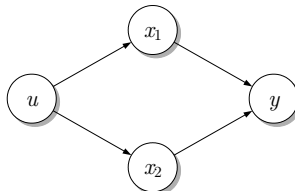
Abbildung A.27 zeigt den Strukturgraphen des Systems

$$\Sigma : \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} u(t), & \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} x_{10} \\ x_{20} \end{pmatrix} \\ y(t) = (c_1 \ c_2) \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}. \end{cases}$$

Das System ist eingangsverbunden, denn es wird angenommen, dass  $b_1 \neq 0$  und  $b_2 \neq 0$  gilt. Die Rangbedingung

$$\text{s-Rang} ([\mathbf{A} \ | \ \mathbf{b}]) = \text{s-Rang} \begin{pmatrix} 0 & 0 & * \\ 0 & 0 & * \end{pmatrix} = 1$$

zeigt, dass die Bedingung an den strukturellen Rang dieser Matrix nicht erfüllt und das System somit nicht strukturell steuerbar ist.



**Abb. A.27:** Strukturgraph zweier parallel geschalteter Integratoren

Dieses Beispiel zeigt, dass die beiden Bedingungen von Satz 5.3 tatsächlich unabhängig voneinander sind. Da die Rangbedingung häufig erfüllt ist, wenn das System eingangsverbunden ist, ist dieses Beispiel bemerkenswert.

Das Beispiel ist ein System, das aus zwei parallel geschalteten Integratoren besteht. Da es nicht strukturell steuerbar ist, ist es auch nicht vollständig steuerbar, d. h., die beiden Zustandsvariablen sind linear voneinander abhängig. Diese Tatsache sieht man aus der Lösung der Zustandsgleichung, die auf die Beziehungen

$$\begin{aligned}x_1(t) &= x_{10} + b_1 \int_0^t u(\tau) d\tau \\x_2(t) &= x_{20} + b_2 \int_0^t u(\tau) d\tau,\end{aligned}$$

und

$$x_1(t) = \frac{b_1}{b_2} x_2(t) - \frac{b_1}{b_2} x_{20} + x_{10} \quad (\text{A.28})$$

führt. Die vollständige Steuerbarkeit verlangt, dass man für einen beliebig vorgegebenen Zustand

$$\mathbf{x}(t_e) = (x_1(t_e) \ x_2(t_e))^T$$

eine Steuerung  $u_{[0,t_e]}$  finden kann, die das System in den Endzustand  $\mathbf{x}(t_e)$  überführt. Für das hier betrachtete System können aber nur Endzustände angesteuert werden, bei denen sich die Zustandsvariablen  $x_1$  und  $x_2$  entsprechend Gl. (A.28) um einen von ihren Anfangswerten festgelegten Betrag unterscheiden.

Die Rangbedingung ist gegenüber der Eingangsverbundenheit nur für Systeme eine Zusatzbedingung, deren Eigenwerte sämtlich gleich null sind. Sobald die Matrix  $\mathbf{A}$  des hier betrachteten Beispiels in den Hauptdiagonalelementen mindestens ein von null verschiedenes Element enthält, ist die Rangbedingung erfüllt.

Übrigens ist das angegebene System auch nicht strukturell beobachtbar.

### Aufgabe 5.6 Strukturelle Analyse des Modells für den Abbremsvorgang eines Fahrzeugs

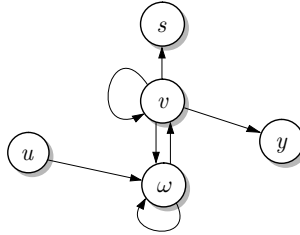
1. Der Strukturgraph besteht aus Knoten für den Eingang  $u$ , die Zustandsvariablen  $\omega$ ,  $v$  und  $s$  sowie den Ausgang  $y = v$ . Um festzustellen, zwischen welchen Knoten gerichtete Kanten auftreten, werden die gegebenen Gleichungen in die Standardform des nichtlinearen Zustandsraummodells gebracht

$$\text{Fahrzeug : } \begin{cases} \begin{pmatrix} \dot{v}(t) \\ \dot{\omega}(t) \\ \dot{s}(t) \end{pmatrix} = \begin{pmatrix} -f_2(f_1(v(t), \omega(t)))g \\ \frac{mr}{J} (f_2(f_1(v(t), \omega(t)))g - u(t)) \\ v(t) \end{pmatrix} \\ y(t) = v(t) \end{cases}$$

und die Abhängigkeiten zwischen den Zustandsvariablen untersucht. Da in der ersten Zeile auf der rechten Seite  $v(t)$  und  $\omega(t)$  auftreten, gibt es im Strukturgraphen gerichtete Kanten von  $v$  und  $\omega$  zum Knoten  $v$ . Die zweite Zeile führt auf Kanten von  $u$  und  $v$  zum Knoten  $\omega$ , die dritte Zeile zu einer Kante von  $v$  nach  $s$  und die vierte Zeile zu einer Verbindung von  $v$  nach  $y$  (Abb. A.28).

Im Vergleich zum Blockschaltbild 3.16 auf S. 109 bezieht sich der Strukturgraph nur auf die als Zustandsvariablen wirkenden Signale  $\omega(t)$ ,  $v(t)$  und  $s(t)$ , also diejenigen Signale, die vor den Integratoren stehen. Die übrigen Signalwege sind zu Kanten zusammengefasst, so dass die Kopplungen





**Abb. A.28:** Strukturgraph des Modells zur Beschreibung des Abbremsvorgangs

direkt zu sehen sind. Der Strukturgraph enthält also weniger Informationen; man könnte allerdings mit einiger Übung die strukturelle Analyse der Steuerbarkeit und Beobachtbarkeit auch direkt am Blockschaltbild durchführen.

- Das System ist strukturell steuerbar, aber nicht strukturell beobachtbar, denn der Graph ist nicht Ausgangsverbunden. Nicht beobachtbar ist der Weg  $s(t)$ , den man nicht aus der Geschwindigkeitsmessung bestimmen kann. Dieses möglicherweise auf den ersten Blick überraschende Ergebnis entsteht dadurch, dass man bei der Betrachtung der Beobachtbarkeit davon ausgeht, dass der Anfangszustand nicht bekannt ist, also für das hier verwendete Modell nicht bekannt ist, bei welcher Position  $s(0)$  die Bewegung beginnt. Dann kann man auch den aktuellen Weg nicht berechnen, denn zur Bestimmung der aktuellen Position aus der Geschwindigkeit braucht man  $s(0)$ :

$$s(t) = s(0) + \int_0^t v(\tau) d\tau.$$

Andererseits zeigt der Strukturgraph, dass aus der gemessenen Geschwindigkeit  $v(t)$  die Winkelgeschwindigkeit  $\omega(t)$  des Rades (und folglich auch der Schlupf  $\lambda(t)$ ) bestimmt werden kann.

Der Strukturgraph gilt jedoch nicht in allen Arbeitspunkten. Untersucht man die ersten beiden in der Aufgabenstellung gegebenen Gleichungen genauer, so erhält man

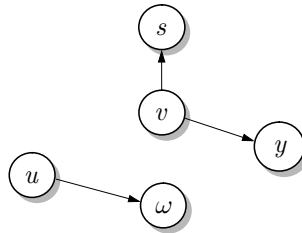
$$\frac{\partial \dot{v}}{\partial v} = \frac{\partial f_2}{\partial f_1} \frac{\partial f_1}{\partial \lambda} \frac{\partial \lambda}{\partial v} = -g k_{\mu\lambda} \frac{\omega r}{v^2},$$

wobei  $k_{\mu\lambda}$  die Steigung der  $\mu(\lambda)$ -Kennlinie bezeichnet. Wenn der Arbeitspunkt im Maximum dieser Kennlinie liegt, so ist dieser Anstieg gleich null. Folglich entfällt im Strukturgraphen die Kante zwischen  $\dot{v}$  und  $v$ . Gleiches trifft auf die Kanten zwischen  $\omega$  und  $v$  sowie die Schleife am Knoten  $\omega$  zu. Der Strukturgraph erhält die in Abb. A.29 gezeigte Form. Offensichtlich ist die strukturelle Steuerbarkeit in diesem Arbeitspunkt verloren gegangen.

### Aufgabe 5.8 Beobachtbarkeit eines Kommunikationssatelliten

Es muss zunächst das Zustandsraummodell für die Satellitenbewegung aufgestellt werden. Wählt man den Zustandsvektor

$$\mathbf{x}(t) = \begin{pmatrix} v(t) \\ \dot{v}(t) \\ h(t) \\ \dot{h}(t) \end{pmatrix},$$



**Abb. A.29:** Strukturgraph für den Arbeitspunkt im Maximum der  $\mu(\lambda)$ -Kennlinie

so kann man aus den angegebenen Gleichungen das folgende Modell ablesen:

$$\text{Satellit : } \left\{ \begin{array}{l} \begin{pmatrix} \dot{v}(t) \\ \ddot{v}(t) \\ \dot{h}(t) \\ \ddot{h}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 3\omega^2 & 0 & 0 & 2\omega \\ 0 & 0 & 0 & 1 \\ 0 & -2\omega & 0 & 0 \end{pmatrix} \begin{pmatrix} v(t) \\ \dot{v}(t) \\ h(t) \\ \dot{h}(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u(t) \\ \\ y(t) = (0 \ 0 \ 1 \ 0) \begin{pmatrix} v(t) \\ \dot{v}(t) \\ h(t) \\ \dot{h}(t) \end{pmatrix} \end{array} \right.$$

Die Beobachtbarkeitsmatrix für dieses System lautet

$$\mathbf{S}_B = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2\omega & 0 & 0 \\ -6\omega^3 & 0 & 0 & -4\omega^2 \end{pmatrix}.$$

Sie hat vollen Rang 4. Also ist der Zustand des Satelliten, der die Lage des Satelliten im  $v/h$ -Koordinatensystem beschreibt und außerdem die Geschwindigkeit in beide Richtungen angibt, aus der Messgröße  $y(t) = h(t)$  beobachtbar.

### **Aufgabe 6.1** Stabilitätsanalyse eines Wärmeübertragers

Nach dem Einsetzen der Parameterwerte erhält man für die Systemmatrix des Wärmeübertragers

$$\mathbf{A} = \begin{pmatrix} -4,14 & 3,14 \\ 6,28 & -9,28 \end{pmatrix}$$

und das charakteristische Polynom

$$\det(\lambda \mathbf{I} - \mathbf{A}) = \lambda^2 + 13,42\lambda + 18,7 = 0.$$

Die Bedingung, dass alle Koeffizienten positiv sind, ist erfüllt. Für die weiteren Schritte der Stabilitätsprüfung muss die Hurwitzmatrix der Dimension (2 x 2) aufgestellt werden:

$$\mathbf{H} = \begin{pmatrix} a_1 & a_3 \\ a_0 & a_2 \end{pmatrix} = \begin{pmatrix} 13,42 & 0 \\ 18,7 & 1 \end{pmatrix}.$$

Die Determinanten heißen

$$D_1 = 13,42 > 0$$

$$D_2 = \det \begin{pmatrix} 13,42 & 0 \\ 18,7 & 1 \end{pmatrix} = 13,42 \cdot 1 > 0,$$

woraus die asymptotische Stabilität des Wärmeübertragers folgt.

**Diskussion.** Dieses sehr einfache Beispiel zeigt, dass man die Stabilität des Wärmeübertragers überprüfen kann, ohne die Eigenwerte zu berechnen (die bei den hier angegebenen Parameterwerten bei  $-1,58$  und  $-11,84$  liegen). Da es sich um ein System zweiter Ordnung handelt, hätte man sich die Aufstellung der Hurwitzmatrix ersparen können, weil dann die Vorzeichenbedingung für die Koeffizienten des charakteristischen Polynoms nicht nur notwendig, sondern auch hinreichend für die Stabilität ist.

Das Ergebnis dieser Stabilitätsanalyse ist nicht überraschend, denn ein Wärmeübertrager ist natürlich ein stabiles System. Weichen die Temperaturen zur Zeit  $t = 0$  von den stationären Arbeitspunktwerten ab, so nähern sie sich für  $t \rightarrow \infty$  asymptotisch wieder den Arbeitspunktwerten an. Diese Eigenschaft ist offenbar unabhängig von den konkreten Parameterwerten.

Dass dies so ist, kann man anhand der Systemmatrix

$$\mathbf{A} = \begin{pmatrix} -\frac{kA+w_1c_1}{m_1c_1} & +\frac{kA}{m_1c_1} \\ +\frac{kA}{m_2c_2} & -\frac{kA+w_2c_2}{m_2c_2} \end{pmatrix}$$

erkennen, wenn man die Stabilitätsanalyse ohne das Einsetzen konkreter Parameterwerte durchführt. Die charakteristische Gleichung lautet

$$\begin{aligned} & \det(\lambda \mathbf{I} - \mathbf{A}) \\ &= \lambda^2 + \left( \frac{kA+w_1c_1}{m_1c_1} + \frac{kA+w_2c_2}{m_2c_2} \right) \lambda + \frac{(kA+w_1c_1)(kA+w_2c_2)}{m_1m_2c_1c_2} - \frac{k^2A^2}{m_1m_2c_1c_2} \\ &= \lambda^2 + \left( \frac{kA+w_1c_1}{m_1c_1} + \frac{kA+w_2c_2}{m_2c_2} \right) \lambda + \frac{kA(w_1c_1+w_2c_2)+w_1w_2c_1c_2}{m_1m_2c_1c_2}. \end{aligned}$$

Für beliebige physikalisch sinnvolle (positive) Parameterwerte sind sämtliche Koeffizienten der charakteristischen Gleichung positiv, so dass der Wärmeübertrager als ein System zweiter Ordnung asymptotisch stabil ist.

### Aufgabe 6.3 Stabilität des Unterwasserfahrzeugs

Für die Linearisierung ist zunächst die Gleichgewichtslage zu bestimmen, die man aus  $\dot{\mathbf{x}} = \mathbf{0}$  erhält:

$$\bar{x}_1 = \frac{m - V_0\rho_0}{(\kappa_W - \kappa_D)\rho_0^2gV_0}$$

$$\bar{x}_2 = 0.$$

Dafür erhält man die linearisierte Zustandsgleichung

$$\dot{\mathbf{x}}(t) = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{V_0\rho_0^2g^2}{m}(\kappa_W - \kappa_D) & -\frac{\kappa_W}{m} \end{pmatrix} \mathbf{x}(t).$$

Alle physikalischen Parameter sind positiv. Das Vorzeichen der Differenz

$$\kappa = \kappa_W - \kappa_D,$$

die mit  $\kappa$  abgekürzt wird, beeinflusst das Vorzeichen des linken unteren Elements der Systemmatrix. Vereinfacht kann man die Systemmatrix deshalb mit positiven Konstanten  $a_{21}$  und  $a_{22}$  als

$$A = \begin{pmatrix} 0 & 1 \\ -a_{21}\kappa & -a_{22} \end{pmatrix}$$

schreiben. Das charakteristische Polynom lautet dann

$$\lambda^2 + a_{22}\lambda + a_{21}\kappa = 0.$$

Da ein System zweiter Ordnung genau dann stabil ist, wenn alle Koeffizienten positiv sind, muss die Differenz

$$\kappa_W - \kappa_D > 0$$

positiv sein, damit das Unterwasserfahrzeug eine stabile Gleichgewichtslage hat.

**Interpretation.** Die Stabilitätsbedingung besagt, dass das Wasser mehr zusammengedrückt werden muss als der Drifter, wenn man sich in Richtung größerer Tiefe bewegt. Befindet sich der Drifter in einer stabilen Ruhelage und wird durch eine Störung ein wenig in die Tiefe gedrückt, verringert sich sein Volumen, aber das umgebende Wasser wird im Verhältnis dazu stärker verdichtet. Das heißt, dass die Masse des verdrängten Wassers ansteigt und der dadurch erhöhte Auftrieb das Unterwasserfahrzeug wieder in Richtung der Ruhelage beschleunigt. Die Auftriebskraft wirkt also wie eine Rückstellfeder. Die Ruhelage ist stabil.

Diese Bedingung ist in der Praxis erfüllt. Typische Werte für die Kompressibilitäten sind  $\kappa_W = 4,9 \cdot 10^{-10} \approx 1/\text{Pa}$  und  $\kappa_D = 2,54 \cdot 10^{-10} \approx 1/\text{Pa}$ . Die Stabilitätsbedingung ist also erfüllt, wenn man den Drifter steif genug konstruiert.

#### Aufgabe 6.4 Stabilitätsanalyse des Schlupfverhaltens beim Abbremsen eines Fahrzeugs

1. Da es sich bei Gl. (6.17) um ein Zustandsraummodell erster Ordnung handelt, kann man die Stabilität sofort an dem vor  $\omega$  auf der rechten Seite stehenden Faktor ablesen, der für ein stabiles System negativ sein muss. Ausschlaggebend für die Stabilität ist folglich das Vorzeichen von  $-\frac{m r^2 q}{J \bar{v}} k_{\mu\lambda}$ , das von der Steigung  $k_{\mu\lambda}$  der  $\mu(\lambda)$ -Kennlinie abhängt. Der Arbeitspunkt  $A_1$  ist stabil, der Arbeitspunkt  $A_2$  aber instabil.
2. Die Stabilität des Arbeitspunkts  $A_1$  bedeutet, dass das System nach einer Auslenkung aus diesem Arbeitspunkt wieder in diesen Arbeitspunkt zurückkehrt. Anhand von Abb. A.17 lässt sich erkennen, warum dies so ist. Wenn man die Kette der Blöcke im Vorwärtszweig entlang geht, so erkennt man folgende Wirkungen einer positiven Eingangsgröße ( $\delta$  bezeichnet die Signaländerung):

$$\delta u > 0 \Rightarrow \delta \dot{\omega} < 0 \Rightarrow \delta \omega < 0 \Rightarrow \delta \lambda > 0 \Rightarrow \delta \mu > 0 \Rightarrow \delta y > 0.$$

Damit wird die Differenz  $\delta y - \delta u$  an der Summationsstelle kleiner und die absoluten Beträge der angegebenen Wirkungen werden kleiner. Das System bewegt sich in Richtung zum Gleichgewichtspunkt. Wird dann  $\delta u = 0$  gesetzt, so kehrt das System auf Grund ähnlicher Ursache-Wirkungsbeziehungen wieder in den alten Arbeitspunkt  $A_1$  zurück.

Bei einer Auslenkung aus dem Arbeitspunkt  $A_2$  entstehen am Anfang der Kette dieselben Wirkungen, allerdings ist  $\delta\mu < 0$  und folglich  $\delta y < 0$ , weil jetzt  $k_{\mu\lambda} < 0$  ist. Damit erhöht sich die Differenz  $\delta y - \delta u$  an der Summationsstelle und die Wirkungskette gilt mit steigenden absoluten Beträgen der angegebenen Signale. Das System entfernt sich immer weiter vom Arbeitspunkt  $A_2$ . Für positive Erregung  $\delta u > 0$  läuft das System in den Punkt  $A_3$ , in dem das Rad blockiert, bei negativer Erregung in den Arbeitspunkt  $A_1$ . Deshalb kann man das System also niemals im Arbeitspunkt  $A_2$  betreiben.

Liegt der Arbeitspunkt im Maximum der Kennlinie, so ist  $k_{\mu\lambda} = 0$ , der Arbeitspunkt also grenzstabil. Das heißt, dass bei einer Auslenkung  $\delta\lambda < 0$  der Arbeitspunkt nach links verschoben wird, das Verhalten dort durch ein Modell mit positivem Parameter  $k_{\mu\lambda}$  beschrieben ist und der Arbeitspunkt wieder in das Maximum zurückgeschoben wird. Bei einer Auslenkung  $\delta\lambda > 0$  wird der Arbeitspunkt auf den fallenden Kennlinienteil verschoben, wo  $k_{\mu\lambda} < 0$  gilt und der Arbeitspunkt folglich an die Begrenzung  $\lambda = 1$  geschoben wird. Das Rad blockiert.

Aus praktischer Sicht ist die Grenzstabilität gleichbedeutend mit einer Instabilität, denn man muss nur so lange warten, bis eine positive Schlupfänderung um den Arbeitspunkt eintritt, wodurch das System in den Punkt  $A_3$  gelenkt wird.

Da diese Stabilitätseigenschaften mit Hilfe eines linearisierten Modells abgeleitet wurden, gelten sie nur in der Umgebung der betrachteten Arbeitspunkte. Diese Umgebung wird hier durch das Maximum der  $\mu(\lambda)$ -Kennlinie begrenzt. Auf beiden Seiten dieses Maximums kann die Kurve in guter Näherung durch eine Gerade approximiert werden, so dass die Stabilität nicht für alle Punkte rechts bzw. links des Maximums getrennt untersucht werden muss.

### Aufgabe 6.5 Stabilisierung eines Systems zweiter Ordnung

- Abbildung A.30 (links) zeigt den betrachteten Regelkreis, der zur Störkompensation dient und deshalb keine Führungsgröße hat. Die Stabilität dieses Regelkreises wird im Folgenden untersucht.

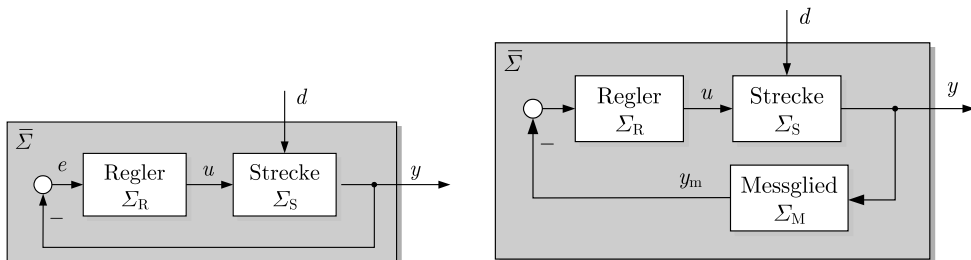


Abb. A.30: Regelkreis (links) und Regelkreis mit Messglied (rechts)

- Durch Verknüpfung der Modelle der Strecke und des Reglers erhält man für den Regelkreis das Zustandsraummodell

$$\bar{\Sigma} : \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} -\frac{1}{T_1} & -\frac{k_P k_S}{T_1} \\ \frac{1}{T_2} & -\frac{1}{T_2} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ \frac{k_d}{T_2} \end{pmatrix} d(t) \\ y(t) = (0 \quad k_S) \mathbf{x}(t), \end{cases}$$

dessen Systemmatrix

$$\bar{\mathbf{A}} = \begin{pmatrix} -\frac{1}{T_1} & -\frac{k_P k_s}{T_1} \\ \frac{1}{T_2} & -\frac{1}{T_2} \end{pmatrix}$$

für die Stabilität maßgebend ist. Das charakteristische Polynom dieser Matrix lautet

$$\det(\lambda \mathbf{I} - \bar{\mathbf{A}}) = \lambda^2 + \left( \frac{1}{T_1} + \frac{1}{T_2} \right) \lambda + \left( \frac{k_P k_s}{T_1 T_2} + \frac{1}{T_1 T_2} \right) = 0.$$

Es führt entsprechend dem Hurwitzkriterium für beliebige Zeitkonstanten  $T_1$  und  $T_2$  auf einen stabilen Regelkreis, sofern die in der Aufgabenstellung angegebene Bedingung  $k_P k_s > 0$  erfüllt ist. Da es sich bei diesem Regelkreis um ein System zweiter Ordnung handelt, muss nur geprüft werden, dass alle drei Koeffizienten des charakteristischen Polynoms positiv sind.

3. Um die bleibende Regelabweichung auszurechnen, wird die statische Verstärkung des Regelkreises zwischen der Störgröße  $d(t)$  und der Regelabweichung  $e(t) = -y(t)$  ermittelt. Das Modell des Regelkreises  $\tilde{\Sigma}$  wird dafür mit der Ausgangsgleichung

$$e(t) = -(0 \quad k_s) \mathbf{x}(t)$$

versehen. Den Verstärkungsfaktor erhält man dann entsprechend Gl. (3.24), die hier für die Eingangsgröße  $d(t)$  und die Ausgangsgröße  $e(t)$  angewendet wird:

$$\begin{aligned} k_{\text{ed}} &= (0 \quad k_s) \begin{pmatrix} -\frac{1}{T_1} & -\frac{k_P k_s}{T_1} \\ \frac{1}{T_2} & -\frac{1}{T_2} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \frac{k_d}{T_2} \end{pmatrix} \\ &= -\frac{k_s k_d}{1 + k_P k_s}. \end{aligned}$$

Die Forderung  $|k_{\text{ed}}| \leq 0,05$  ist also für

$$\begin{aligned} |1 + k_P k_s| &\geq \left| \frac{k_s k_d}{0,05} \right| \\ k_P k_s &\geq 20 |k_s k_d| - 1 \end{aligned}$$

erfüllt. Da die Stabilität des Regelkreises für alle Reglerparameter  $k_P$ , für die  $k_P k_s > 0$  gilt, gewährleistet ist, kann  $k_P$  so groß gewählt werden, dass diese Ungleichung erfüllt ist. Der Regler kann also so eingestellt werden, dass der Regelkreis beide Forderungen erfüllt, nämlich asymptotisch stabil ist und eine bleibende Regelabweichung von weniger als 5% besitzt.

4. Das erweiterte Blockschaltbild ist im rechten Teil von Abb. A.30 zu sehen. Das Messglied ist in die Rückführung eingetragen, weil  $y$  und nicht die Messgröße  $y_m$  die Regelgröße ist.
5. Die Reihenschaltung  $\tilde{\Sigma}$  von Regelstrecke und Messglied hat drei Zustandsvariable, nämlich die beiden Zustandsgrößen  $x_1$  und  $x_2$  der Strecke und die Zustandsvariable  $x_m$  des Messglieds. Sie ist durch das Modell

$$\tilde{\Sigma} : \begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_m(t) \end{pmatrix} = \begin{pmatrix} -\frac{1}{T_1} & 0 & -\frac{k_P}{T_1} \\ \frac{1}{T_2} & -\frac{1}{T_2} & 0 \\ 0 & \frac{k_s}{T_m} & -\frac{1}{T_m} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_m(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{k_d}{T_2} \\ 0 \end{pmatrix} d(t) \\ \\ \hat{y}(t) = (0 \quad 0 \quad 1) \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_m(t) \end{pmatrix} \end{cases}$$

beschrieben. Für die Systemmatrix

$$\mathbf{A} = \begin{pmatrix} -\frac{1}{T_1} & 0 & -\frac{k_P}{T_1} \\ \frac{1}{T_2} & -\frac{1}{T_2} & 0 \\ 0 & \frac{k_S}{T_m} & -\frac{1}{T_m} \end{pmatrix}$$

erhält man die charakteristische Gleichung

$$\lambda^3 + \left( \frac{1}{T_1} + \frac{1}{T_2} + \frac{1}{T_m} \right) \lambda^2 + \left( \frac{1}{T_1 T_2} + \frac{1}{T_1 T_m} + \frac{1}{T_2 T_m} \right) \lambda + \left( \frac{1}{T_1 T_2 T_m} + \frac{k_P k_S}{T_1 T_2 T_m} \right) = 0.$$

Alle Koeffizienten sind positiv. Aus der Hurwitzmatrix

$$\mathbf{H} = \begin{pmatrix} \frac{1}{T_1 T_2} + \frac{1}{T_1 T_m} + \frac{1}{T_2 T_m} & 1 & 0 \\ \frac{1+k_P k_S}{T_1 T_2 T_m} & \frac{1}{T_1} + \frac{1}{T_2} + \frac{1}{T_m} & 0 \\ 0 & \frac{1}{T_1 T_2} + \frac{1}{T_1 T_m} + \frac{1}{T_2 T_m} & 1 \end{pmatrix}$$

erhält man die zusätzliche Bedingung

$$\left( \frac{1}{T_1 T_2} + \frac{1}{T_1 T_m} + \frac{1}{T_2 T_m} \right) \cdot \left( \frac{1}{T_1} + \frac{1}{T_2} + \frac{1}{T_m} \right) - \frac{1+k_P k_S}{T_1 T_2 T_m} > 0,$$

die in

$$k_P k_S < (T_1 + T_2 + T_m) \left( \frac{1}{T_1} + \frac{1}{T_2} + \frac{1}{T_m} \right) - 1$$

umgeformt werden kann. Der Einbau des Messgliedes führt also auf eine zusätzliche obere Schranke für den Reglerparameter  $k_P$ . Damit wird der Spielraum eingeschränkt, innerhalb dessen man den Reglerparameter auch im Hinblick auf die Erfüllung der Forderungen an die bleibende Regelabweichung auswählen kann. Ob die Wirkung der Störung auf 5% reduziert werden kann, hängt von den Parametern  $T_1$ ,  $T_2$  und  $T_m$  der Regelstrecke und des Messgliedes ab.

### Aufgabe 6.6 Stabilitätsprüfung des geregelten invertierten Pendels

1. Das invertierte Pendel hat das Zustandsraummodell (6.31)

$$\text{Pendel : } \begin{cases} \begin{pmatrix} \dot{\phi}(t) \\ \ddot{\phi}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{3}{ml^2} \end{pmatrix} u(t), & \mathbf{x}(0) = \begin{pmatrix} \phi(0) \\ \dot{\phi}(0) \end{pmatrix} \\ y(t) = (1 \ 0) \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix}. \end{cases} \quad (\text{A.29})$$

Aus der Systemmatrix erhält man folgendes charakteristisches Polynom:

$$\det \left( \lambda \mathbf{I} - \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} & 0 \end{pmatrix} \right) = \lambda^2 - \frac{3g}{2l}.$$

Dieses Polynom kann das Hurwitzkriterium nicht erfüllen, weil der Summand mit  $\lambda$  fehlt (also  $a_1 = 0$  gilt) und weil der rechte Summand negativ ist. Das Pendel ist also für beliebige (physikalisch sinnvolle) Parameterwerte instabil.

## 2. Versucht man mit dem Regler

$$\Sigma_R : u(t) = -k_P y(t)$$

das Pendel zu stabilisieren, so erhält man nach dem Einsetzen des Reglergesetzes in das Pendelmodell (A.29) für den Regelkreis das Zustandsraummodell

$$\text{geregeltes Pendel : } \begin{cases} \begin{pmatrix} \dot{\phi}(t) \\ \ddot{\phi}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{3g}{2l} - k_P \frac{3}{ml^2} & 0 \end{pmatrix} \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} \\ y(t) = (1 \ 0) \begin{pmatrix} \phi(t) \\ \dot{\phi}(t) \end{pmatrix} \end{cases}$$

und daraus das charakteristische Polynom

$$\lambda^2 - \frac{3g}{2l} + k_P \frac{3}{ml^2} = 0,$$

bei dem der Term  $a_1 \lambda$  fehlt. Mit dem Hurwitzkriterium erkennt man deshalb sofort, dass das Pendel für keine Wahl des Reglerparameters  $k_P$  asymptotisch stabil ist, weil der Summand mit  $\lambda$  fehlt. Das heißt, es ist nicht möglich, das invertierte Pendel dadurch zu stabilisieren, dass man die Kraft  $u(t)$  proportional dem Pendelwinkel  $y(t) = \phi(t)$  anpasst.

### Aufgabe 6.7 Stabilität des Temperaturregelkreises eines Reaktors

Für die Stabilitätsanalyse muss zunächst das Zustandsraummodell des Regelkreises  $\bar{\Sigma}$  aufgestellt werden. Es handelt sich hier um ein rückgekoppeltes System, dessen Zustand  $\bar{x}$  sich aus dem dreidimensionalen Zustandsvektor  $x$  der Regelstrecke und dem skalaren Zustand  $x_r$  des Reglers zusammensetzt:

$$\bar{x}(t) = \begin{pmatrix} x(t) \\ x_r(t) \end{pmatrix}$$

(vgl. Abschn. 3.6). Durch Verknüpfung der Gln. (6.32) für die Regelstrecke und (6.33) für den Regler erhält man das gesuchte Modell:

$$\bar{\Sigma} : \begin{cases} \begin{pmatrix} \dot{x}(t) \\ \dot{x}_r(t) \end{pmatrix} = \begin{pmatrix} -8,16 & -17,82 & -10,03 & -k_I \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 14,36 & 0 \end{pmatrix} \begin{pmatrix} x(t) \\ x_r(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} w(t) \\ \begin{pmatrix} x(0) \\ x_r(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ x_{r0} \end{pmatrix} \\ y(t) = (0 \ 0 \ 14,36 \ 0) \begin{pmatrix} x(t) \\ x_r(t) \end{pmatrix}. \end{cases}$$

Für die Stabilität ist die Systemmatrix maßgebend, deren charakteristische Gleichung folgendermaßen lautet:

$$\det \left( \lambda I - \begin{pmatrix} -8,16 & -17,82 & -10,03 & -k_I \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 14,36 & 0 \end{pmatrix} \right) = \det \begin{pmatrix} \lambda + 8,16 & 17,82 & 10,03 & k_I \\ -1 & \lambda & 0 & 0 \\ 0 & -1 & \lambda & 0 \\ 0 & 0 & -14,36 & \lambda \end{pmatrix}$$



$$\begin{aligned}
&= \lambda \det \begin{pmatrix} \lambda + 8,16 & 17,82 & 10,03 \\ -1 & \lambda & 0 \\ 0 & -1 & \lambda \end{pmatrix} - k_I \det \begin{pmatrix} -1 & \lambda & 0 \\ 0 & -1 & \lambda \\ 0 & 0 & -14,36 \end{pmatrix} \\
&= \lambda^4 + 8,16\lambda^3 + 17,82\lambda^2 + 10,03\lambda + 14,36k_I.
\end{aligned}$$

Dabei wurde die erste Determinante nach der letzten Spalte entwickelt.

Das Hurwitzkriterium fordert erstens, dass alle Koeffizienten des charakteristischen Polynoms positiv sind. Daraus erhält man die erste Forderung

$$k_I > 0$$

an den Parameter  $k_I$  des I-Reglers.

Zweitens muss man die Hurwitzmatrix der Ordnung vier aufstellen:

$$\mathbf{H} = \begin{pmatrix} 10,03 & 8,16 & 0 & 0 \\ 14,36k_I & 17,82 & 1 & 0 \\ 0 & 10,03 & 8,16 & 0 \\ 0 & 14,36k_I & 17,82 & 1 \end{pmatrix}.$$

Die vier Determinantenbedingungen führen auf folgende Ungleichungen:

$$\begin{aligned}
D_1 &= 10,03 > 0 \\
D_2 &= 10,03 \cdot 17,82 - 14,36 \cdot 8,16 \cdot k_I > 0 \\
D_3 &= 8,16 \cdot D_2 - 10,03^2 > 0 \\
D_4 &= D_3 > 0.
\end{aligned}$$

Für die Reglerparameter sind die zweite und dritte Ungleichung wichtig, aus denen man

$$k_I < 1,53$$

$$k_I < 1,42$$

erhält. Die zweite Bedingung ist die schärfere von beiden. Zusammengefasst erhält man einen stabilen Regelkreis, sofern der Reglerparameter  $k_I$  im Intervall

$$0 < k_I < 1,42 \quad (\text{A.30})$$

liegt.

### Aufgabe 6.8 Robustheit eines Antiblockiersystems

1. Der Zustand des Regelkreises setzt sich aus der Zustandsgröße  $x(t) = \omega(t)$  der Regelstrecke und der Zustandsgröße  $x_r(t)$  des Reglers zusammen. Mit den Abkürzungen

$$a = -\frac{mr^2g}{J\bar{v}}k_{\mu\lambda}, \quad b = -\frac{mr}{J} \quad \text{und} \quad c = -\frac{r}{\bar{v}}$$

erhält man für den Regelkreis  $\bar{\Sigma}$  die Gleichungen

$$\bar{\Sigma} : \begin{cases} \begin{pmatrix} \dot{x}(t) \\ \dot{x}_r(t) \end{pmatrix} = \underbrace{\begin{pmatrix} a - bk_P c & -bk_I \\ c & 0 \end{pmatrix}}_{\bar{A}} \begin{pmatrix} x(t) \\ x_r(t) \end{pmatrix} + \begin{pmatrix} bk_P \\ -1 \end{pmatrix} w(t) \\ y(t) = (c \ 0) \begin{pmatrix} x(t) \\ x_r(t) \end{pmatrix}. \end{cases}$$

2. Die charakteristische Gleichung des Regelkreises lautet

$$\det(s\mathbf{I} - \bar{A}) = \det \begin{pmatrix} s - a + bk_P c & bk_I \\ -c & s \end{pmatrix} = s^2 + (-a + bk_P c)s + bck_I = 0,$$

wobei hier, um Verwechslungen mit dem Schlupf  $\lambda$  zu vermeiden,  $s$  als unabhängige Variable verwendet wurde. Nach dem Hurwitzkriterium ist das System genau dann stabil, wenn die beiden Bedingungen

$$\begin{aligned} -a + bk_P c &= \frac{mr^2 g k_{\mu\lambda}}{J\bar{v}} + \frac{mr}{J} k_P \frac{r}{\bar{v}} > 0 \\ bck_I &= \frac{mr^2}{J\bar{v}} k_I > 0, \end{aligned}$$

also

$$k_P > -g k_{\mu\lambda} \quad (\text{A.31})$$

$$k_I > 0 \quad (\text{A.32})$$

erfüllt sind.

3. Die Robustheitsuntersuchung verlangt, die Extremwerte  $k_{\mu\lambda, \min}$  und  $k_{\mu\lambda, \max}$  von  $k_{\mu\lambda}(\bar{\lambda})$  zu bestimmen und dann nach Reglerparametern zu suchen, für die die Stabilitätsbedingungen (A.31) und (A.32) erfüllt sind. Aus Abb. 3.15 erhält man etwa die folgenden Werte:

$$k_{\mu\lambda, \min} = -0,5$$

$$k_{\mu\lambda, \max} = 10.$$

Deshalb ist der Regelkreis unter allen Umständen stabil, wenn

$$k_P > 0,5g$$

$$k_I > 0$$

ist, also eine Mindestverstärkung von  $0,5g$  des Proportionalanteils und eine positive Verstärkung des Integralanteils verwendet wird. Demzufolge gibt es keinen reinen I-Regler, für den der Regelkreis stabil ist.

**Diskussion.** Die Analyse zeigt, dass es Regler gibt, die die geforderten Arbeitspunkte unter allen Straßenverhältnissen stabilisieren. Da für die Analyse linearisierte Modelle verwendet wurden, muss man nun untersuchen, inwieweit sich diese Ergebnisse auf das nichtlineare System übertragen lassen. Dies führt hier auf keine wesentlichen Voraussetzungen, weil die  $\mu(\lambda)$ -Kennlinie auf beiden Seiten des Maximums in guter Näherung durch Geraden (also lineare Modelle) approximiert werden kann.

Die robuste Stabilität ist jedoch nicht die einzige Forderung, die man an ein Antiblockiersystem stellt. Es muss auch untersucht werden, welche Regelgüte das Antiblockiersystem unter allen Fahrbahnverhältnissen aufweist, wie also der dynamische Übergangsvorgang bei einem Abbremsvorgang aussieht. Darauf wird im Kap. 8 eingegangen.

**Aufgabe 7.3 Verhalten des Gleichstrommotors bei Störungen und Modellunbestimmtheiten**

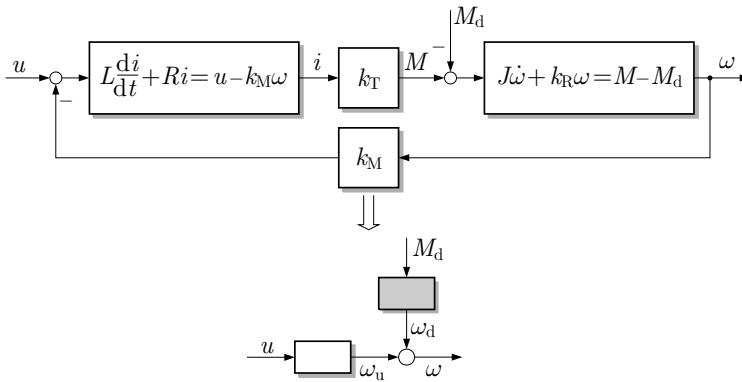
Das Moment  $M_d(t)$  wirkt als zusätzliche Eingangsgröße auf den Motor, die vom Antriebsmoment  $M(t)$  in Gl. (3.10) subtrahiert werden muss:

$$M(t) - M_R(t) - M_B(t) - M_d(t) = 0.$$

Die im Beispiel 3.2 auf S. 90 angegebenen Gleichungen führen mit dieser Ergänzung auf die Beziehungen

$$\begin{aligned} \frac{di(t)}{dt} &= -\frac{R}{L}i(t) - \frac{k_M}{L}\omega(t) + \frac{1}{L}u(t) \\ \frac{d\omega(t)}{dt} &= \frac{k_T}{J}i(t) - \frac{k_R}{J}\omega(t) - \frac{1}{J}M_d(t), \end{aligned}$$

in der das Störmoment als zusätzliches Signal für die Berechnung der Drehgeschwindigkeit vorkommt. In Abb. A.31 ist dieses Modell durch ein Blockschaltbild veranschaulicht, in dessen Zusammenfassung zu zwei Blöcken die Wirkung des Störmoments durch ein Störgrößenmodell dargestellt wird.



**Abb. A.31:** Gleichstrommotor mit zusätzlichem Störmoment  $M_d(t)$

Wenn der Motor geschwindigkeitsabhängig gebremst wird, entsteht eine zusätzliche Rückwirkung der Drehzahl auf das Drehmoment (Abb. A.32), die durch

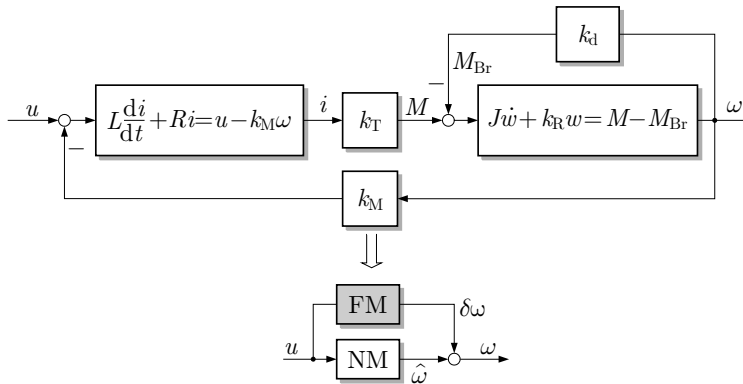
$$M_{Br}(t) = k_d \omega(t)$$

beschrieben wird. Das Moment  $M_{Br}(t)$  verändert die Momentengleichung (3.10) in gleicher Weise wie zuvor das Moment  $M_d(t)$ :

$$M(t) - M_R(t) - M_B(t) - M_{Br}(t) = 0.$$

Allerdings ist  $M_{Br}(t)$  keine Störung, sondern ein Signal, das von der Winkelgeschwindigkeit  $\omega$  des Motors abhängt und das deshalb das E/A-Verhalten des Motors verändert. In der Zusammenfassung erhält man zwei parallel geschaltete Blöcke, von denen einer das „Fehlermodell“ darstellt, das die Veränderung des Motorverhaltens durch das zusätzliche Bremsmoment beschreibt. Ob es sich dabei um einen Fehler oder eine bekannte Veränderung des Motors handelt, ist anwendungsabhängig.

In der zweiten Gleichung des Motormodells wird durch diese Rückkopplung der Einfluss von  $\omega(t)$  auf  $\dot{\omega}(t)$  verändert:



**Abb. A.32:** Gleichstrommotor mit zusätzlichem geschwindigkeitsproportionalem Bremsmoment  $M_{Br}(t)$

$$\frac{di(t)}{dt} = -\frac{R}{L}i(t) - \frac{k_M}{L}\omega(t) + \frac{1}{L}u(t)$$

$$\frac{d\omega(t)}{dt} = \frac{k_T}{J}i(t) - \left(\frac{k_R}{J} - \frac{1}{J}k_d\right)\omega(t).$$

Dies zeigt, dass sich die Motoreigenschaften verändern. Das Bremsmoment tritt nicht mehr als zusätzliche Eingangsgröße auf, sondern seine Wirkung ist durch die Änderung der Differentialgleichung für die Winkelgeschwindigkeit  $\omega(t)$  enthalten.

### Aufgabe 8.1 Einstellung eines PID-Reglers

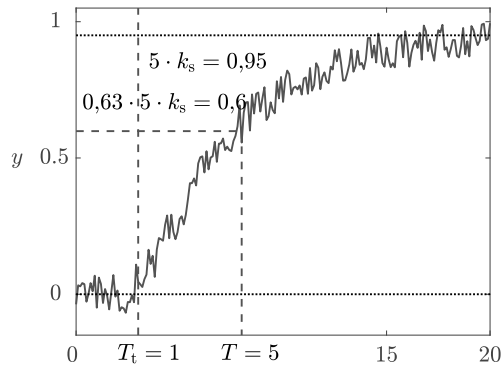
- Für die Anwendung der ersten Einstellregeln werden für die Regelstrecke die Parameter  $T_r$ ,  $T$  und  $k_s$  bestimmt. Dabei muss beachtet werden, dass die Eingangsgröße nicht zur Zeit  $t = 0$ , sondern zur Zeit  $t = 2$  verändert wurde und dass es sich dabei um eine Veränderung um 5 Einheiten handelte. Abbildung A.33 zeigt, dass aus der Messkurve folgende Parameter abgelesen wurden:

$$5 \cdot k_s = 0,95, \quad \text{also } k_s = 0,19$$

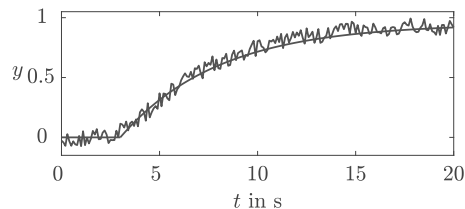
$$T_t = 1 \text{ s}$$

$$T = 5 \text{ s.}$$

Abbildung A.34 zeigt, dass das Verhalten der Regelstrecke mit diesen Parametern sehr gut nachgebildet wird. Die durchgezogene Linie zeigt die mit dem Modell berechnete Ausgangsgröße.



**Abb. A.33:** Bestimmung der Parameter der Regelstrecke



**Abb. A.34:** Vergleich der im Experiment gemessenen und der mit dem Modell berechneten Ausgangsgröße

Entsprechend der Einstellregeln erhält man damit folgende Reglerparameter:

P-Regler:  $k_P = 26,3$

PI-Regler:  $k_P = 23,7$ ,  $T_I = 3,33$

PID-Regler:  $k_P = 31,6$ ,  $T_I = 2$ ,  $T_D = 0,5$ .

Die Führungsübergangsfunktionen der Regelkreise mit diesen drei Reglern sind im linken Teil von Abb. A.35 zu sehen. Der Regelkreis mit dem PID-Regler ist nur geringfügig schneller als der PI-Regelkreis, weil der Parameter  $T_D$  in Relation zur Zeitkonstante  $T$  klein ist.

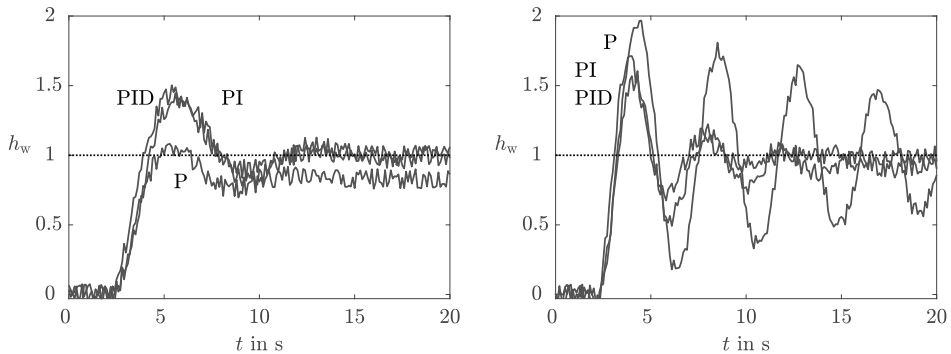
- Um die zweiten Einstellregeln anwenden zu können, muss die kritische Verstärkung bestimmt werden. Aus den Messkurven ist ersichtlich, dass  $k_{krit} = 145$  gilt. Für die entsprechende Kurve wurde die Periodendauer  $T_{krit} = 2,8$  s abgelesen. Damit erhält man folgende Reglerparameter:

P-Regler:  $k_P = 72$

PI-Regler:  $k_P = 65$ ,  $T_I = 2,38$

PID-Regler:  $k_P = 87$ ,  $T_I = 1,4$ ,  $T_D = 0,34$ .

Die Proportionalverstärkung ist wesentlich höher als bei der ersten Reglereinstellung. Deshalb schwingt der Regelkreis stärker, wie der rechte Teil von Abb. A.35 zeigt. Das Regelkreisverhalten für die reine P-Regelung ist nicht akzeptabel, aber auch für den PI- und PID-Regler erhält man ein unbefriedigendes Führungsverhalten.



**Abb. A.35:** Führungsübergangsfunktionen des Regelkreises (links: erste Reglereinstellung; rechts: zweite Reglereinstellung)

### Aufgabe 9.1 Entwurf eines Beobachters für zwei gekoppelte Vorratsbehälter

1. Das Modell wird entsprechend dem Algorithmus 3.2 auf S. 92 aufgestellt, wobei hier die vier dort angegebenen Schritte hier in abgekürzter Form angegeben werden. Für die beiden Tanks ergibt sich die Änderung der Füllhöhen aus der Summe des Zuflusses und des Abflusses:

$$\begin{aligned}\dot{h}_1(t) &= \frac{1}{A_1}(q_u(t) - q_{12}(t)) \\ \dot{h}_2(t) &= \frac{1}{A_2}(q_{12}(t) - q_y(t)).\end{aligned}$$

Es wird vereinfachend angenommen, dass die Volumenströme proportional zu den entsprechenden Füllhöhen sind:

$$\begin{aligned}q_{12}(t) &= k_{12}(h_1(t) - h_2(t)) \\ q_y(t) &= k_y h_2(t).\end{aligned}$$

Mit den Bezeichnungen

$$\begin{aligned}q_u(t) &= u(t) \\ q_y(t) &= y(t) \\ \mathbf{x}(t) &= \begin{pmatrix} h_1(t) \\ h_2(t) \end{pmatrix}\end{aligned}$$

ergibt sich das folgende Zustandsraummodell:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \begin{pmatrix} -\frac{k_{12}}{A_1} & \frac{k_{12}}{A_1} \\ \frac{k_{12}}{A_2} & -\frac{k_{12}+k_y}{A_2} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} \frac{1}{A_1} \\ 0 \end{pmatrix} u(t), \quad \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) &= (0 \quad k_y) \mathbf{x}(t).\end{aligned}$$

Setzt man die gegebenen Zahlenwerte ein, so erhält man:

$$\text{Behälter : } \begin{cases} \dot{\mathbf{x}}(t) = \begin{pmatrix} -\frac{1}{10} & \frac{1}{10} \\ \frac{1}{12} & -\frac{1}{8} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} \frac{1}{10} \\ 0 \end{pmatrix} u(t), & \mathbf{x}(0) = \mathbf{x}_0 \\ y(t) = (0 \quad 0,5) \mathbf{x}(t). \end{cases}$$

2. Eine Anwendung des Beobachters ist nur sinnvoll, wenn das System vollständig beobachtbar ist. Dafür muss der Rang der Beobachtbarkeitsmatrix  $\mathbf{S}_B$  gleich zwei sein:

$$\text{Rang}(\mathbf{S}_B) = \text{Rang} \begin{pmatrix} \mathbf{c}^T \\ \mathbf{c}^T \mathbf{A} \end{pmatrix} = \text{Rang} \begin{pmatrix} 0 & 0,5 \\ \frac{1}{24} & -\frac{1}{16} \end{pmatrix} = 2.$$

Diese Bedingung ist hier erfüllt.

3. Um zu wissen, wohin man die Eigenwerte des Beobachters legt, müssen zuerst die Eigenwerte des Systems berechnet werden.

$$\det(\lambda \mathbf{I} - \mathbf{A}) = \left( \lambda + \frac{1}{10} \right) \left( \lambda + \frac{1}{8} \right) - \frac{1}{120}.$$

Dabei erhält man

$$\lambda_1 \approx -0,02, \quad \lambda_2 \approx -0,2.$$

Die Eigenwerte des Beobachters sollen betragsmäßig größer als die dominierenden Systemeigenwerte sein, was z. B. für  $\lambda_{B1} = -0,3$  und  $\lambda_{B2} = -0,4$  erfüllt ist. Dann gilt für das charakteristische Polynom des Beobachters die Beziehung

$$(\lambda + 0,3)(\lambda + 0,4) = \lambda^2 + 0,7\lambda + 0,12 = \lambda^2 + a_{B1}\lambda + a_{B0}.$$

Die Rückführmatrix berechnet sich aus

$$\mathbf{l} = (\mathbf{s}_B, \mathbf{A}\mathbf{s}_B, \mathbf{A}^2\mathbf{s}_B) \begin{pmatrix} a_{B0} \\ a_{B1} \\ 1 \end{pmatrix}$$

mit

$$\mathbf{s}_B = \mathbf{S}_B^{-1} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 24 \\ 0 \end{pmatrix}$$

und führt auf

$$\mathbf{l} = \begin{pmatrix} 1,64 \\ 0,95 \end{pmatrix}.$$

### Aufgabe 10.2 Fehlerdiagnose eines Reaktors mit Füllstandsregelung

1. Das zu diagnostizierende System besteht aus dem in Abb. A.36 gezeigten Regelkreis, dessen Modell durch kompositionale Modellbildung (Abschn. 3.6) dadurch bestimmt wird, dass zunächst die Modelle der Regelstrecke und des Reglers aufgestellt und diese dann zum Modell des Regelkreises kombiniert werden.

Das Modell des Behälters enthält die Differentialgleichung für den Füllstand

$$\dot{h}(t) = \frac{1}{A}(q_u(t) - q(t) - q_f(t)), \quad h(0) = h_0$$

mit den Volumenströmen

$$\begin{aligned} q(t) &= kh(t) \\ q_f(t) &= k_f h(t), \end{aligned}$$

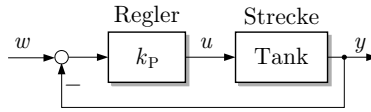


Abb. A.36: Blockschaltbild des Regelkreises

wobei  $k$  den Proportionalitätsfaktor zwischen der Füllhöhe und dem Auslaufvolumenstrom darstellt. Im Fehlerfall gilt  $k_f = k$  und im fehlerfreien Fall  $k_f = 0$ .

Es wird ein P-Regler

$$q_u(t) = k_P(w - h(t))$$

mit der Reglerverstärkung  $k_P$  eingesetzt. Die Beschreibungen der Strecke und des Reglers ergeben gemeinsam das folgende Regelkreismodell mit dem Zustand  $x(t) = h(t)$ :

$$\bar{\Sigma} : \begin{cases} \dot{x}(t) = -\frac{k_P + k + k_f}{A}x(t) + \frac{k_P}{A}w, & x(0) = h_0 \\ y(t) = kx(t). \end{cases}$$

Da  $w = \text{konst.}$  angenommen wird, ist das Modell affin und somit nichtlinear. Die Linearisierung eines solchen Systems wurde in Aufgabe 3.7 behandelt. Das um den Arbeitspunkt

$$\bar{x} = \frac{k_P}{k_P + k + k_f}w \quad (\text{A.33})$$

linearisierte Modell lautet

$$\bar{\Sigma}_{\text{lin}} : \begin{cases} \dot{x}(t) = -\frac{k_P + k + k_f}{A}x(t), & x(0) = h_0 - \bar{x} \\ y(t) = kx(t), \end{cases} \quad (\text{A.34})$$

wobei die Abweichungen vom Arbeitspunkt wieder mit den üblichen Signalname  $x$ ,  $u$  und  $y$  bezeichnet werden. Die Modelle für den Fehlerfall und den fehlerfreien Fall unterscheiden sich in der Zeitkonstante

$$T = \frac{A}{k_P + k + k_f} \quad (\text{A.35})$$

dieses Systems, die im Fehlerfall durch den zusätzlichen Auslauf kleiner ist. Im Fehlerfall wird der Beobachter deshalb ein nicht verschwindendes Residuum liefern.

2. Das Blockschaltbild ist in Abb. A.37 dargestellt. Wie aus dem linearisierten Modell hervorgeht, hat das betrachtete System keine Eingangsgröße. Der Sollwert  $w$  ist konstant und geht als Parameter in den Arbeitspunkt des linearisierten Modells ein.
3. Der Beobachter wird für das System (A.34) mit  $k_f = 0$  entworfen. Das System ist beobachtbar, weil die Ausgangsgröße proportional zum (skalaren) Zustand  $x(t)$  ist. Der Fehler ist folglich diagnostizierbar.

Aus dem Modell (A.34) des geregelten Reaktors erhält man die folgende Beobachtermatrix:

$$\dot{\hat{x}}(t) = -\left(\frac{k_P + k}{A} + lk\right)\hat{x}(t) + ly(t).$$

Der Beobachter hat die Zeitkonstante

$$T_B = \frac{A}{k_P + k + Alk},$$



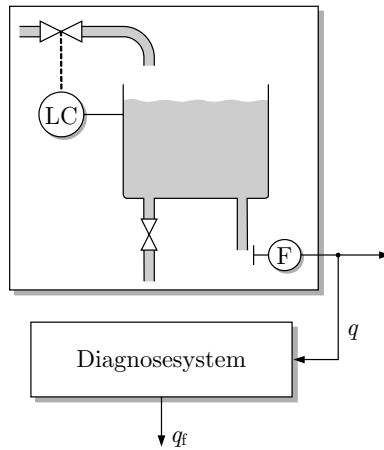


Abb. A.37: Diagnose des Behältersystems

die halb so groß sein soll wie die Zeitkonstante (A.35) mit  $k_f = 0$ . Dafür erhält man die Beobachter-rückführverstärkung

$$l = \frac{k_P + k}{kA}$$

und die Beobachtergleichung

$$\Sigma_B : \dot{\hat{x}}(t) = -2\frac{k_P + k}{A}\hat{x}(t) + \frac{k_P + k}{kA}y(t).$$

#### 4. Um das Residuum

$$r(t) = y(t) - k\hat{x}(t)$$

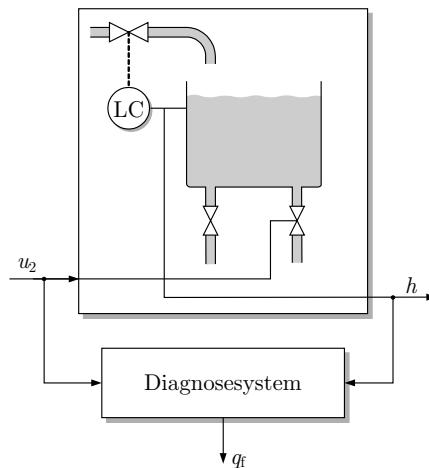
zu berechnen, muss man das Modell (A.34) des fehlerbehafteten Systems mit dem Beobachter kombinieren, woraus man

$$\begin{pmatrix} \dot{x}(t) \\ \dot{r}(t) \end{pmatrix} = \begin{pmatrix} -\frac{k_P + k + k_f}{A} & 0 \\ -\frac{kk_f}{A} & -\frac{2k_P + k}{A} \end{pmatrix} \begin{pmatrix} x(t) \\ r(t) \end{pmatrix}, \quad \begin{pmatrix} x(0) \\ r(0) \end{pmatrix} = \begin{pmatrix} h_0 \\ 0 \end{pmatrix}$$

erhält. Für den fehlerfreien Zustand gilt  $k_f = 0$  und folglich  $r(t) = 0$  für alle  $t \geq 0$ . Je größer  $k_f$  ist, umso stärker beeinflusst  $x(t)$  die Differentialgleichung für  $r(t)$ .

5. In der veränderten Aufgabenstellung tritt mit  $u_2(t)$  eine von außen auf das System einwirkende Eingangsgröße auf, die das Diagnosesystem verarbeiten muss. An Stelle des Ausflusses wird jetzt der Füllstand  $h(t)$  als Ausgangsgröße gemessen. Die neue Struktur ist in Abb. A.38 zu sehen. Im Modell wird jetzt  $q(t) = k_2 u_2(t)$  gesetzt, wodurch sich die Gleichungen für das linearisierte Modell und den Beobachter verändern.
6. Der Fehler wirkt sich in der ersten Anordnung auf den statischen Wert des Füllstands aus, denn der Arbeitspunkt (A.33) des affinen Systems und folglich der zugehörige Ausfluss

$$\bar{y} = k \frac{k_P}{k_P + k + k_f} w$$



**Abb. A.38:** Veränderte Kopplung von Reaktor und Diagnosesystem

sind von  $k_f$  abhängig. Diese Gleichung zeigt, dass der Ausfluss statisch umso kleiner ist, je größer der Fehler  $k_f$  ist, was für die Diagnose genutzt werden kann, wenn die Änderung deutlich größer als der auftretende Messfehler ist.

In der zweiten Anordnung kann man den Fehler nur dann aus dem statischen Wert des Füllstands erkennen, wenn die Eingangsgröße  $u_2$  konstant ist, so dass sich ein statischer Endwert einstellt. Die beobachtergestützte Diagnose hat den Vorteil, dass sie den Fehler auch bei einer ständigen Anregung des Systems erkennen kann.

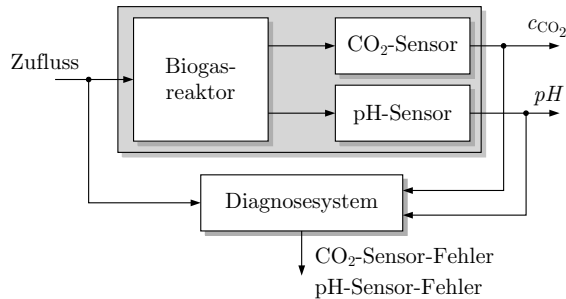
Die Fehlererkennung aus den statischen Werten ist nicht mehr möglich, wenn an Stelle des P-Reglers ein PI-Regler eingesetzt wird, weil dann im Regelkreis keine bleibende Regelabweichung für den Füllstand mehr auftritt. Dann ist die Fehlerdiagnose in beiden Anordnungen nur mit dem beobachtergestützten Verfahren möglich.

#### Aufgabe 10.4 Intelligenter pH-Sensor

Der wichtigste Unterschied zwischen einem intelligenten Sensor und einem Sensorüberwachungssystem besteht in den bei der Diagnose verarbeiteten Informationen. Das Diagnosesystem des in Abb. 10.22 gezeigten intelligenten Sensors kann nur interne Signale des Sensors auswerten, denn die „Intelligenz“ ist im Sensor selbst eingebaut. Ein solches Diagnosesystem kann überwachen, ob der Sensor mit der richtigen Betriebsspannung versorgt wird, ob die Elektronik des Sensors funktionstüchtig ist und ob das vom Sensor erzeugte Messsignal einen Wert innerhalb eines vorgegebenen Wertebereichs hat. Das Diagnosesystem merkt also, wenn eine elektrische Versorgungsleitung gestört ist oder wenn auf Grund interner Fehler ein Messwert erzeugt wird, der physikalisch unsinnig ist.

Das Diagnosesystem kann jedoch keine Fehler erkennen, die den Zusammenhang zwischen dem aktuellen physikalischen Wert der zu messenden Größe, der in Abb. 10.22 als pH-Wert bezeichnet ist, und dem aktuellen Wert des Messsignals betreffen. Dieser Zusammenhang ist bei pH-Sensoren beispielsweise durch eine falsche Kalibrierung bzw. durch die bei diesen Sensoren übliche Drift oder durch ein Zuwachsen des Sensors durch Bioorganismen verfälscht. Da das Diagnosesystem keine redundanten Informationen über den wahren Wert besitzt, kann es Fehler bei der Umwandlung des physikalischen Wertes in

das Messsignal nicht feststellen. Das Diagnosesystem des pH-Sensors kann also, bei aller Intelligenz, auf Grund fehlender Informationen über den wahren physikalischen Wert gar nicht in der Lage sein, beispielsweise einen Kalibrierungsfehler des pH-Sensors anzuzeigen.



**Abb. A.39:** Überwachung zweier Sensoren

Die Überwachung derartiger Fehler ist nur mit einem Diagnosesystem möglich, das redundante Informationen über den physikalischen Messwert besitzt, also den zu überwachenden Sensor im Zusammenspiel mit dem technischen System, an dem die Messgröße entsteht, betrachtet. Das im Beispiel 10.3 behandelte Sensorüberwachungssystem verfügt über redundante Informationen, weil es Eingangsgrößen des Biogasreaktors erhält und außer dem pH-Wert auch die  $\text{CO}_2$ -Konzentration kennt. Diese Größen ermöglichen es zusammen mit dem dynamischen Modell des Reaktors, den wahren pH-Wert zu bestimmen und mit dem Messwert zu vergleichen. Die Diagnose beruht, im Prinzip, auf einer Rekonstruktion der vom pH-Sensor zu bestimmenden physikalischen Größe aus anderen Informationen als denen, die der Sensor verwendet bzw. erzeugt.

Bei dieser Sensorüberwachung wurde also mit dem Blockschaltbild nach Abb. A.39 gearbeitet. Die von dem Überwachungssystem ausgegebenen Fehler betreffen den Zusammenhang zwischen den physikalischen Werten und den Messsignalwerten.

Beide Arten von Sensorüberwachungssystemen ergänzen sich bezüglich ihres Funktionsumfanges. Intelligente Sensoren nehmen dem Anwender die Diagnose der prinzipiellen Funktionsweise des Sensors ab. Um die Richtigkeit des Messwertes feststellen zu können, müssen jedoch Sensorüberwachungssysteme mit der im Abschn. 10.3 gezeigten Informationsstruktur eingesetzt werden.

### Aufgabe 10.5 Diagnostizierbarkeit eines Servoventils

1. Da im Beispiel 10.4 gezeigt wurde, dass mit einem beobachtergestützten Diagnosesystem die beiden Fehler  $f_1$  und  $f_2$  voneinander unterschieden werden können, muss das bei der Diagnose verwendete Modell des Servoventils zusammen mit den beiden Fehlermodellen vollständig beobachtbar sein. Weil außerdem die strukturelle Beobachtbarkeit eine notwendige Bedingung für die (numerische) Beobachtbarkeit ist, muss das erweiterte Ventilmodell auch strukturell beobachtbar sein. Dieses anhand des Beispiels 10.4 zu vermutende Ergebnis einer strukturellen Beobachtbarkeitsanalyse erhält man folgendermaßen. Vom Modell (10.29), (10.30) sind die Systemmatrix

$$\tilde{\mathbf{A}} = \begin{pmatrix} \mathbf{A}_V & \mathbf{e}_{V1} & \mathbf{e}_{V2} \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & -\frac{1}{T_{F2}} \end{pmatrix} = \begin{pmatrix} -\frac{1}{T_D} & \frac{k_{PV}}{T_D} & 0 & 0 & \frac{1}{T_D} \\ 0 & 0 & 1 & 0 & 0 \\ -\frac{k_{NA_M}}{m} & -\frac{k_F}{m} & -\frac{k_R}{m} & \frac{k_{NA_M}}{m} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{T_{F2}} \end{pmatrix}$$

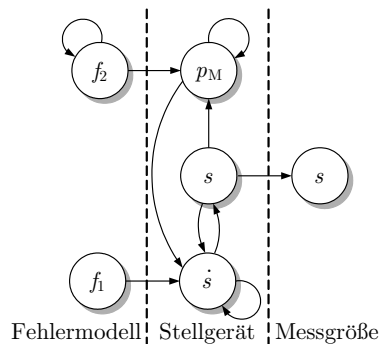
und der Ausgangsvektor

$$\tilde{\mathbf{c}}^T = (\mathbf{c}_V^T \ 0 \ 0) = (0 \ 1 \ 0 \ 0 \ 0)$$

maßgebend. Daraus erhält man die folgenden Strukturmatrizen

$$[\tilde{\mathbf{A}}] = \begin{pmatrix} * & * & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 \\ * & * & * & * & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * \end{pmatrix}, \quad [\tilde{\mathbf{c}}^T] = (0 \ * \ 0 \ 0 \ 0),$$

die auf den in Abb. A.40 gezeigten Strukturgraphen führen. Die Eingangsgröße  $u$  des Stellgerätes tritt im Strukturgraphen nicht auf, weil nur eine Beobachtbarkeitsanalyse durchgeführt wird. Der rechte  $s$ -Knoten ist die Messgröße.



**Abb. A.40:** Strukturgraph des Stellgerätes für die strukturelle Beobachtbarkeitsanalyse

Das System ist ausgangsverbunden, denn von allen Zustandsknoten führt ein Pfad zum Ausgangsknoten  $s$ , so dass die erste Bedingung für strukturelle Beobachtbarkeit des Satzes 5.3 auf S. 172 erfüllt ist. Es ist als zweites die Rangbedingung zu überprüfen, die sich auf die Matrix

$$\begin{pmatrix} [\tilde{\mathbf{A}}] \\ [\tilde{\mathbf{c}}^T] \end{pmatrix} = \begin{pmatrix} * & * & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 \\ * & * & * & * & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * \\ 0 & * & 0 & 0 & 0 \end{pmatrix}$$

bezieht. Der  $s$ -Rang dieser Matrix ist gleich fünf, denn in der Matrix können fünf  $*$ -Elemente gefunden werden, die in verschiedenen Zeilen und Spalten stehen:

$$\begin{pmatrix} \bullet & * & 0 & 0 & * \\ 0 & 0 & \bullet & 0 & 0 \\ * & * & * & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet \\ 0 & \bullet & 0 & 0 & 0 \end{pmatrix}.$$

Das Paar  $(\tilde{\mathbf{A}}, \tilde{\mathbf{c}}^T)$  ist also strukturell beobachtbar, was in der Anwendung auf die Fehlerdiagnose bedeutet, dass die Fehler  $f_1$  und  $f_2$  (strukturell) diagnostizierbar sind. Dies ist eine notwendige Bedingung für die (numerische) Diagnostizierbarkeit, die bei dem Stellgerät gegeben ist, wie das Beispiel 10.4 zeigt.

2. Wenn beide Fehler konstant sind, haben sie dasselbe Fehlermodell

$$\dot{f}_i = 0.$$

Damit erscheint in der Matrix  $\tilde{\mathbf{A}}$  in der unteren Zeile eine Null, die dann auch in der betreffenden Strukturmatrix steht:

$$\tilde{\mathbf{A}} = \begin{pmatrix} -\frac{1}{T_D} & \frac{k_{PV}}{T_D} & 0 & 0 & \frac{1}{T_D} \\ 0 & 0 & 1 & 0 & 0 \\ -\frac{k_N A_M}{m} & -\frac{k_F}{m} & -\frac{k_R}{m} & \frac{k_N A_M}{m} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{0} \end{pmatrix}, \quad [\tilde{\mathbf{A}}] = \begin{pmatrix} * & * & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 \\ * & * & * & * & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{0} \end{pmatrix}.$$

Damit entfällt im Strukturgraphen die Schleife am Knoten  $f_2$ , die nichts an der Ausgangsverbundenheit des Systems ändert. Bei der Untersuchung des strukturellen Rangs stellt man jedoch fest, dass die Änderung des Fehlermodells einen Rangabfall erzeugt, denn die Matrix

$$\begin{pmatrix} [\tilde{\mathbf{A}}] \\ [\tilde{\mathbf{c}}^T] \end{pmatrix} = \begin{pmatrix} * & * & 0 & 0 & * \\ 0 & 0 & * & 0 & 0 \\ * & * & * & * & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{0} \\ 0 & * & 0 & 0 & 0 \end{pmatrix}$$

hat nur noch vier \*-Elemente in unterschiedlichen Zeilen und Spalten. Das System ist nicht strukturell beobachtbar, die Fehler also nicht identifizierbar.

Das Beispiel zeigt, dass die Fehler  $f_1$  und  $f_2$  nur auf Grund ihrer unterschiedlichen dynamischen Eigenschaften auseinandergehalten werden können. Dies ist nicht verwunderlich, denn sie führen beide zu einem Druckabfall oberhalb der Membran und deshalb zu einem sehr ähnlichen Fehlverhalten des Stellgerätes. Da sich der Fehler  $f_1$  jedoch in einer bleibenden Druckänderung und der Fehler  $f_2$  in einer abklingenden Druckänderung äußert, kann der Diagnosealgorithmus beide Fehler identifizieren. Die Informationen über die Fehlerdynamik sind also für die Diagnose wesentlich.

Diese Tatsache findet man bei einer genauen Betrachtung von Abb. 10.36 auf S. 327 bestätigt. Dort sieht man, dass der Beobachter im Moment des Fehlereintritts in beiden Zuständen  $\hat{f}_1$  und  $\hat{f}_2$  große Ausschläge zeigt, also zunächst nur erkennt, dass ein Fehler eingetreten ist. Der Beobachter schwingt dann schnell ein, wobei  $\hat{f}_1 = 0$  gilt und  $\hat{f}_2$  dem wahren Verlauf von  $f_2$  folgt. Durch die unterschiedliche Dynamik kann zwischen beiden Fehlern unterschieden, die Fehler also identifiziert werden.

Man kann dieses Ergebnis übrigens dahingehend erweitern, dass man bei einem System, das nur eine Messgröße hat, höchstens einen konstanten Fehler identifizieren kann, weil bei mehr als einem Fehler die Rangbedingung stets verletzt ist.

**Aufgabe 11.1**    *Modell eines Getränkeautomaten*

Zwei Wahlmöglichkeiten für das Getränk, der Geldeinwurf und das Drücken der Rückgabetaste des Geldes sind die vier Eingaben des Getränkeautomaten, die den Werten 1 bis 4 der Eingangsgröße  $v$  folgendermaßen zugeordnet werden:

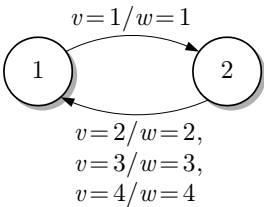
$v$	Bedeutung
1	Geld einwerfen
2	Kaffee wählen
3	Kakao wählen
4	Geldrückgabeknopf drücken

Darauf „antwortet“ der Getränkeautomat mit folgenden Ausgaben  $w$ :

$w$	Bedeutung
1	Wahltasten für Kaffee und Kakao blinken
2	Zubereitung von Kaffee
3	Zubereitung von Kakao
4	Geldrückgabe

Um die Zustände  $z$  des Getränkeautomaten zu definieren, muss man sich vorstellen, welche Eingaben die Getränkevorbereitung voranbringen, welche den Getränkeautomaten in den Ruhezustand zurücksetzen und bei welchen Eingaben nichts passiert. Dabei erhält man folgende Zustandsdefinitionen:

$z$	Bedeutung
1	Ruhezustand (Warten auf den Geldeinwurf)
2	ausreichende Menge Geld erhalten (Warten auf Getränkewahl)



**Abb. A.41:** Automatengraph des Getränkeautomaten

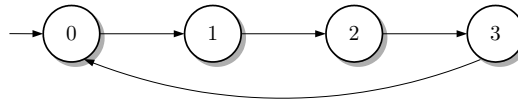
Ein Zustandsübergang  $1 \rightarrow 2$  erfolgt, wenn ausreichend Geld eingeworfen wurde ( $v = 1$ ). Ein Zustandsübergang  $2 \rightarrow 1$  tritt nach den Eingaben  $v = 2, 3$  oder  $4$  auf. Dabei wird die Reaktion des Getränkeautomaten durch  $w = 2, 3$  oder  $4$  beschrieben. Der Automatengraph ist in Abb. A.41 zu sehen. Die Zustandsübergangsfunktion kann auch durch die folgende Automatentabelle beschrieben werden:

$z'$	$w$	$z$	$v$
2	1	1	1
1	2	2	2
1	3	2	3
1	4	2	4

**Aufgabe 11.2** *Modell eines Zählers*

Als Beispiel wird hier ein Modulo-4-Zähler angegeben. Dieser Zähler zählt von null bis drei und startet dann wieder bei null. Der Automatengraph des Zählers ist in Abbildung A.42 dargestellt. Der analytische Ausdruck für die Zustandsübergangsfunktion lautet

$$z(k+1) = (z(k) + 1) \bmod 4.$$



**Abb. A.42:** Darstellung eines Modulo-4-Zählers als Automatengraph

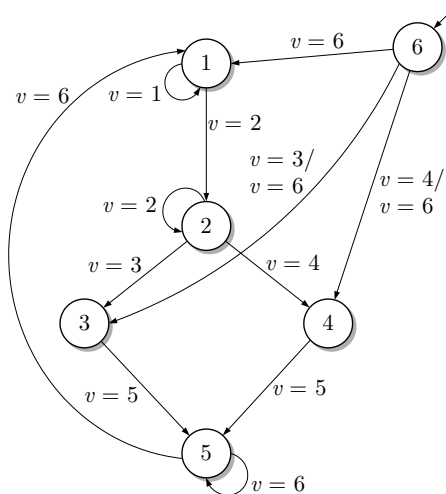
**Aufgabe 11.3** *Ereignisdiskrete Beschreibung des Autofahrens*

Um die Darstellung der Lösung etwas zu vereinfachen, wird die Reaktion des Fahrers auf den Straßenverkehr vernachlässigt und nur die Reaktion auf den Straßenverlauf beschrieben. Der Automatengraph des Fahrvorgangs ist in Abb. A.43 dargestellt, wobei die in der Tabelle aufgeführten Abkürzungen verwendet werden.

$z$	Zustand
1	Straße führt geradeaus, keine Kurve in Sicht
2	Kurve in Sicht
3	Rechtskurve
4	Linkskurve
5	Kurve zu Ende
6	Starten

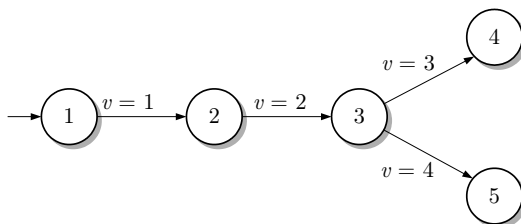
$v$	Eingang
1	Geschwindigkeit beibehalten, nicht lenken
2	Geschwindigkeit verringern
3	nach rechts einlenken
4	nach links einlenken
5	Lenkrad auf geradeaus stellen, beschleunigen
6	beschleunigen

Der Fahrvorgang beginnt im Zustand  $z = 6$ . Der Automat kehrt in diesen Zustand während der Fahrt nicht wieder zurück. In Abhängigkeit davon, ob das Auto in einer Kurve oder einer Geraden startet, wechselt der Automat in die Zustände  $z = 1$ ,  $z = 3$  oder  $z = 4$ . Befindet sich der Automat im Zustand  $z = 1$ , führt die Straße also geradeaus, kann die Geschwindigkeit beibehalten werden. Nähert sich das Fahrzeug einer Kurve ( $z = 2$ ), muss die Geschwindigkeit reduziert werden. Beginnend mit dem Eintritt in die Kurve muss das Lenkrad eingeschlagen werden. Am Ende der Kurve kann wieder beschleunigt und geradeaus gelenkt werden.



**Abb. A.43:** Automatengraph des Fahrverhaltens abhängig von der Straßenlage

Dieses Modell beschreibt das Autofahren in den sich prinzipiell voneinander unterscheidenden Stufen. Es sagt allerdings nichts darüber aus, wie das Fahrzeug beschleunigt oder abgebremst bzw. wie das Lenkrad eingeschlagen wird.



**Abb. A.44:** Automatengraph eines Bestellvorgangs

#### Aufgabe 11.4 Beschreibung eines Bestellvorgangs

Es soll folgender Prozess beschrieben werden:

1. Ein Buch wird von der Buchhandlung beim Großhändler bestellt.
2. Das Buch wird angeliefert.
3. Das Buch wird in das Regal einsortiert.
4. Das Buch wird verkauft.
5. Das Buch wird ausgemustert.



Dieser Ablauf ist in Abb. A.44 als Automatengraph dargestellt, wobei die Eingaben  $v$  die Handlungen der Buchhändler und die Zustände  $z$  den Standort des Buches beschreiben:

$z$	Bedeutung
1	Das Buch liegt beim Großhändler.
2	Das Buch liegt im Lager der Buchhandlung.
3	Das Buch liegt im Regal.
4	Das Buch ist verkauft und befindet sich beim Kunden.
5	Das Buch ist ausgemustert.

$v$	Bedeutung
1	Das Buch wird bestellt und geliefert.
2	Das Buch wird in das Regal einsortiert.
3	Ein Kunde kauft das Buch.
4	Das Buch wird beschädigt.

### Aufgabe 11.7 Modellierung einer Fertigungszelle

Die Zustandsmenge der Fertigungszelle ist durch die Kombination der in der Aufgabenstellung gegebenen Zustände des Greifers und der zwei Maschinen gegeben, die im Vektor

$$z = \begin{pmatrix} z_G \\ z_{M_1} \\ z_{M_2} \end{pmatrix}$$

zusammengefasst werden. Es gibt  $3 \cdot 3 \cdot 2 = 18$  verschiedene Zustände. Allerdings wird sich zeigen, dass die Fertigungszelle nicht alle diese Zustände annehmen kann.

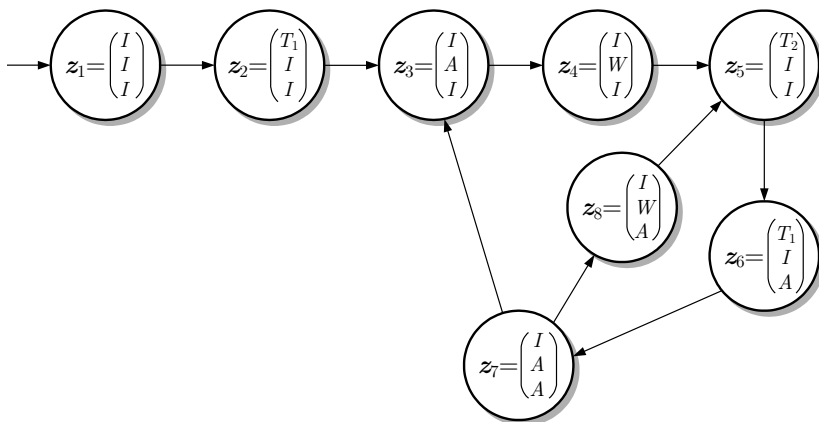


Abb. A.45: Modell der Fertigungszelle

Die Regeln, nach denen die Steuerungen der drei Komponenten arbeiten, definieren die Zustandsübergänge. Mit ihnen erhält man den in Abb. A.45 gezeigten Automatengraphen, in dem die Zustandsübergänge folgende Bedeutung haben:

Zustandsübergang	Bedeutung
$z_1 \rightarrow z_2$	Der Greifer reagiert auf den Ruhezustand der Maschine $M_1$ .
$z_2 \rightarrow z_3$	Der Transport eines Teiles zur Maschine $M_1$ ist abgeschlossen.
$z_3 \rightarrow z_4$	Die Maschine $M_1$ hat die Bearbeitung beendet.
$z_4 \rightarrow z_5$	Der Greifer reagiert auf die Beendigung der Bearbeitung durch die Maschine $M_1$ .
$z_5 \rightarrow z_6$	Der Transport eines Teiles von $M_1$ zu $M_2$ ist abgeschlossen.
$z_6 \rightarrow z_7$	Der Transport eines Teiles zur Maschine $M_1$ ist abgeschlossen.
$z_7 \rightarrow z_8$	Die Bearbeitung eines Teiles durch die Maschine $M_1$ ist abgeschlossen.
$z_7 \rightarrow z_3$	Die Bearbeitung eines Teiles durch die Maschine $M_2$ ist abgeschlossen.
$z_8 \rightarrow z_5$	Die Bearbeitung eines Teiles durch die Maschine $M_2$ ist abgeschlossen.

Im Automatengraphen kommen nur acht der 18 möglichen Zustände vor. Das hängt z. B. damit zusammen, dass der Greifer nur dann ein Teil zur Maschine  $M_2$  transportieren kann (Greiferzustand  $T_2$ ), wenn sich  $M_2$  im Zustand  $I$  befindet.

Der Automat ist nichtdeterministisch, weil es von den Bearbeitungszeiten der beiden Maschinen abhängt, ob die Fertigungszelle vom Zustand  $z_7$  in den Nachfolgezustand  $z_3$  oder  $z_8$  übergeht. Dieser Nichtdeterminismus hat nichts damit zu tun, dass die Steuerung in jeder Situation stets „weiß“, was sie als nächstes tun soll.

### Aufgabe 11.8 Funktion einer Raumbeleuchtung

Der Zustand des Systems ist durch die Zustände der Beleuchtung („außer Betrieb“, „betriebsbereit“), des Schalters („eingeschaltet“, „ausgeschaltet“) und der Lampe („funktionsfähig“, „defekt“) beschrieben, wobei die folgenden Kombinationen für das Verhalten wichtig sind:

$z$	Sicherung	Schalter	Lampe
1	herausgeschraubt	ausgeschaltet	funktionsfähig
2	betriebsbereit	ausgeschaltet	funktionsfähig
3	betriebsbereit	eingeschaltet	funktionsfähig
4	*	*	defekt
5	herausgeschraubt	eingeschaltet	funktionsfähig

Der Stern zeigt, dass bestimmte Werte für die Definition dieses Systemzustands unbedeutend sind. Als Eingaben kommen folgende Aktionen in Frage

$v$	Bedeutung
1	Die Sicherung wird hereingeschraubt.
2	Die Sicherung wird herausgeschraubt.
3	Der Schalter wird eingeschaltet.
4	Der Schalter wird ausgeschaltet.

Die Ausgabe kann folgende Werte annehmen:

$w$	Bedeutung
1	Die Lampe brennt.
0	Die Lampe brennt nicht.

Die Raumbeleuchtung verhält sich nichtdeterministisch, weil die Lampe durchbrennen kann oder nicht. Der zweite Fall kann entsprechend der Aufgabenstellung nur beim Schließen des Stromkreises (Sicherung wird hereingeschraubt oder der Schalter wird geschlossen) passieren.

Der Automat wird durch die Gl. (11.36) mit

$G(z, v)$	$z = 1$	$z = 2$	$z = 3$	$z = 4$	$z = 5$
$v = 1$	$\{2\}$	–	–	$\{4\}$	$\{3, 4\}$
$v = 2$	–	$\{1\}$	$\{5\}$	$\{4\}$	–
$v = 3$	$\{5\}$	$\{3, 4\}$	–	$\{4\}$	–
$v = 4$	–	–	$\{2\}$	$\{4\}$	$\{1\}$

$$H(z) = \begin{cases} 1 & \text{für } z = 3 \\ 0 & \text{für } z \neq 3 \end{cases}$$

beschrieben. Die Striche in der Tabelle weisen darauf hin, dass  $G$  für die betreffenden Argumentwerte nicht definiert ist, weil die betreffende Eingangsgröße bei dem betrachteten Zustand nicht angewendet werden kann. Der Wert der Funktion  $G$  ist eine Menge, die bei nichtdeterministischen Zustandsübergängen mehr als ein Element umfasst. Der Automat ist ein Moore-Automat, dessen Ausgabe nur vom Zustand abhängt.

Auf die Form (11.38) kommt man nach Einführung des Vektors  $\mathbf{p}(k) = (p_1(k) \dots p_5(k))^T$ , indem man die o. a. Tabelle in die Matrix  $\mathbf{G}(v)$  überführt:

$$\begin{aligned} \mathbf{G}(v=1) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, & \mathbf{G}(v=2) &= \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}, \\ \mathbf{G}(v=3) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}, & \mathbf{G}(v=4) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

Der Automat erzeugt für die Eingangsfolge

$$V(0 \dots 5) = (1, 3, 4, 3, 4, 2) \quad (\text{A.36})$$

und den Anfangszustand  $z(0) = 1$  die durch

$$Z(0 \dots 6) = \left( \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \right)$$

beschriebenen Trajektorien. Die Menge dieser Trajektorien ist

$$\mathcal{B} = \{(1, 2, 3, 2, 3, 2, 1), (1, 2, 3, 2, 4, 4, 4), (1, 2, 4, 4, 4, 4, 4)\}.$$

Die erste Trajektorie beschreibt das Verhalten der Raumbeleuchtung im normalen Zustand, wobei folgende Ausgangsfolge entsteht

$$W(0\dots 6) = (0, 0, 1, 0, 1, 0, 0).$$

Die zweite Zustandsfolge stellt das Verhalten der Raumbeleuchtung für den Fall dar, dass beim zweiten Einschalten die Lampe durchbrennt. Die dritte Trajektorie trifft zu, wenn beim ersten Einschalten die Glühlampe durchbrennt.

Wenn man das Modell für den Fall erweitern will, dass der Schalter klemmt, so kann der Automat durch Ausschalten des Schalters den neu einzuführenden Fehlerzustand („Schalter klemmt“) wieder verlassen, was beim Durchbrennen der Lampe (Zustand  $z = 4$ ) nicht möglich ist.

### Aufgabe 11.11 Erweiterung des Modells einer Raumbeleuchtung

Eine Erweiterung auf einen stochastischen Automaten ist möglich, wenn bekannt ist, mit welchen Wahrscheinlichkeiten die Lampe beim Schließen des Stromkreises durchbrennt und der Schalter beim Einschalten klemmt. Dann stehen an Stelle der Einsen diese Wahrscheinlichkeiten in der Matrix  $G$ . Für den Fall, dass nur die Lampe durchbrennen kann, entstehen folgende Matrizen

$$\begin{aligned} G(v=1) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,99 \\ 0 & 0 & 0 & 1 & 0,01 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, & G(v=2) &= \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \\ G(v=3) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0,99 & 0 & 0 & 0 \\ 0 & 0,01 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}, & G(v=4) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \end{aligned}$$

wenn die Wahrscheinlichkeit für das Durchbrennen der Lampe 0,01% beträgt. Die Wahrscheinlichkeitsangaben ersetzen die zwei in einer gemeinsamen Spalte stehenden Einsen in den Matrizen von Aufg. 11.8. In einigen Spalten der angegebenen Matrizen ist die Eigenschaft (11.43) nicht erfüllt, weil die betreffende Eingangsgröße  $v$  für den Zustand, der zu dieser Spalte gehört, nicht zugelassen ist.

Zur Berechnung der Zustandsfolge muss man noch wissen, mit welcher Wahrscheinlichkeit die Lampe von vornherein defekt ist, was hier mit 0,02% angenommen wird. Für die in Aufgabe 11.8 verwendete Eingangsfolge (A.36) erhält man mit dem stochastischen Automaten

$$Z(0\dots 6) = \left( \begin{pmatrix} 0,98 \\ 0 \\ 0 \\ 0,02 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0,98 \\ 0 \\ 0,02 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0,97 \\ 0,03 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0,97 \\ 0 \\ 0,03 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0,96 \\ 0,04 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0,96 \\ 0 \\ 0,04 \\ 0 \end{pmatrix}, \begin{pmatrix} 0,96 \\ 0 \\ 0 \\ 0,04 \\ 0 \end{pmatrix} \right).$$

Für jeden Zeitpunkt  $k$  beschreibt der zugehörige Vektor  $p(k)$ , mit welchen Wahrscheinlichkeiten die einzelnen Zustände angenommen werden. Die Lampe brennt nach dem zweiten Einschalten mit der Wahrscheinlichkeit

$$\text{Prob}(z(4)=3) = p_3(4) = 0,96.$$

Nachdem die Sicherung herausgeschraubt wurde, ist die Beleuchtung außer Betrieb, und zwar mit der Wahrscheinlichkeit

$$\text{Prob}(z(6)=4) = p_4(6) = 0,04$$

mit einer defekten Lampe. Als Ausgangsfolge erhält man die folgenden Wahrscheinlichkeiten für das Brennen der Lampe

$$W = (0, 0, 0,97, 0, 0,96, 0, 0).$$

Bei der Erweiterung der Aufgabe um einen Fehler im Schalter muss ein neuer Automatenzustand für den defekten Schalter eingeführt und die nichtdeterministischen Zustandsübergänge mit den entsprechenden Zustandsübergangswahrscheinlichkeiten beschrieben werden.

**Aufgabe 11.12** *Beschreibung eines Regensensors durch einen stochastischen Automaten*

Die Angabe

$$\text{Prob}(Z(k+1) = 2 \mid Z(k) = 1, V(k) = 2) = 0,95$$

besagt, dass der Regensensor, der bisher trockenes Wetter angezeigt hat, den Regen in 95% der Fälle erkennt, in denen es regnet. Er geht deshalb bei der Eingabe 1 („Es regnet.“) vom Zustand 1 in den Zustand 2 über. Wenn es weiterhin nicht regnet, so reagiert der Sensor in 98% aller Fälle richtig und verbleibt im Zustand 1:

$$\text{Prob}(Z(k+1) = 1 \mid Z(k) = 1, V(k) = 1) = 0,98.$$

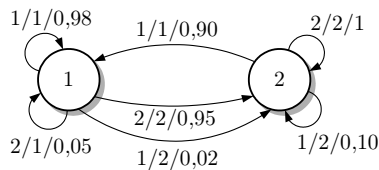
Das Ende des Regens erkennt der Sensor entsprechend der Angabe

$$\text{Prob}(Z(k+1) = 1 \mid Z(k) = 2, V(k) = 1) = 0,90$$

in 90% aller Fälle. Die letzte Angabe

$$\text{Prob}(Z(k+1) = 2 \mid Z(k) = 2, V(k) = 2) = 1$$

besagt, dass bei anhaltendem Regen der Sensor stets das richtige Ergebnis ausgibt.



**Abb. A.46:** Beschreibung eines Regensensors durch einen nichtdeterministischen Automaten

Diese Angaben finden sich an vier Kanten im Automatengraphen in Abb. A.46. Weitere Angaben können aus der Beziehung (11.57) ermittelt werden, denn die in der Aufgabenstellung gegebenen Angaben entsprechen gerade den Zustandsübergangswahrscheinlichkeiten

$$G(z' \mid z, v) = \text{Prob}(Z(k+1) = z' \mid Z(k) = z, V(k) = v).$$

Man erhält, ausführlich geschrieben, aus der ersten Angabe und Gl. (11.57) folgende Beziehungen:

$$\begin{aligned} \text{Prob}(Z(k+1) = 1 \mid Z(k) = 1, V(k) = 2) + \text{Prob}(Z(k+1) = 2 \mid Z(k) = 1, V(k) = 2) &= 1 \\ \text{Prob}(Z(k+1) = 1 \mid Z(k) = 1, V(k) = 2) &= 1 - \text{Prob}(Z(k+1) = 2 \mid Z(k) = 1, V(k) = 2) \\ &= 1 - 0,95 = 0,05. \end{aligned}$$

Diese Wahrscheinlichkeit ist an der für die Eingabe 2 gültigen Schlinge des Zustands 1 angetragen. Weiterhin erhält man

$$\begin{aligned} \text{Prob}(Z(k+1) = 2 \mid Z(k) = 1, V(k) = 1) &= 1 - \text{Prob}(Z(k+1) = 1 \mid Z(k) = 1, V(k) = 1) = 0,02 \\ \text{Prob}(Z(k+1) = 2 \mid Z(k) = 2, V(k) = 1) &= 1 - \text{Prob}(Z(k+1) = 1 \mid Z(k) = 2, V(k) = 1) = 0,10 \\ \text{Prob}(Z(k+1) = 1 \mid Z(k) = 2, V(k) = 2) &= 1 - \text{Prob}(Z(k+1) = 2 \mid Z(k) = 2, V(k) = 2) = 0. \end{aligned}$$

Die letzte Wahrscheinlichkeit besagt, dass die im Graphen des nichtdeterministischen Automaten eingetragene Kante vom Zustand 2 zum Zustand 1 für die Eingabe 2 niemals durchlaufen wird und deshalb entfallen kann.

### Aufgabe 11.13 Beschreibung eines Batchprozesses durch ein Petrinetz

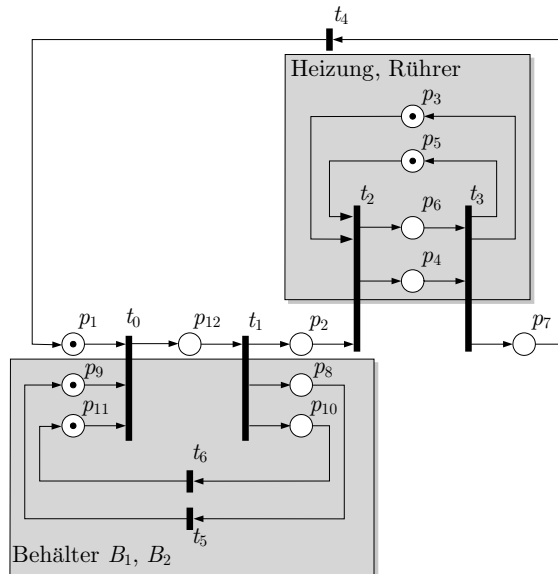
Für die Beschreibung des Batchprozesses werden folgende Situationen eingeführt:

Stelle	Situationsbeschreibung
$p_1$	Der Behälter $B_3$ ist leer.
$p_2$	Der Behälter $B_3$ ist mit kalter Flüssigkeit gefüllt.
$p_3$	Der Rührer ist ausgeschaltet.
$p_4$	Der Rührer ist eingeschaltet.
$p_5$	Die Heizung ist ausgeschaltet.
$p_6$	Die Heizung ist eingeschaltet.
$p_7$	Der Behälter $B_3$ ist mit erhitzter Flüssigkeit gefüllt.
$p_8$	Der Behälter $B_1$ ist leer.
$p_9$	Der Behälter $B_1$ ist gefüllt.
$p_{10}$	Der Behälter $B_2$ ist leer.
$p_{11}$	Der Behälter $B_2$ ist gefüllt.
$p_{12}$	Der Behälter $B_3$ wird gefüllt.

Während die Stellen  $p_1$  bis  $p_{11}$  statische Situationen beschreiben, in denen das System über eine beliebig lange Zeit verharren könnte, beschreibt die Stelle  $p_{12}$  einen dynamischen Vorgang, bei dem der Behälter  $B_3$  aus den Behältern  $B_1$  und  $B_2$  gefüllt wird.

Die während des Batchprozesses ablaufenden Teilprozesse führen auf folgende Bedingungen bzw. Aktionen, die das Ende der Teilprozesse anzeigen und durch Transitionen dargestellt werden:

Transition	Bedeutung
$t_0$	Das Füllen des Behälters $B_3$ wird begonnen.
$t_1$	Das Füllen des Behälters $B_3$ und das Leeren der Behälter $B_1$ und $B_2$ sind abgeschlossen.
$t_2$	Die Heizung und der Rührer werden eingeschaltet.
$t_3$	Die Heizung und der Rührer werden ausgeschaltet.
$t_4$	Das Entleeren des Behälters $B_3$ wird abgeschlossen.
$t_5$	Das Füllen des Behälters $B_1$ wird abgeschlossen.
$t_6$	Das Füllen des Behälters $B_2$ wird abgeschlossen.



**Abb. A.47:** Petrinetz zur Beschreibung des Batchprozesses

Das aus diesen Stellen und Transitionen bestehende Petrinetz ist in Abb. A.47 gezeigt. Obwohl natürlich alle Elemente untereinander verkoppelt sind, erkennt man die unterschiedlichen Bestandteile des Netzes. Der untere linke Teil zeigt das Füllen und Leeren der Behälter  $B_1$  und  $B_2$ , der obere rechte Teil das An- und Ausschalten der Heizung und des Rührers.

Im Petrinetz sind mehrere Synchronisationen vorhanden, wobei die Transition auf der linken Seite die auffälligste ist. Es müssen sowohl die Behälter  $B_1$  und  $B_2$  gefüllt und gleichzeitig der Behälter  $B_3$  entleert sein, damit der Füllvorgang von  $B_3$  beginnt.

Bei der Beschreibung der Stellen und Transitionen ist darauf zu achten, dass Stellen das Verharren oder die Bewegung des Systems innerhalb bestimmter Zeiträume darstellen, während das Schalten von Transitionen einen Zeitpunkt beschreibt.

#### Aufgabe 11.14 Steuerung einer Digitalkamera

1. Die betrachteten Prozesse sind in der folgenden Tabelle zusammengestellt:

Prozess	Komponente	Bedingung
Fokussieren	Objektiv	Auslöser ist halb gedrückt.
Belichtung messen	Belichtungsmesser	Auslöser ist halb gedrückt.
Belichten	Verschluss	Auslöser ist vollständig gedrückt, Fokussierung ist beendet, Belichtung ist gemessen.
Bild speichern	Speicher	Belichtung ist abgeschlossen.

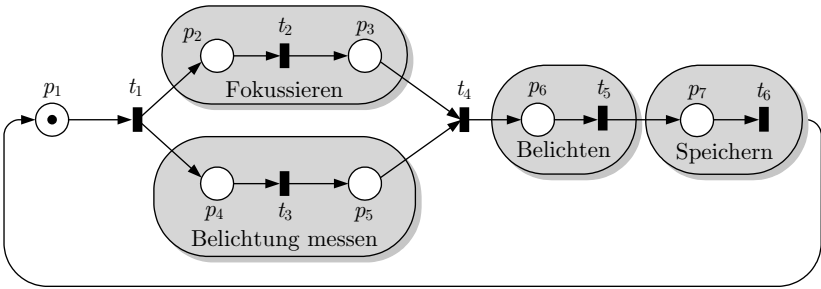


Abb. A.48: Beschreibung des Fotografierens mit einer Digitalkamera

2. Das in Abb. A.48 gezeigte Petrinetz verknüpft die vier betrachteten Prozesse in der gewünschten Weise. Die Stellen und Transitionen haben folgende Bedeutung:

Stelle	Bedeutung
$p_1$	Kamera ist in Ruhestellung
$p_2$	Apparat ist fokussiert
$p_3$	Fokussierung ist beendet
$p_4$	Belichtung wird gemessen
$p_5$	Belichtungszeit ist festgelegt
$p_6$	Bild wird belichtet
$p_7$	Daten werden gespeichert
Transition	Bedeutung
$t_1$	Fokussierung und Belichtungsmessung beginnen
$t_2$	Fokussierung ist beendet
$t_3$	Belichtungsmessung ist beendet
$t_4$	Fokussierung und Belichtungsmessung sind beendet
$t_5$	Belichtung ist abgeschlossen und Speichern kann beginnen
$t_6$	Speicherung der Daten ist abgeschlossen

Da sowohl das Fokussieren als auch die Belichtungsmessung abgeschlossen sein müssen, bevor das Bild belichtet wird, werden für die Darstellung dieser beiden parallel ablaufenden Vorgänge je zwei Stellen und eine Transition benötigt. Die beiden Marken warten in den Stellen  $p_3$  und  $p_5$  aufeinander.

Da während der Belichtung eines Bildes kein anderer Prozess abläuft, wird dieser Vorgang mit einer einzigen Stelle erfasst. Man könnte auch zwei Stellen einführen, von denen die eine den Beginn und die andere das Ende des Belichtungsvorgangs und die dazwischen liegende Transition den Belichtungsvorgang selbst kennzeichnet.

Das Petrinetz ist ein Synchronisationsgraph mit zwei parallelen und zwei sequenziellen Teilprozessen.

3. Die Steuerung erhält sowohl vom Nutzer als auch vom Fotoapparat Eingaben  $v_i$ , die das Schalten der Transitionen beeinflussen, und sendet Ausgaben  $w_i$  an die einzelnen Komponenten des Fotoapparates, um die Prozesse zu aktivieren. Die Eingaben und Ausgaben haben folgende Bedeutung:



Eingabe	Bedeutung
$v_1$	Der Auslöser ist halb betätigt.
$v_2$	Die Fokussierung ist beendet.
$v_3$	Die Belichtungsmessung ist beendet.
$v_4$	Der Auslöser ist vollständig betätigt.
$v_5$	Die Belichtung ist abgeschlossen.
$v_6$	Die Speicherung der Daten ist abgeschlossen.
Ausgabe	Bedeutung
$w_2$	Fokussierung soll beginnen
$w_4$	Belichtungsmessung soll beginnen
$w_6$	Belichtung soll beginnen
$w_7$	Speicherung der Daten soll beginnen

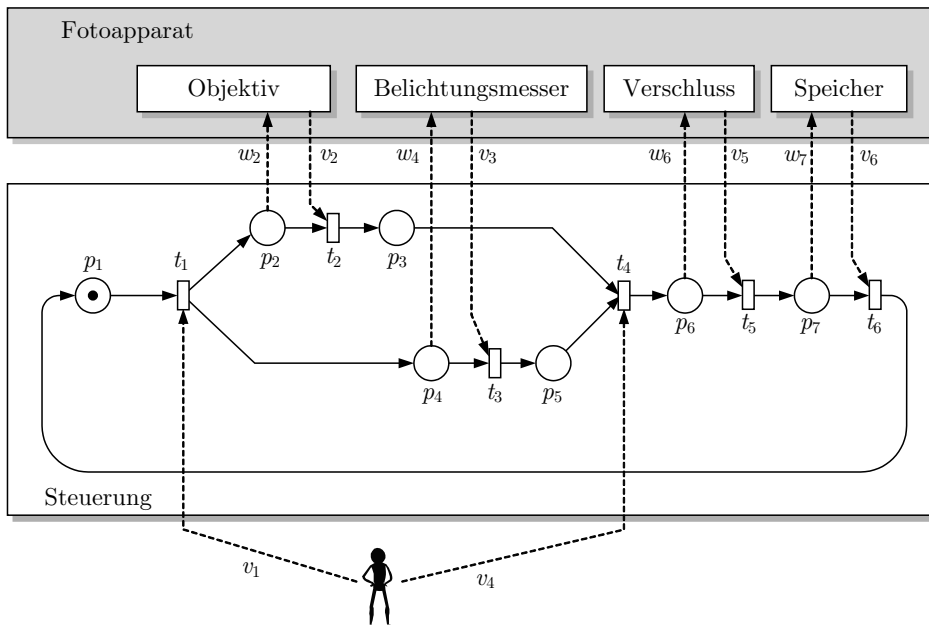


Abb. A.49: Blockschaltbild einer Digitalkamera

4. Das Blockschaltbild ist in Abb. A.49 dargestellt. Es zeigt die Kommunikation zwischen dem Nutzer, der Steuerung und den Komponenten des Fotoapparates, wobei hier gestrichelte Pfeile verwendet wurden, um diese Pfeile von den Kanten des Petrinetzes zu unterscheiden. Wenn eine Stelle markiert wird, erzeugt die Steuerung die zugehörige Ausgabe und startet dabei einen der Teilprozesse. Die Steuerung „merkt“, wann die Teilprozesse beendet sind aufgrund von Eingaben, die im Petrinetz einen Markenfluss auslösen.

Das Modell in Form eines Petrinetzes, das den Ablauf des Fotografierens beschreibt, wird also als Reglergesetz verwendet, wobei die einigen Stellen zugeordneten Ausgaben  $w_i$  einen Teilprozess

des Fotografierens auslösen und die Ausgaben  $v_i$  des Fotoapparats den Markenfluss im Petrinetz so beeinflussen, dass die Markierung des Petrinetzes stets dem aktuellen Zustand des Fotografiervorgangs entspricht. Die Bezeichnungen  $w_i$  und  $v_i$  sind in der Abbildung aus der Sicht der Steuerung vergeben, wobei  $w_i$  die Ausgaben und  $v_i$  die Eingaben der Steuereinrichtung darstellen.

**Aufgabe 11.15** *Beschreibung eines Streckenabschnitts der Eisenbahn*

Die Eingaben werden folgendermaßen definiert:

Eingabe	Beschreibung
$v_A$	Fahrtwunsch für den Streckenabschnitt $A \rightarrow C$
$v_2$	Fahrtwunsch für den Streckenabschnitt $C \rightarrow B$
$v_4$	Fahrtwunsch für den Streckenabschnitt $C \rightarrow A$
$v_B$	Fahrtwunsch für den Streckenabschnitt $B \rightarrow C$

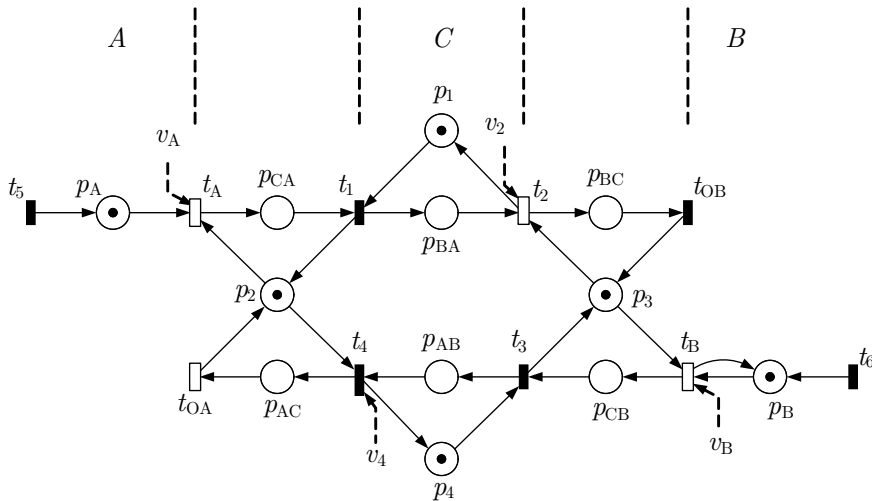
Die Stellen des Petrinetzes beschreiben die Belegung der Strecken, wobei markierte Stellen folgende Situationen kennzeichnen:

Stelle	Beschreibung
$p_A$	Es wartet ein Zug in $A$ , um nach $B$ zu fahren.
$p_B$	Es wartet ein Zug in $B$ , um nach $A$ zu fahren.
$p_{BA}$	Es wartet ein Zug in $C$ , um nach $B$ zu fahren.
$p_{AB}$	Es wartet ein Zug in $C$ , um nach $A$ zu fahren.
$p_{CA}$	Es fährt ein Zug von $A$ nach $C$ .
$p_{AC}$	Es fährt ein Zug von $C$ nach $A$ .
$p_{CB}$	Es fährt ein Zug von $B$ nach $C$ .
$p_{BC}$	Es fährt ein Zug von $C$ nach $B$ .
$p_1$	Die Ausweichstelle ist in Richtung $A \rightarrow B$ frei.
$p_2$	Der Streckenabschnitt zwischen $A$ und $C$ ist frei.
$p_3$	Der Streckenabschnitt zwischen $B$ und $C$ ist frei.
$p_4$	Die Ausweichstelle ist in Richtung $B \rightarrow A$ frei.

Abbildung A.50 zeigt das Petrinetz, das den Zugverkehr auf dem betrachteten Streckenabschnitt beschreibt. Die Stellen  $p_A$  und  $p_B$  sind aufgrund der Transitionen  $t_5$  und  $t_6$  ständig markiert, was bedeutet, dass entsprechend der Aufgabenstellung ständig Züge in  $A$  und  $B$  warten. Die in der Abbildung dargestellte Anfangsmarkierung bedeutet, dass alle Streckenabschnitte frei sind.

Wenn ein Zug von  $A$  nach  $C$  fahren soll ( $v_A = 1$ ), so ist dies nur möglich, wenn der Streckenabschnitt frei ist, was durch eine Markierung der Stelle  $p_2$  angezeigt wird. Dann erhält die Stelle  $p_{CA}$  eine Marke und die Stelle  $p_2$  verliert die Marke. Damit ist ausgeschlossen, dass der Streckenabschnitt mit einem zweiten Zug belegt wird. In derselben Weise sperren unmarkierte Stellen  $p_1$ ,  $p_3$  und  $p_4$  die anderen Streckenabschnitte.

**Diskussion.** Das Modell beschreibt, wie der durch die Eingaben ausgedrückte Fahrtwunsch mit der Streckenbelegung verknüpft werden muss, um den Zugverkehr zu steuern. Nur wenn beispielsweise  $v_A = 1$  gilt und die Stelle  $p_2$  belegt ist, kann ein Zug von  $A$  nach  $C$  fahren. Diese beiden Bedingungen können deshalb für die Steuerung des vor dem Streckenabschnitt stehenden Signals verwendet werden, das in diesem Fall auf grün gestellt wird.



**Abb. A.50:** Petrinetz, das den Eisenbahnverkehr auf dem in Abb. 11.43 gezeigten Streckenabschnitt beschreibt

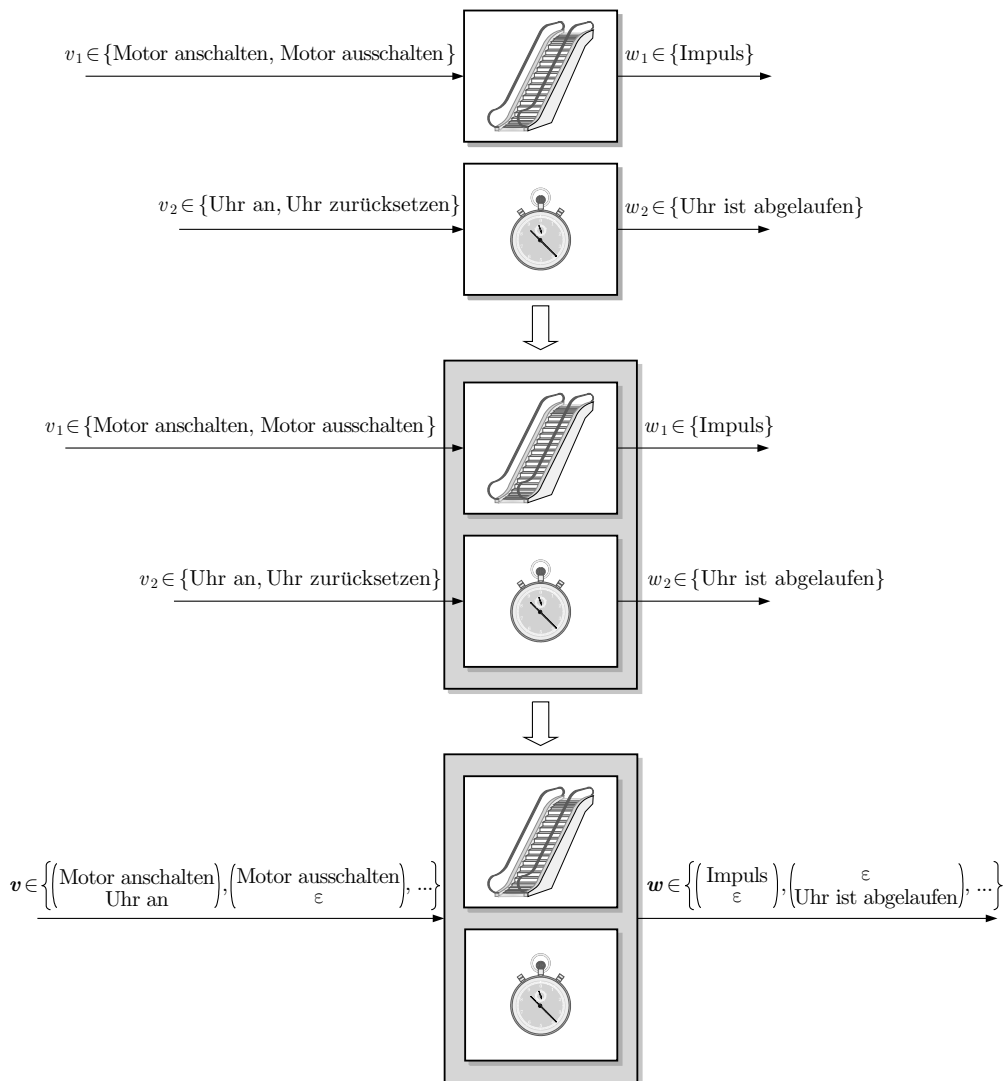
Die Stellen  $p_1$  und  $p_4$  kann man weglassen, denn entsprechend der Schaltregel kann eine bereits markierte Stelle nicht mit zusätzlichen Marken belegt werden (vgl. „Kontakt“ auf S. 392). Wenn also die Stelle  $p_{BA}$  markiert ist, so kann ein Zug vom Streckenabschnitt  $A \rightarrow C$  (Stelle  $p_{CA}$  markiert) nicht in die Ausweichstelle  $C$  einfahren. Dies wird im Petrinetz aus Abb. A.50 durch die unmarkierte Stelle  $p_1$  realisiert, derentwegen die Transition  $t_1$  nicht schalten kann.

Das Netz kann erweitert werden, um auszuschließen, dass Züge bei besetzter Ausweichstelle in die eingleisigen Abschnitte einfahren. Es kann dann verhindert werden, dass beispielsweise ein Zug von  $A$  nach  $C$  fährt, wenn die Ausweichstelle in  $C$  in der Richtung  $A \rightarrow B$  besetzt (also die Stelle  $p_{BA}$  markiert) ist.

### Aufgabe 11.18 Synchronisierte Darstellung der Steuerstrecke einer Rolltreppensteuerung

Im oberen Teil von Abb. A.51 sind die beiden Teilsysteme mit ihren Eingaben und Ausgaben dargestellt. Beide Teilsysteme haben jeweils zwei Modellzustände, die ihren Betriebszuständen „eingeschaltet“ und „ausgeschaltet“ entsprechen. Der von der Rolltreppe ausgegebene Impuls hat nichts mit dem Rolltreppenzustand zu tun, weil er durch Benutzer erzeugt wird, also in beiden Zuständen gleichermaßen auftreten kann.

Die Synchronisation beider Teilsysteme bedeutet, dass sie zu einem gemeinsamen System zusammengefügt werden. Das dabei entstehende System hat als Zustandsmenge das kartesische Produkt der Zustandsmengen beider Teilsysteme. Es hat zwei Eingangssignale und zwei Ausgangssignale, die zu einer gemeinsamen Eingabe bzw. Ausgabe zusammengefasst werden. Zweckmäßig ist hierbei die vektorielle Darstellung, aus der man erkennt, welche Werte die einzelnen Signale haben. Da beide Teilsysteme zu unterschiedlichen Zeiten Eingaben erhalten können, können die Eingaben beider Eingangsgrößen in beliebiger Kombination und darüber hinaus auch einzeln auftreten. Der letzte Fall bedeutet, dass die andere Eingangsgröße das leere Symbol  $\varepsilon$  besitzt. Gleiches gilt für die Ausgaben.



**Abb. A.51:** Synchronisation der Automaten für die Rolltreppe und die Uhr

### Aufgabe 11.19 Verallgemeinerung der Reihenschaltung

Für die drei Automaten gelten folgende Koppelbeziehungen:

$$v_{31}(k) = w_1(k)$$

$$v_{32}(k) = w_{22}(k).$$

Die Eingabe der Reihenschaltung setzt sich aus den Eingaben der beiden linken Automaten zusammen

$$\mathbf{v}(k) = \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix},$$

die Ausgabe aus denen der Automaten  $\mathcal{A}_2$  und  $\mathcal{A}_3$ :

$$\mathbf{w}(k) = \begin{pmatrix} w_3(k) \\ w_{21}(k) \end{pmatrix}.$$

Für die Berechnung der in Abb. 11.56 gezeigten Automaten werden für

$$\mathcal{A}_2 = (\mathcal{Z}_2, \mathcal{V}_2, \mathcal{W}_{21} \times \mathcal{W}_{22}, G_2, (H_{21}, H_{22}), z_{20})$$

zweckmäßigerweise zwei Ausgabegleichungen aufgeschrieben

$$\begin{aligned} z_2(k+1) &= G_2(z_2(k), v_2(k)) \\ w_{21}(k) &= H_{21}(z_2(k), v_2(k)) \\ w_{22}(k) &= H_{22}(z_2(k), v_2(k)), \end{aligned}$$

während die Zustandsübergangsfunktion des Automaten

$$\mathcal{A}_3 = (\mathcal{Z}_3, \mathcal{V}_{31} \times \mathcal{V}_{32}, \mathcal{W}_3, G_3, H_3, z_{30})$$

in Abhängigkeit von den zwei Eingaben  $v_{31}$  und  $v_{32}$  notiert wird:

$$\begin{aligned} z_3(k+1) &= G_3(z_3(k), v_{31}(k), v_{32}(k)) \\ w_3(k) &= H_3(z_3(k), v_{31}(k), v_{32}(k)). \end{aligned}$$

Für den Gesamtautomaten

$$\mathcal{A} = (\mathcal{Z}, \mathcal{V}, \mathcal{W}, G, H, z_0)$$

erhält man folgende Kompositionsregeln:

$$\begin{aligned} \mathcal{Z} &= \mathcal{Z}_1 \times \mathcal{Z}_2 \times \mathcal{Z}_3 \\ \mathcal{V} &= \mathcal{V}_1 \times \mathcal{V}_2 \\ \mathcal{W} &= \mathcal{W}_{21} \times \mathcal{W}_3 \end{aligned}$$

$$\begin{aligned} G \left( \begin{pmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \end{pmatrix}, \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix} \right) &= \begin{pmatrix} G_1(z_1(k), v_1(k)) \\ G_2(z_2(k), v_2(k)) \\ G_3(z_3(k), H_1(z_1(k), v_1(k)), H_{22}(z_2(k), v_2(k))) \end{pmatrix} \\ H \left( \begin{pmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \end{pmatrix}, \begin{pmatrix} v_1(k) \\ v_2(k) \end{pmatrix} \right) &= \begin{pmatrix} H_3(z_3(k), H_1(z_1(k), v_1(k)), H_{22}(z_2(k), v_2(k))) \\ H_{21}(z_2(k), v_2(k)) \end{pmatrix}. \end{aligned}$$

**Aufgabe 11.20** Rückkopplung mit mehreren Automaten

Für den Rückführautomaten müssen die folgenden Relationen gelten:

$$\begin{aligned}\mathcal{V}_1 &\supseteq \mathcal{W}_2 \\ \mathcal{V}_2 &\supseteq \mathcal{W}_1.\end{aligned}$$

Wenn einer der beiden Automaten ein Moore-Automat ist, ist der Rückführautomat wohldefiniert. Wenn beide Automaten Mealy-Automaten sind, entsteht eine „algebraische Schleife“. Dann muss gelten

$$\begin{aligned}v_1(k) &= w_2(k) \\ &= H_2(z_2(k), v_2(k)) \\ &= H_2(z_2(k), w_1(k)) \\ &= H_2(z_2(k), H_1(z_1(k), v_1(k))),\end{aligned}$$

oder kürzer geschrieben

$$v_1 = H_2(z_2, H_1(z_1, v_1)). \quad (\text{A.37})$$

Diese Gleichung muss für alle Zustandskombinationen  $\mathbf{z} = (z_1 \ z_2)^T \in \mathcal{Z}_1 \times \mathcal{Z}_2$  eine eindeutige Lösung in  $v_1$  haben, damit der Rückführautomat wohldefiniert ist. Wenn diese Lösung existiert, kann sie durch zwei Funktionen  $K_1$  und  $K_2$  in Abhängigkeit von den Zuständen  $z_1$  und  $z_2$  dargestellt werden:

$$\begin{aligned}v_1 &= K_1(z_1, z_2) \\ v_2 &= K_2(z_1, z_2).\end{aligned}$$

Für die Zustandsübergangsfunktion  $G$  des Rückführautomaten gilt

$$G\left(\begin{pmatrix} z_1(k) \\ z_2(k) \end{pmatrix}\right) = \begin{pmatrix} G_1(z_1(k), K_1(z_1, z_2)) \\ G_2(z_2(k), K_2(z_1, z_2)) \end{pmatrix}.$$

**Aufgabe 11.21** Berechnung zweier Rückführautomaten

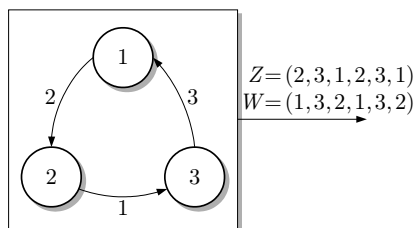
Rückgekoppelte Automaten sind nur dann wohldefiniert, wenn es in jedem Zustand genau eine Eingabe gibt, für die der Automat dieselbe Ausgabe erzeugt, also im Automatengraphen von jedem Zustand genau eine Kante mit einer Kantenbewertung der Form  $w/w$  abgeht. Diese Bedingung erfüllt nur der in Abb. 11.62 (a) gezeigte Automat. Deshalb existiert der Rückführautomat nur für diesen Automaten.

Der zugehörige Rückführautomat ist in Abb. A.52 zu sehen. Es ist ein autonomer Automat, der vom Anfangszustand  $z_0 = 2$  die angegebene Zustands- und Ausgabenfolge erzeugt.

**Aufgabe 11.22** Beschreibung eines Personenaufzugs durch gekoppelte Automaten

1. Für den Fahrkorb werden die folgenden Zustände  $z_1$  eingeführt:

$z_1$	Bedeutung
$z_O$	Der Fahrkorb steht im Obergeschoss.
$z_{EO}$	Der Fahrkorb fährt vom Obergeschoss ins Erdgeschoss.
$z_E$	Der Fahrkorb steht im Erdgeschoss.
$z_{OE}$	Der Fahrkorb fährt vom Erdgeschoss ins Obergeschoss



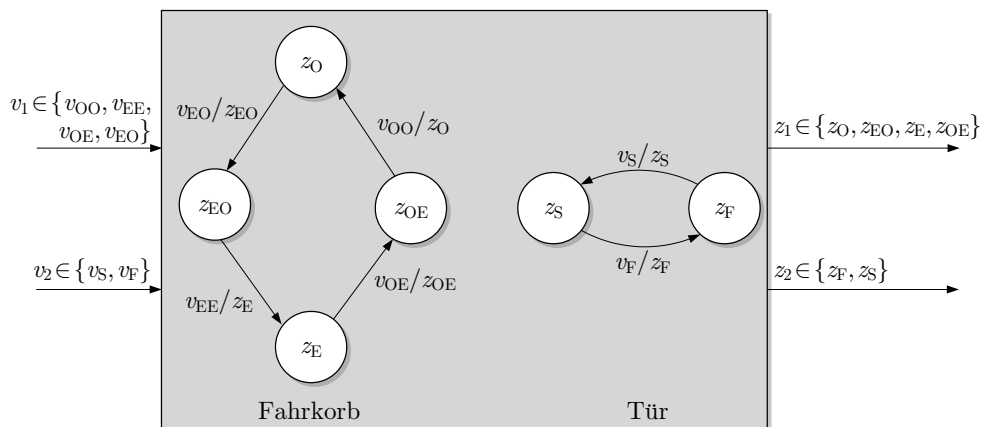
**Abb. A.52:** Rückführautomat, der den rückgekoppelten Automaten aus Abb. 11.62 (a) beschreibt

Der Fahrkorb wird vom Schalten des Motors beeinflusst, für den folgende Eingaben  $v_1$  definiert werden:

$v_1$	Bedeutung
$v_{OE}$	Fahrkorbmotor für Aufwärtsbewegung anschalten.
$v_{EO}$	Fahrkorbmotor für Abwärtsbewegung anschalten.

Dabei wird angenommen, dass sich der Motor selbstständig ausschaltet, wenn der Fahrkorb die neue Endposition erreicht hat. Diese Endposition wird durch Endlagenschalter überwacht, so dass der Zustand des Fahrkorbs bekannt ist. Folglich gilt

$$w_1(k) = z_1(k).$$



**Abb. A.53:** Beschreibung des Personenaufzugs durch zwei getrennte Automaten

Für die Tür werden die folgenden zwei Zustände  $z_2$  definiert:

$z_2$	Bedeutung
$z_S$	Die Tür ist geschlossen.
$z_F$	Die Tür ist geöffnet.

Für die Bewegung der Tür gibt es folgende Eingaben  $v_2$ :

$v_2$	Bedeutung
$v_S$	Tür schließen
$v_F$	Tür öffnen

2. Da Tür und Fahrkorb unabhängige Komponenten sind, erhält man das Gesamtmodell durch Synchronisation beider Teilautomaten (Abb. A.54). Aus den 4 bzw. 2 Zuständen der beiden Automaten entstehen durch Kombination  $2 \cdot 4 = 8$  Zustände des Gesamtautomaten, die als Vektor notiert werden. Zur Vereinfachung der Darstellung werden die an den Kanten stehenden Eingaben nicht als Vektor geschrieben. Auf Grund der Bezeichnungen ist offensichtlich, ob es sich um eine Eingabe für den Fahrkorb oder die Tür handelt.

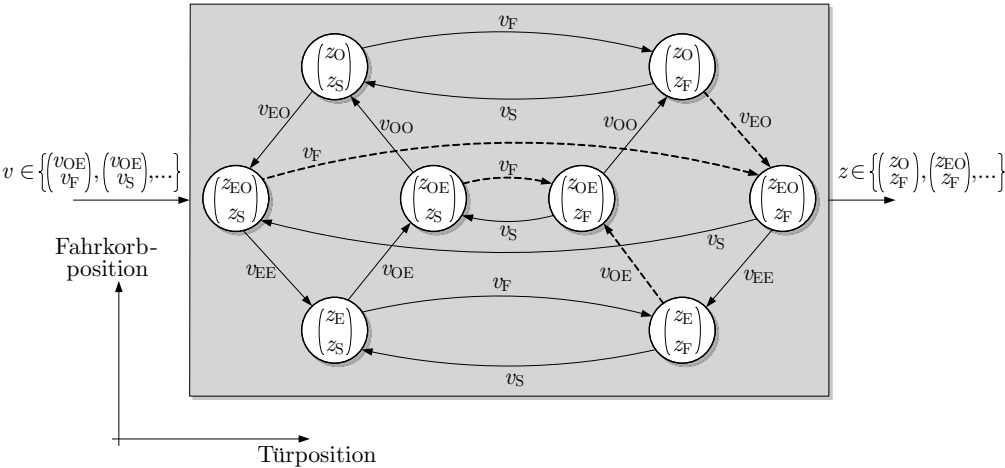


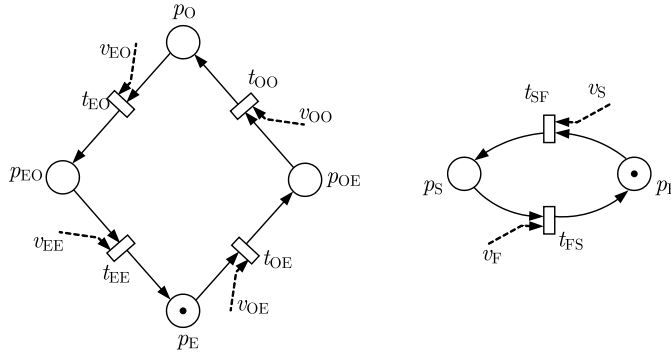
Abb. A.54: Automat, der den Personenaufzug beschreibt

Die gestrichelt eingetragenen Kanten sind nicht zulässig, denn sie bedeuten, dass der Fahrstuhl mit offener Tür fährt bzw. während der Fahrt die Tür öffnet. Diese Kanten zeigen, welche Eingaben bei welchen Zuständen  $z$  verboten sind, z. B.

$z$	verbotene Eingabe $v$	Bedeutung
$\begin{pmatrix} z_E \\ z_C \end{pmatrix}$	$\begin{pmatrix} v_{OE} \\ * \end{pmatrix}$	Fahrt EG $\rightarrow$ OG mit offener Tür
$\begin{pmatrix} z_O \\ z_C \end{pmatrix}$	$\begin{pmatrix} v_{EO} \\ * \end{pmatrix}$	Fahrt OG $\rightarrow$ EG mit offener Tür
$\begin{pmatrix} z_{OE} \\ z_F \end{pmatrix}$	$\begin{pmatrix} * \\ v_F \end{pmatrix}$	Öffnen der Tür während der Fahrt
$\begin{pmatrix} z_{EO} \\ z_F \end{pmatrix}$	$\begin{pmatrix} * \\ v_F \end{pmatrix}$	Öffnen der Tür während der Fahrt



3. Wenn man den Personenaufzug durch ein Petrinetz beschreibt, so werden die beiden unabhängigen Prozesse durch zwei getrennte Petrinetze beschrieben, wie Abb. A.55 zeigt. Die Stellen haben dieselben Indizes wie die Zustände des Automaten.



**Abb. A.55:** Petrinetz zur Beschreibung des Personenaufzugs

4. Die im ersten und dritten Schritt erhaltenen Modelle sind fast identisch, denn die beiden unabhängigen Petrinetze stellen Zustandsmaschinen dar, die auch als die beiden unabhängigen Automaten des ersten Modells interpretiert werden können. Die Zusammenführung der beiden Automaten zu einem Gesamtautomaten ergibt ein wesentlich größeres Modell (bezüglich der Zahl der Zustände). In diesem Modell können aber auch für bestimmte Zustände nicht anwendbare Eingaben erkannt werden, was mit den beiden anderen Modellen nicht ohne Weiteres möglich ist.

#### **Aufgabe 11.24** Kompositionale Modellbildung eines geregelten Vorratsbehälters

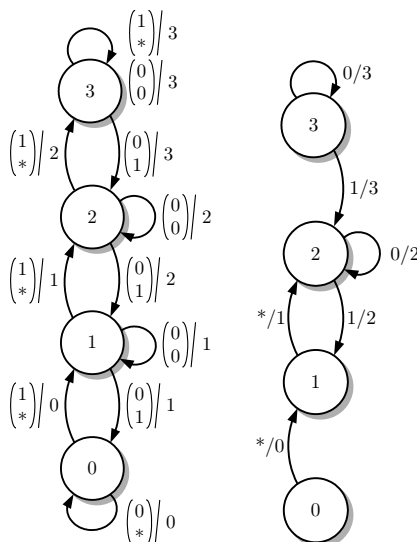
1. Zustand  $z$  und Ausgabe  $w$  stimmen bei diesem System überein, so dass ein Moore-Automat entsteht. Die vier Zustände beschreiben die vier diskreten Füllstände. Der Automat ist im linken Teil von Abb. A.56 gezeigt.

Die Eingabe des Behälters ist der Vektor  $\mathbf{v} = (v_1, v_2)^T$ . Wenn der Übergang von der Eingabe  $v_2$  unabhängig ist, weil bei geöffneten Ventilen voraussetzungsgemäß mehr Flüssigkeit zufließt als abfließt, so steht in der Position von  $v_2$  ein Stern.

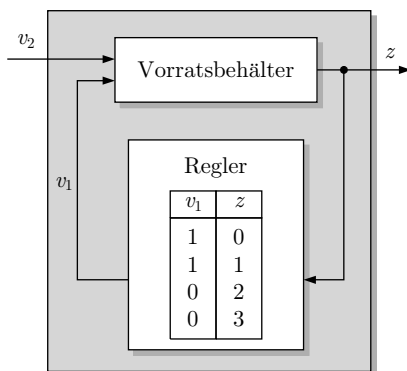
2. Im Regelkreis wird der Zustand  $z$  durch den Regler auf den Eingang  $v_1$  zurückgeführt (Abb. A.57). Man erhält das Modell des geschlossenen Kreises, indem man den Automaten für den Vorratsbehälter mit der gegebenen Rückführung kombiniert. Dabei verschwindet die Eingabe  $v_1$  an den Kanten des Automatengraphen und es bleibt ein Automat übrig, der die Eingabe  $v_2$  und die Ausgabe  $w$  hat.

Den im rechten Teil von Abb. A.56 gezeigten Automaten erhält man aus dem linken Automaten, indem man für jeden Zustand nur noch diejenigen Kanten zulässt, die an diesem Zustand beginnen und zur Eingabe  $v_1$  gehören, mit der die Regelung auf den Vorratsbehälter einwirkt. Beispielsweise gehen im linken Automaten vom Zustand 1 drei Kanten ab, die mit den Eingaben

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{bzw.} \quad \begin{pmatrix} 1 \\ * \end{pmatrix}$$



**Abb. A.56:** Beschreibung des Vorratsbehälters (links) und des geregelten Vorratsbehälters (rechts) durch deterministische Automaten



**Abb. A.57:** Regelkreis

markiert sind. Auf den Zustand 1 reagiert der Regler jedoch mit der Eingabe  $v_1 = 1$ . Folglich ist im geschlossenen Kreis nur die mit der dritten Markierung versehene Kante des Automatengraphen wirksam, denn die beiden anderen Kanten gehören zum Verhalten des Vorratsbehälters unter der Wirkung der für diesen Zustand nicht auftretenden Eingabe  $v_1 = 0$ . Da im Automaten des geschlossenen Kreises nur noch die verbleibende Eingabe  $v_2$  an die Kanten geschrieben wird, bleibt als Markierung des Eingangs der Stern, der einen beliebigen Wert der Eingabe  $v_2$  kennzeichnet. Die Ausgabe ist wie im Automat für die Steuerstrecke  $w = 1$ .

- Das Regelungsziel ist erreicht, denn unabhängig von der Eingangsgröße  $v_2$  bleibt der Füllstand des Vorratsbehälters in den diskreten Werten 1 oder 2. Die Zustände 0 und 3 sind auf Grund der Regelung nicht mehr von den beiden anderen Zuständen aus erreichbar. Andere Werte sind nur dann möglich,

wenn der Anfangszustand den Wert 0 oder 3 annimmt. Die Regelung sorgt aber dafür, dass auch in diesen Fällen der Füllstand die diskreten Werte 1 oder 2 annimmt.

4. Aus dem Automatengraphen des Regelkreises kann die folgende Zustandsfolge abgelesen werden:

$$Z = (2, 1, 2, 1, 2, 2).$$

**Diskussion.** Bei der verwendeten Steuerung handelt es sich um eine Verknüpfungssteuerung (vgl. Abschn. 13.2), also eine statische Zuordnung der Eingabe  $v_1$  zum gemessenen diskreten Zustand  $z$ . Da diese Steuerung keine eigenen Zustände hat, verändert sich die Zustandsmenge nicht, wenn das Modell des Behälters mit der Steuerung kombiniert wird.

### Aufgabe 12.2 Verhalten eines Regensors

Aus Gl. (12.12) für  $k = 1$

$$\text{Prob}(Z(1)=z') = \sum_{z \in \mathcal{Z}} \text{Prob}(Z(1)=z' \mid Z(0)=z, V(0)=v) \cdot \text{Prob}(Z(0)=z)$$

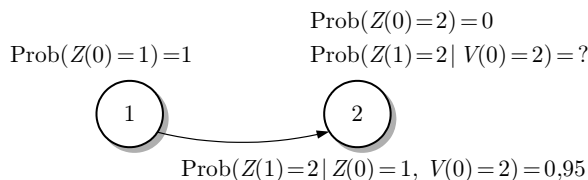
erhält man im Fall (a) für  $z' = 1$  mit der angegebenen Eingabe  $V(0) = 2$  die Beziehung

$$\begin{aligned} \text{Prob}(Z(1) = 1) &= \text{Prob}(Z(1)=1 \mid Z(0)=1, V(0)=2) \cdot \text{Prob}(Z(0)=1) + \\ &\quad \text{Prob}(Z(1)=1 \mid Z(0)=2, V(0)=2) \cdot \text{Prob}(Z(0)=2) \\ &= 0,05 \end{aligned}$$

und für  $z' = 2$

$$\begin{aligned} \text{Prob}(Z(1) = 2) &= \text{Prob}(Z(1)=2 \mid Z(0)=1, V(0)=2) \cdot \text{Prob}(Z(0)=1) + \\ &\quad \text{Prob}(Z(1)=2 \mid Z(0)=2, V(0)=2) \cdot \text{Prob}(Z(0)=2) \\ &= 0,95. \end{aligned}$$

Wenn der Regensor im Zustand 1 ist (bisher also keinen Regen angezeigt hat) und die Eingabe 2 erhalten hat („Es regnet.“), so schaltet er in 95% aller Fälle in den Zustand 2 und zeigt den Regen an. Die letzte Beziehung zeigt, dass auf Grund der verschwindenden Wahrscheinlichkeit  $\text{Prob}(Z(0)=2) = 0$  dieser Wert nur von der Zustandsübergangswahrscheinlichkeit für den Zustandswechsel  $1 \rightarrow 2$  abhängt. Abbildung A.58 zeigt den dafür maßgebenden Ausschnitt aus dem Automatengraphen.



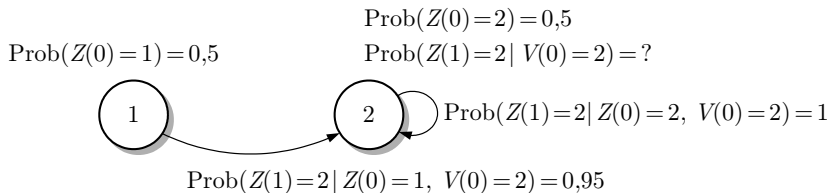
**Abb. A.58:** Zustandsübergang des Regensors für  $k = 0 \rightarrow k = 1$   
bei  $z_0 = 1$

Im Fall (b) verwendet man dieselben Beziehungen, jetzt jedoch mit anderen Daten:

$$\begin{aligned}\text{Prob}(Z(1) = 1) &= \text{Prob}(Z(1) = 1 \mid Z(0) = 1, V(0) = 2) \cdot \text{Prob}(Z(0) = 1) + \\ &\quad \text{Prob}(Z(1) = 1 \mid Z(0) = 2, V(0) = 2) \cdot \text{Prob}(Z(0) = 2) \\ &= 0,025\end{aligned}$$

$$\begin{aligned}\text{Prob}(Z(1) = 2) &= \text{Prob}(Z(1) = 2 \mid Z(0) = 1, V(0) = 2) \cdot \text{Prob}(Z(0) = 1) + \\ &\quad \text{Prob}(Z(1) = 2 \mid Z(0) = 2, V(0) = 2) \cdot \text{Prob}(Z(0) = 2) \\ &= 0,95 \cdot 0,5 + 1 \cdot 0,5 \\ &= 0,975\end{aligned}$$

(siehe Abb. A.59). Die Wahrscheinlichkeit hat sich gegenüber dem ersten Fall erhöht, weil der Regensensor bei vorhandenem Regen niemals in den Zustand 1 („Es regnet nicht.“) zurückschaltet, wenn es weiter regnet.



**Abb. A.59:** Zustandsübergang des Regensensors für  $k = 0 \rightarrow k = 1$  für die zweite Anfangszustandswahrscheinlichkeitsverteilung

Dieses einfache Beispiel zeigt, dass alle Angaben über das Verhalten stochastischer Automaten von der A-priori-Wahrscheinlichkeitsverteilung  $p_0(z)$  abhängen.

#### Aufgabe 12.4 Invarianten von Synchronisationsgraphen und Zustandsmaschinen

1. Die Netzmatrix des Synchronisationsgraphen lautet folgendermaßen:

$$N = \begin{matrix} & \begin{matrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 & t_9 \end{matrix} \\ \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \\ p_8 \\ p_9 \\ p_{10} \\ p_{11} \end{matrix} & \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{pmatrix} \end{matrix},$$

Für T-Invarianten  $\mathbf{t}$  muss die Beziehung  $\mathbf{N}\mathbf{t} = \mathbf{0}$  gelten. Diese Gleichung ist für alle  $\mathbf{t}$  der Form

$$\mathbf{t} = k(1, 1, 1, 1, 1, 1, 1, 1, 1)^T, \quad k \in \mathbb{N}$$

erfüllt. Die elementare Lösung erhält man bei  $k = 1$ . Das heißt, dass das Netz immer dann wieder in den Ausgangszustand zurückkehrt, wenn alle Transitionen gleich oft schalten.

Für die S-Invarianten muss die Beziehung  $\mathbf{s}^T \mathbf{N} = \mathbf{0}$  gelten. Daraus ergibt sich folgendes Gleichungssystem:

$$\begin{aligned} s_1 &= s_2 + s_3, & s_2 &= s_4, \\ s_3 &= s_9, & s_4 &= s_5 + s_6, \\ s_5 &= s_7, & s_6 &= s_8, \\ s_7 &= -s_8 + s_{11}, & s_9 &= s_{10}, \\ s_{10} &= -s_{11} + s_1. \end{aligned}$$

Für dieses Gleichungssystem gibt es die Lösungen

$$\begin{aligned} \mathbf{s}_1^T &= (1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1) \\ \mathbf{s}_2^T &= (1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1) \\ \mathbf{s}_3^T &= (1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0) \end{aligned}$$

und alle Linearkombinationen dieser Lösungen. Diese drei S-Invarianten entsprechen den drei Pfaden durch das Netz. Auf diesen wandern die Marken, so dass die Summe der Marken in diesen Stellen konstant bleibt.

2. Da in Zustandsmaschinen jeweils zwei Stellen durch eine Transition verbunden sind, kann sich die Markenanzahl beim Schalten einer Transition nicht verändern. Deshalb bleibt die Summe aller sich im Netz befindenden Marken konstant, was die S-Invariante

$$\mathbf{s}^T = (1 \ 1 \ 1 \ \dots \ 1)$$

aussagt. Wenn der Graph der Zustandsmaschine stark zusammenhängend ist, ist dies sogar die einzige S-Invariante.

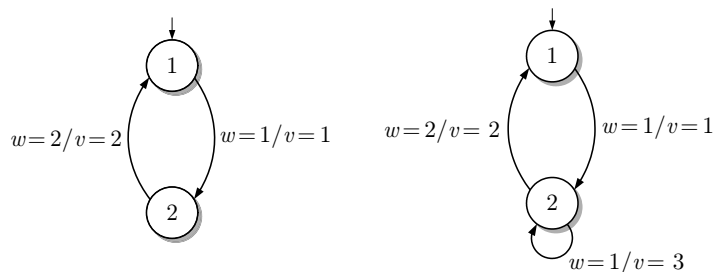
Da die Netzmatrix  $\mathbf{N}$  einer Zustandsmaschine in jeder Spalte genau eine  $+1$  und eine  $-1$  hat, hat sie einen kleineren als den maximal möglichen Rang und es existieren stets Invarianten.

### Aufgabe 13.2 Steuerung einer Treppenhausbeleuchtung

1. Die verwendeten Eingaben, Ausgaben und Zustände des um die Uhr der Steuereinrichtung erweiterten Prozesses sind in der folgenden Tabelle aufgeführt.

Zustand $z$	Bedeutung
1	Licht ist ausgeschaltet, Uhr steht
2	Licht ist angeschaltet, Uhr läuft
Ausgabe $w$	Bedeutung
1	Schalter wurde gedrückt
2	Uhr ist abgelaufen
Eingabe $v$	Bedeutung
1	Licht anschalten und Uhr starten
2	Licht ausschalten

2. Der Automat wird durch den in Abb. A.60 links gezeigten Graphen beschrieben. Die Ausgabe  $w$  der Treppenhausbeleuchtung ist die Eingabe des Steuerungsalgorithmus und die Eingabe  $v$  dessen Ausgabe.



**Abb. A.60:** Automatengraph der Steuerungen der Treppenhausbeleuchtung

Die folgende Automatentabelle ergibt sich direkt aus dem Graphen:

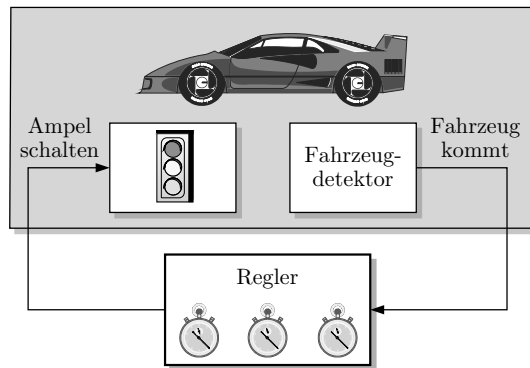
$z'$	$v$	$z$	$w$
2	1	1	1
1	2	2	2

3. Wenn die Uhr der Treppenhausbeleuchtung auch zurückgesetzt werden soll, wenn die Beleuchtung noch in Betrieb ist, so muss der Automat aus Abb. A.60 (links) um eine Schlinge am Zustand  $z = 2$  mit der Eingabe  $w = 1$  und der Ausgabe  $v = 3$  erweitert werden, wobei die neue Eingabe  $v = 3$  des gesteuerten Prozesses ein Rücksetzen der Uhr bewirkt (Abb. A.60 (rechts)).

### Aufgabe 13.3 Steuerung einer „Bremsampel“

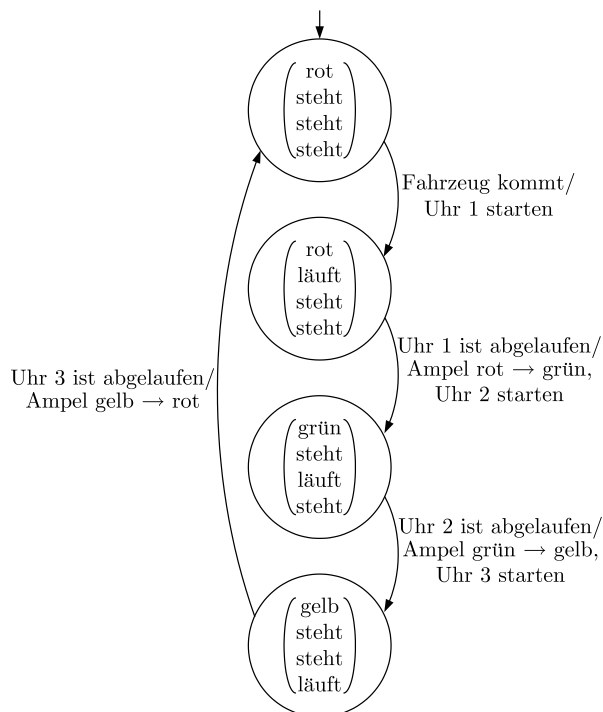
- Abbildung A.61 zeigt das Blockschaltbild der gesteuerten Ampel. Zur Steuerstrecke gehören der Sensor, die Ampel und das ankommende Fahrzeug. Sensor und Ampel sind nicht miteinander verkoppelt. Die Steuereinrichtung enthält drei Uhren, mit denen folgende Zeitintervalle gemessen werden:
  - die Zeit von der Annäherung des Fahrzeugs an das Ortsschild bis zur Umschaltung der Ampel auf grün,
  - die Zeitspanne von der Umschaltung auf grün bis zum Zurückschalten auf gelb und
  - die Zeitspanne, die die Ampel gelb zeigen soll, bevor sie wieder auf rot schaltet.
- Der Zustand der gesteuerten Ampel wird außer durch den Zustand dieser drei Uhren auch durch den gegenwärtigen Schaltzustand der Ampel bestimmt. Zum Vektor

$$\begin{pmatrix} \text{Ampelzustand} \\ \text{Zustand der Uhr 1} \\ \text{Zustand der Uhr 2} \\ \text{Zustand der Uhr 3} \end{pmatrix}$$



**Abb. A.61:** Blockschaltbild der gesteuerten „Bremsampel“

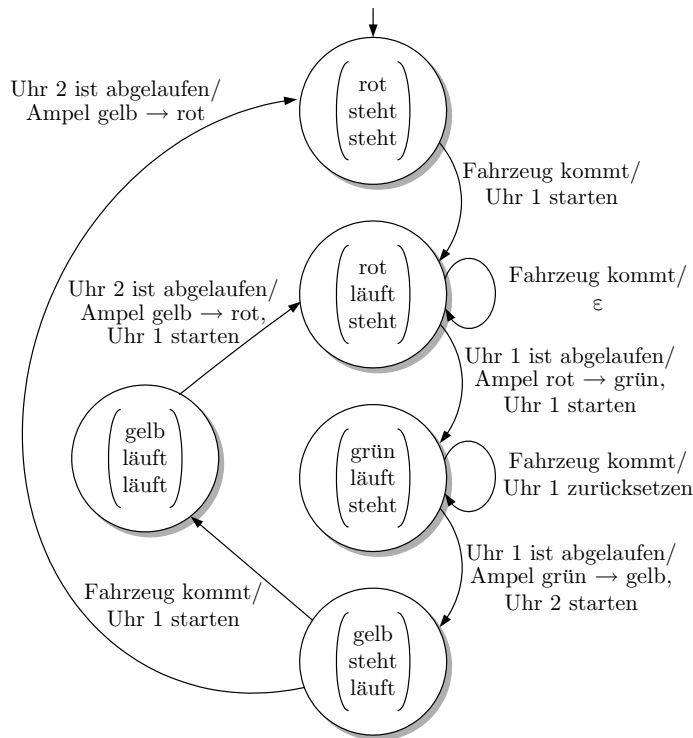
zusammengefasst ergeben sich die vier in Abb. A.62 gezeigten Zustände des Automaten, der die gesteuerte Ampel beschreibt.



**Abb. A.62:** Beschreibung der gesteuerten Ampel durch einen Automaten

Die gesteuerte Ampel ist ein autonomes System. Ihr Modell ist der in Abb. A.62 gezeigte Automat ohne die an den Kanten angegebenen Eingaben und Ausgaben. Das Modell zeigt die Folge der diskreten Zustände, die die gesteuerte Ampel durchläuft, also die Zustandsfolge, die man sieht, wenn man die Ampel über eine bestimmte Zeitspanne beobachtet.

3. Das Modell der gesteuerten Ampel kann man als Steuerungsalgorithmus einsetzen. Dafür nimmt man das Ampelmodell und schreibt die die Zustandsübergänge auslösenden Ereignisse als Ausgaben der Steuerstrecke und folglich als Eingaben des Steuerungsalgorithmus an die Kanten des Automatengraphen. Außerdem schreibt man die für die Zustandsübergänge notwendigen Schalthandlungen als Ausgaben der Steuerung an die Kanten dieses Graphen. Dabei entsteht aus dem Modell des autonomen Systems der in Abb. A.62 gezeigte Automatengraph mit den Kantenbeschriftungen.



**Abb. A.63:** Erweiterung der Beschreibung der Ampelsteuerung

An den Kanten des Automatengraphen in Abb. A.62 sind die Ausgaben der Steuerstrecke („Fahrzeug kommt“, „Uhr 1 ist abgelaufen“, „Uhr 2 ist abgelaufen“, „Uhr 3 ist abgelaufen“) als Eingaben und die durch die Steuerung vorgegebenen Befehle („Uhr 1 starten“, „Uhr 2 starten“, „Uhr 3 starten“, „Ampel von rot auf grün schalten“, „Ampel von grün auf gelb schalten“, „Ampel von gelb auf rot schalten“) als Ausgaben geschrieben. Dabei wurden wie im Beispiel 13.3 die Uhren zur Regelstrecke gezählt. Man kann sich anhand der von diesem Automaten erzeugten Zustandsfolge davon überzeugen, dass die Ampel wie gewünscht funktioniert.



4. Wenn auf mehrere Fahrzeuge Rücksicht genommen werden soll, so darf man die Ampel nicht auf rot schalten, wenn für ein nachfolgendes Fahrzeug nicht eindeutig feststeht, dass es zu schnell gefahren ist. Deshalb legt man die Zeit, nach der die Ampel von grün über gelb auf rot zurückgeschaltet wird, nicht beliebig fest, sondern wartet wiederum 7 Sekunden, was man mit der Uhr 1 messen kann. Damit enthält die Steuerung nur zwei Uhren, wobei die zweite Uhr die Zeit zwischen dem Umschalten auf gelb und dem Weiterschalten auf rot bestimmt. Der Zustand der Steuerung wird also außer dem Ampelzustand von zwei Uhren bestimmt.

Der über die vier in Abb. A.63 untereinander gezeichneten Zustände führende Schaltzyklus ist derselbe wie in Abb. A.62, nur dass hier nur zwei Uhren auftreten (was man auch für das Einzelfahrzeug hätte tun können). Zusätzliche Schleifen an den beiden mittleren Zuständen berücksichtigen nachfolgende Fahrzeuge, wobei die Uhr 1 durch nachfolgende Fahrzeuge nicht beeinflusst wird, weil sie nur die Geschwindigkeit des ersten Fahrzeugs begrenzen soll. Wenn ein solches Fahrzeug kommt, erhält die Ampelsteuerung die Eingabe „Fahrzeug kommt“, führt aber keine sichtbare Reaktion aus, was durch das leere Symbol  $\epsilon$  als Ausgabe ausgedrückt wird. Wenn die Uhr 1 zum zweitenmal angeschaltet wird, soll sie die Zeit bis zum Zurückschalten auf rot bestimmen. Nachfolgende Fahrzeuge setzen jetzt die Uhr zurück, weil für sie die Ampel nicht rechtzeitig auf rot geschaltet werden kann (Diese Fahrzeuge werden also bei zu hoher Geschwindigkeit nicht durch die Ampel gebremst). Wenn während der Gelbphase ein Fahrzeug am Ortsschild vorbeifährt, kann die Ampel noch rechtzeitig auf rot geschaltet werden, so dass diese Fahrzeuge bei zu hoher Geschwindigkeit vor der Ampel halten müssen.

#### Aufgabe 13.4 Steuerung eines Batchprozesses

1. Abbildung A.64 zeigt das Blockschaltbild des gesteuerten Batchprozesses. Die Steuereinrichtung erhält die Sensorsignale als einzige Informationen über den aktuellen Prozesszustand und schaltet die Pumpe und die beiden Ventile.

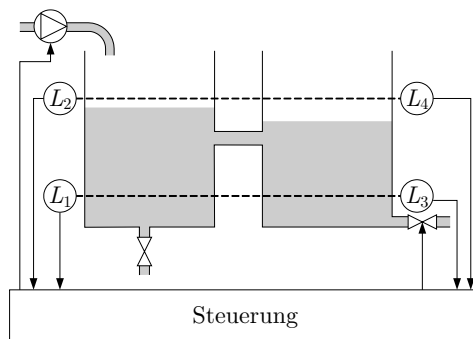


Abb. A.64: Blockschaltbild des gesteuerten Batchprozesses

2. Die vier binären Sensorwerte ergeben neun diskrete Prozesszustände, die wie in Abb. 12.6 auf S. 440 nummeriert werden. Davon sind für die Steuerungseinrichtung nur die Zustände 1, 2 und 9 wichtig:

$z$	Sensorwerte	Füllstände
1	$L_1 = L_2 = L_3 = L_4 = 0$	$h_1 < l_1, h_2 < l_3$
2	$L_1 = 1, L_2 = L_3 = L_4 = 0$	$l_1 \leq h_1 < l_2, h_2 < l_3$
9	$L_1 = L_2 = L_3 = L_4 = 1$	$l_2 \leq h_1, l_4 \leq h_2$

In der Tabelle bezeichnen  $l_i$ , ( $i = 1, \dots, 4$ ) die Positionen der Sensoren und  $L_i$  die Sensorwerte. Die Steuereinrichtung überprüft kontinuierlich die Sensorwerte um festzustellen, wann das Behältersystem in einen der drei angegebenen Zustände übergeht. Diese Zustandsübergänge werden bei der hier verwendeten diskreten Betrachtungsweise so interpretiert, dass das Behältersystem zu diesen Zeitpunkten die Ausgabe  $w = 1$ ,  $w = 2$  bzw.  $w = 9$  erzeugt. Die Berechnung dieser Ausgaben ist bei dem hier verwendeten Blockschaltbild eine Aufgabe der Steuereinrichtung.

3. Die Steuerung muss sich merken, in welchem der drei in der Spezifikation angegebenen Schritte sich der Prozess befindet. Deshalb hat die Steuerung den Charakter einer Ablaufsteuerung, für die die folgenden Zustände definiert werden:

$z_s$	Bedeutung
0	Ruhezustand
1	Die Behälter werden gefüllt.
2	Der rechte Behälter wird entleert.
3	Der linke Behälter wird entleert.

Die Eingaben  $v$  in den Prozess, die durch die Steuereinrichtung erzeugt werden, sind folgendermaßen bezeichnet:

$v$	Bedeutung
1	Pumpe einschalten, Ventile schließen
2	Pumpe ausschalten, Ventil $V_2$ öffnen
3	Ventil $V_2$ schließen, Ventil $V_1$ öffnen

Der Automatengraph der Steuerung ist in Abb. A.65 zu sehen. Die Steuerung befindet sich zu Beginn im Ruhezustand  $z_s = 0$ , wobei entsprechend der Spezifikation vorausgesetzt wird, dass beide Behälter leer sind. Das Anschalten des Prozesses wird durch die Eingabe 0 repräsentiert.

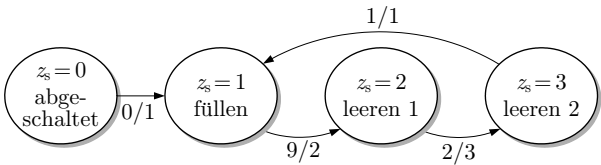


Abb. A.65: Automatengraph der Steuerung des Batchprozesses

4. Abbildung A.66 zeigt den zyklischen Batchprozess. Im oberen Teil der Abbildung sind die beiden Füllstände zu sehen, wobei die obere Kurve zu  $h_1$  und die untere zu  $h_2$  gehört. Durch die gestrichelten Linien sind die Positionen der Füllstandssensoren eingetragen.
- Zu dem gezeigten kontinuierlichen Verhalten gehört das im mittleren Teil der Abbildung dargestellte ereignisdiskrete Verhalten. Für die Steuereinrichtung ist nur wichtig, wann die Behälter die

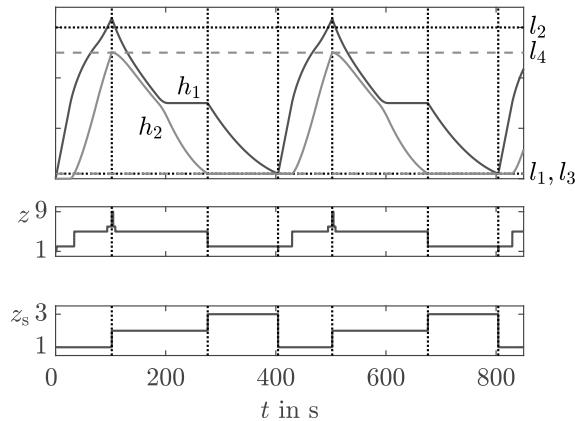


Abb. A.66: Verhalten des gesteuerten Behältersystems

diskreten Zustände 1, 2 und 9 erreichen. Diese Zeitpunkte sind im oberen Abbildungsteil durch die senkrechten Linien gekennzeichnet. Zu diesen Zeitpunkten wechselt die Steuerung ihren Zustand  $z_s$ , wie es im unteren Teil der Abbildung zu sehen ist, und gibt die neuen Stellgrößen aus.

Zwischen den Zuständen 1 und 9 bzw. 9 und 2 nimmt das Behältersysteme mehrere diskrete Zwischenzustände an, die für die betrachtete Steuerungsaufgabe nicht von Bedeutung sind. Die komplexe kontinuierliche Dynamik entsteht, weil das Verbindungsrohr nur Flüssigkeit transportiert, wenn sich wenigstens einer der beiden Füllstände oberhalb dieses Rohrs befindet. Deshalb sinkt beim Entleeren des rechten Behälters auch der Füllstand im linken Behälter bis zu dieser Höhe.

#### Aufgabe 14.1 Verhinderung verbotener Zustände bei einem Personenaufzug

Das in Abb. A.55 auf S. 643 gezeigte Petrinetz ist durch folgende Netzmatrix beschrieben:

$$N = \begin{matrix} & t_{OE} & t_{OO} & t_{EO} & t_{EE} & t_{SF} & t_{FS} \\ \begin{matrix} p_E \\ p_{OE} \\ p_{EO} \\ p_O \\ p_F \\ p_S \end{matrix} & \begin{pmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \end{matrix}.$$

Um die verbotenen Zustände zu verhindern, müssen die Ungleichungen

$$\begin{aligned} p_F(k) + p_{OE}(k) &\leq 1 \\ p_F(k) + p_{EO}(k) &\leq 1 \end{aligned}$$

gelten, was unter Verwendung der Matrix

$$L = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

und des Vektors

$$\mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

in der Form (14.6) geschrieben werden kann. Damit erhält man für die Steuerung (14.10) die Netzmatrix

$$\mathbf{N}_s = -\mathbf{L}\mathbf{N} = \begin{matrix} & t_{OE} & t_{OO} & t_{EO} & t_{EE} & t_{SF} & t_{FS} \\ \begin{matrix} p_{s1} \\ p_{s2} \end{matrix} & \begin{pmatrix} -1 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 & 1 & -1 \end{pmatrix} \end{matrix}.$$

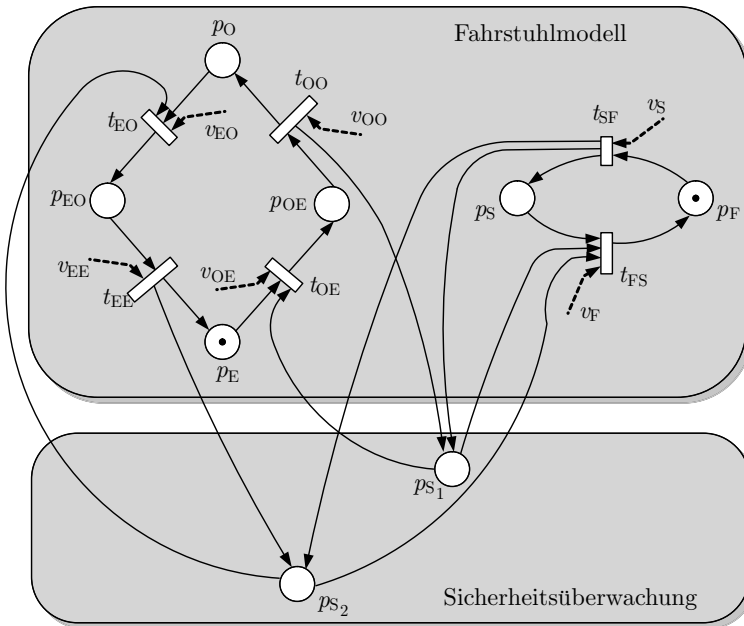
Der Fahrstuhl sei zur Zeit  $k = 0$  im Erdgeschoss und die Tür sei geschlossen. Dieser Anfangszustand ist durch die in Abb. A.55 angegebene Markierung

$$\mathbf{p}(0) = (1 \ 0 \ 0 \ 0 \ 1 \ 0)^T$$

beschrieben. Dafür erhält man die Anfangsmarkierung

$$\mathbf{p}_s = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

für die beiden Stellen  $p_{s1}$  und  $p_{s2}$ .



**Abb. A.67:** Petrinetz des gesteuerten Personenaufzugs

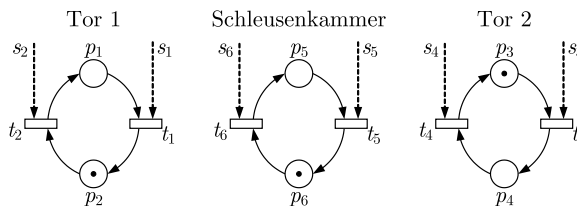
Abbildung A.67 zeigt die Wirkungsweise der entworfenen Sicherheitsüberwachung. Das im oberen Teil der Abbildung gezeigte Petrinetz beschreibt das Verhalten des Personenaufzugs, unten ist der Teil des Petrinetzes gezeigt, der die Wirkung der Sicherheitsüberwachung darstellt. Die Bewegung der Tür und des

Fahrkorbs sollen durch die von außen kommenden Eingaben  $v_{EO}$ ,  $v_{OO}$ ,  $v_{OE}$  und  $v_{EE}$  für den Fahrkorb und  $v_S$  und  $v_F$  für die Tür ausgelöst werden, die Eingangsgrößen der entsprechenden Transitionen sind. Die diesen Eingaben zugeordneten Bewegungen des Personenaufzugs kommen nur zustande, wenn die Stellen der Sicherheitsüberwachung entsprechend markiert sind. Beispielsweise bedeutet die in der Abbildung gezeigte Markierung, dass der Fahrkorb im Erdgeschoss steht und die Tür geöffnet ist. Wenn jetzt durch die Eingabe  $v_{OE}$  der Fahrkorb in Richtung Obergeschoss gestartet werden soll, so hat diese Eingabe keine Wirkung, weil die Stelle  $p_{S_1}$  der Sicherheitsüberwachung nicht markiert ist. Die Sicherheitsüberwachung weiß, dass die Tür geöffnet ist, und verhindert die Abfahrt des Fahrkorbs. Wird allerdings die Eingabe  $v_S$  zum Schließen der Tür angelegt, so wird die Tür geschlossen und damit auch die Stelle  $p_{S_1}$  der Sicherheitsüberwachung markiert. Anschließend kann sich der Fahrkorb, wenn er die Eingabe  $v_{OE}$  erhält, ins Obergeschoss bewegen. Während der Bewegung ist die Stelle  $p_{S_1}$  unmarkiert, was verhindert, dass man mit der Eingabe  $v_F$  die Tür während der Fahrt öffnen kann. Dasselbe passiert bei der Rückfahrt mit Hilfe der Stelle  $p_{S_2}$ .

Das in der Abbildung dargestellte Modell des Personenaufzugs mit Sicherheitsüberwachung zeigt, dass die Sicherheitsanforderungen erfüllt werden. Für die Realisierung der Sicherheitsüberwachung muss man das erhaltene Steuerungsgesetz so einsetzen, wie es in den Abbildungen 14.6 bis 14.8 auf S. 496 für die Sicherheitsschleuse gezeigt ist. Das in Abb. A.67 gezeigte Modell wird verwendet, um die von einem Nutzer des Personenaufzugs oder einer dort installierten Steuerung vorgegebenen Eingaben  $v_{EO}$ ,  $v_{OO}$ , ...,  $v_F$ , die in Abb. 14.6 zum Vektor  $s$  zusammengefasst sind, auf ihre Zulässigkeit zu untersuchen und nur dann als Kommandos  $v$  zu den Motoren der Tür bzw. des Fahrkorbs weiterzuleiten, wenn dadurch die Sicherheitsforderungen nicht verletzt werden. Die Stellsignale, die an die Strecke geleitet werden, entstehen dabei wie in Abb. 14.7 und 14.8 als Ausgaben der Stellen des in Abb. A.67 gezeigten Petrinetzes. Das gezeigte Petrinetz wirkt dabei wie bei der Sicherheitstür als Steuerung (in der offenen Wirkungskette) oder als Regler.

### Aufgabe 14.2 Steuerung einer Schiffsschleuse

1. Das Petrinetz ist in Abb. A.68 dargestellt. Die Bedeutung der Kommandos  $s_i$  ergibt sich aus der Bedeutung der Transitionen, an denen sie angreifen. Das Modell enthält die folgenden Stellen und Transitionen:



**Abb. A.68:** Modell der ungesteuerten Schiffsschleuse

Stelle	Bedeutung
$p_1$	Tor 1 ist offen.
$p_2$	Tor 1 ist geschlossen.
$p_3$	Tor 2 ist offen.
$p_4$	Tor 2 ist geschlossen.
$p_5$	Der Wasserstand liegt auf dem Niveau des Oberwassers.
$p_6$	Der Wasserstand liegt auf dem Niveau des Unterwassers.
Transition	Bedeutung
$t_1$	Das Tor 1 wird geschlossen.
$t_2$	Das Tor 1 wird geöffnet.
$t_3$	Das Tor 2 wird geschlossen.
$t_4$	Das Tor 2 wird geöffnet.
$t_5$	Der Wasserstand wird abgesenkt.
$t_6$	Der Wasserstand wird angehoben.

Das Modell hat die Netzmatrix

$$N = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}.$$

2. In der Form (14.6) aufgeschrieben heißt die Spezifikation

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}}_L \begin{pmatrix} p_1(k) \\ p_2(k) \\ p_3(k) \\ p_4(k) \\ p_5(k) \\ p_6(k) \end{pmatrix} \leq \underbrace{\begin{pmatrix} 1 \\ 1 \end{pmatrix}}_b.$$

Man kann diese Ungleichungen um die Bedingung erweitern, dass beide Tore nicht gleichzeitig offen sein dürfen ( $p_1(k) + p_3(k) \leq 1$ ), aber diese Bedingung ist implizit in den beiden angegebenen enthalten. Die mit der erweiterten Spezifikation berechnete Steuerung hat eine Stelle mehr und erfüllt die angegebenen Bedingungen gleichfalls.

3. Entsprechend Satz 14.1 verhindert die Steuerung mit der Netzmatrix

$$N_s = -LN = \begin{pmatrix} 1 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 & 1 & -1 \end{pmatrix}$$

und der Anfangsmarkierung

$$p_{s0} = b - Lp(0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

die verbotenen Zustände. Die Kopplung dieser Steuerung mit dem Modell der Schiffsschleuse ist in Abb. A.69 zu sehen.

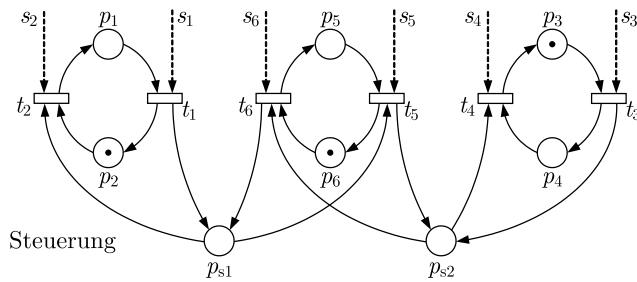


Abb. A.69: Modell der gesteuerten Schiffsschleuse

4. Wie im Beispiel 14.3 kann die Steuerung in der offenen Wirkungskette eingesetzt werden, wenn Sie entsprechend dem aktuellen Schleusenzustand initialisiert wird. Im geschlossenen Kreis, bei dem die aktuellen Torpositionen und der Wasserstand in der Schleusenammer als zusätzliche Eingänge an die Transitionen geführt werden, können Störungen vermieden werden, die beispielsweise das vollständige Schließen eines Tores verhindern.

**Aufgabe 15.1 Zustandsbeobachtung eines nichtdeterministischen Automaten**

Die Anwendung des Algorithmus 15.1 auf das E/A-Paar (15.6) führt für den in Abb. 15.1 gezeigten Automaten mit der Anfangsinformation

$$\mathcal{Z} = \mathcal{Z}(0 \mid -1) = \{1, 2, 3, 4, 5, 6\}$$

auf folgende Ergebnisse:

1.  $k_e = 0$ : E/A-Paar  $(v(0)=2, w(0)=1)$

$$\mathcal{Z} = \mathcal{Z}(0 \mid 0) = \{2, 4\}$$

$$\mathcal{Z}' = \mathcal{Z}(1 \mid 0) = \{1, 5\}$$

(vgl. die obere Zeile von Abb. A.70).

2.  $k_e = 1$ : E/A-Paar  $(v(1)=2, w(1)=2)$

$$\mathcal{Z} = \mathcal{Z}(1 \mid 1) = \{5\}$$

$$\mathcal{Z}' = \mathcal{Z}(2 \mid 1) = \{5, 6\}$$

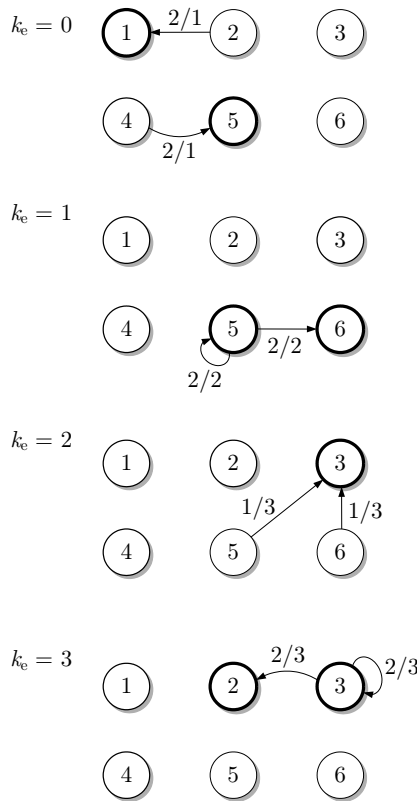
(vgl. zweite Zeile von Abb. A.70).

3.  $k_e = 2$ : E/A-Paar  $(v(2)=1, w(2)=3)$

$$\mathcal{Z} = \mathcal{Z}(2 \mid 2) = \{5, 6\}$$

$$\mathcal{Z}' = \mathcal{Z}(3 \mid 2) = \{3\}$$

(vgl. dritte Zeile von Abb. A.70). Jetzt ist der Automatenzustand zum Zeitpunkt  $k = 3$  exakt bekannt, obwohl erst die Eingaben und Ausgaben bis zur Zeit  $k = 2$  ausgewertet wurden.



**Abb. A.70:** Ergebnisse der Zustandsbeobachtung für das E/A-Paar (15.6)

4.  $k_e = 3$ : E/A-Paar ( $v(3) = 2, w(3) = 3$ )

$$\mathcal{Z} = \mathcal{Z}(3 | 3) = \{3\}$$

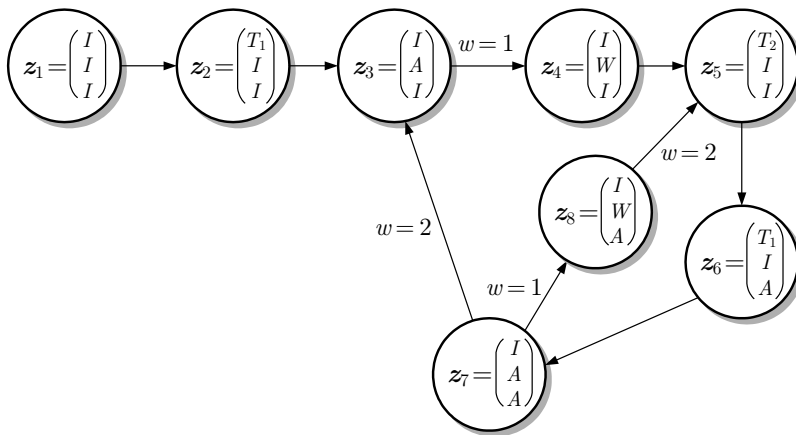
$$\mathcal{Z}' = \mathcal{Z}(4 | 3) = \{2, 3\}$$

(vgl. vierte Zeile von Abb. A.70). Auf Grund des Nichtdeterminismus des Automaten sind jetzt wieder zwei Zustände möglich.

### Aufgabe 15.2 Zustandsbeobachtung einer Fertigungszelle

- Die Ausgabe  $w = 1$  wird erzeugt, wenn die Maschine  $M_1$  den Zustandsübergang  $A \rightarrow W$  ausführt. Dazu gehören die Kanten  $z_3 \rightarrow z_4$  und  $z_7 \rightarrow z_8$  des Automatengraphen. Die Ausgabe  $w = 2$  erscheint beim Zustandsübergang  $A \rightarrow I$  der Maschine  $M_2$ , also im Automatengraphen bei den Übergängen  $z_7 \rightarrow z_3$  und  $z_8 \rightarrow z_5$ . Dadurch erhält man den in Abb. A.71 gezeigten Automatengraphen.





**Abb. A.71:** Automatengraph der Fertigungszelle mit Kennzeichnung der Ausgaben

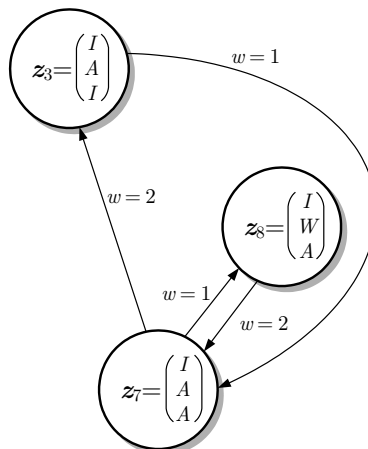
2. Die folgenden Zustandsübergänge finden statt, ohne dass die Überwachungseinrichtung der Fertigungszelle dies erkennen kann:

$$z_1 \rightarrow z_2, \quad z_2 \rightarrow z_3$$

$$z_4 \rightarrow z_5, \quad z_5 \rightarrow z_6$$

$$z_6 \rightarrow z_7.$$

Nach dem Streichen dieser Kanten entsteht der in Abb. A.72 gezeigte Automatengraph.



**Abb. A.72:** Reduzierter Automat der Fertigungszelle

Die Überwachungseinrichtung kann also nur zwischen den Zuständen  $z_3$ ,  $z_7$  und  $z_8$  unterscheiden. Dabei repräsentiert jeder dieser Zustände mehrere Zustände des ausführlichen Modells, nämlich

$$z_3 \sim \{z_1, z_2, z_3\}$$

$$z_7 \sim \{z_4, z_5, z_6, z_7\}$$

$$z_8 \sim \{z_8\}.$$

Nur wenn sich die Fertigungszelle im Zustand  $z_8$  befindet, ist der Zustand auch im Bezug auf das ausführliche Modell eindeutig festgelegt. Der Grund hierfür liegt in der Tatsache, dass vor und nach diesem Zustand ein Zustandsübergang liegt, der von der Überwachungseinrichtung aufgrund einer Ausgabe  $w \in \{1, 2\}$  erkennbar ist.

### 3. Die Anwendung des Beobachtungsalgorithmus 15.1 mit der Initialisierung

$$\mathcal{Z}' = \{z_3, z_7, z_8\}$$

führt auf folgende Ergebnisse, wobei hier an Stelle eines E/A-Paares  $(v, w)$  nur die Ausgabe  $w$  verarbeitet wird:

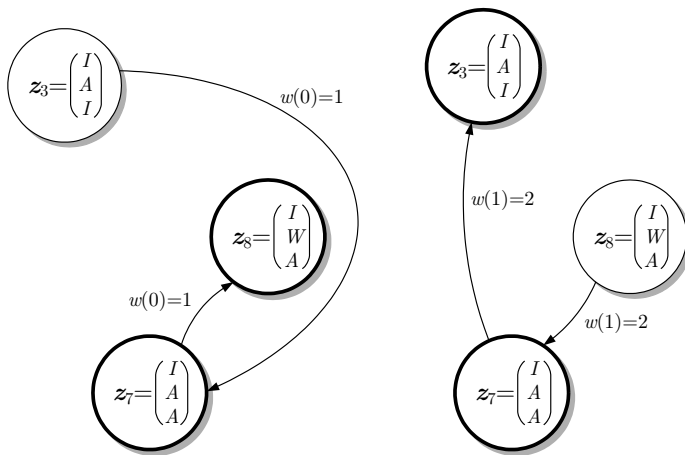


Abb. A.73: Beobachtungsergebnis der Fertigungszelle

#### Ausgabefolge (a):

- $k_e=0$ : Ausgabe  $w(0) = 1$

$$\mathcal{Z} = \mathcal{Z}(0 | 0) = \{z_3, z_7\}$$

$$\mathcal{Z}' = \mathcal{Z}(1 | 0) = \{z_7, z_8\}$$

Das Ergebnis ist im linken Teil von Abb. A.73 angegeben. Die Fertigmeldung der Maschine  $M_1$  allein ermöglicht keine exakte Bestimmung des aktuellen Zustands der Fertigungszelle, reduziert die Zustände jedoch auf zwei mögliche ( $z(1) \in \{z_7, z_8\}$ ).

- $k_e = 1$ : Ausgabe  $w(1) = 2$

$$\mathcal{Z} = \mathcal{Z}(1 | 1) = \{z_7, z_8\}$$

$$\mathcal{Z}' = \mathcal{Z}(2 | 1) = \{z_3, z_7\}$$

(siehe rechten Teil von Abb. A.73). Auch nach der zweiten Ausgabe ist der Zustand der Fertigungszelle nicht eindeutig bestimmt.

Das Beispiel zeigt, dass auch bei diskreten Systemen einzelne Messungen keine vollständigen Informationen über den Systemzustand geben. Nur unter Verwendung eines dynamischen Modells und durch Auswertung einer Folge von Ausgaben (und Eingaben) kann man den Zustand exakt bestimmen.

**Ausgabefolge (b):** Für die zweite Ausgabefolge liefert der Beobachtungsalgorithmus 15.1 im ersten Schritt dasselbe Ergebnis wie bei der ersten Ausgabefolge:

$$\mathcal{Z} = \mathcal{Z}(0 | 0) = \{z_3, z_7\}$$

$$\mathcal{Z}' = \mathcal{Z}(1 | 0) = \{z_7, z_8\}.$$

Nach der Auswertung der anschließenden Ausgabe  $w(1) = 1$  erhält man folgendes Ergebnis:

$$\mathcal{Z} = \mathcal{Z}(1 | 1) = \{z_7\}$$

$$\mathcal{Z}' = \mathcal{Z}(2 | 1) = \{z_8\}.$$

Das heißt, die Fertigungszelle befindet sich im Zustand  $z(2) = z_8$ .

### Aufgabe 16.1 Diagnose eines Batchprozesses

Für das E/A-Paar (16.7) erhält man durch Anwendung des Algorithmus 16.2 für  $k_e = 0$  dieselben Ergebnisse wie im Beispiel 16.1. Ausgehend von

$$\mathcal{Z}'(0) = \{1, 4, 5, 6\}$$

$$\mathcal{Z}'(1) = \{1\}$$

bekommt man für das zweite E/A-Paar ( $v(1) = 2, w(1) = 3$ ) die folgenden Mengen:

$$\mathcal{Z}(0) = \{4, 5, 6\}$$

$$\mathcal{Z}'(0) = \{1, 4, 5, 6\}$$

$$\mathcal{Z}(1) = \emptyset$$

$$\mathcal{Z}'(1) = \emptyset.$$

Das E/A-Paar ist mit dem fehlerbehafteten System inkonsistent. Für die weiteren E/A-Paare ( $v(2) = 2, w(2) = 2$ ) und ( $v(3) = 2, w(3) = 2$ ) bleibt das Verhalten konsistent mit dem Modell des fehlerfreien Systems. Damit ist erwiesen, dass kein Fehler aufgetreten ist (Fehlererkennung). Da das E/A-Paar mit dem Modell des fehlerfreien Systems konsistent ist, kann der Fehler ausgeschlossen werden (Fehleridentifikation). Der Ausschluss von Fehlern ist übrigens auch ein Diagnoseergebnis (und zwar das im Normalbetrieb erhoffte!).

Für das E/A-Paar (16.8) erhält man folgende Ergebnisse:

1.  $k_e = 0$ : E/A-Paar ( $v(0) = 1, w(0) = 3$ )

$$\mathcal{Z}(0) = \{4, 5, 6\}$$

$$\mathcal{Z}'(0) = \{2, 3\}$$

$$\mathcal{Z}(1) = \{4, 5, 6\}$$

$$\mathcal{Z}'(1) = \{1\}$$

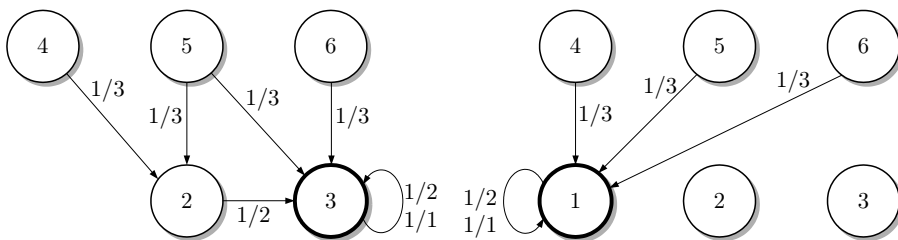
2.  $k_e = 1$ : E/A-Paar ( $v(1) = 1, w(1) = 2$ )

$$\mathcal{Z}(0) = \{2, 3\}$$

$$\mathcal{Z}'(0) = \{3\}$$

$$\mathcal{Z}(1) = \{1\}$$

$$\mathcal{Z}'(1) = \{1\}$$



**Abb. A.74:** Teile der Automatengraphen, die für die Lösung der Diagnoseaufgabe wesentlich sind

3.  $k_e = 2$ : E/A-Paar ( $v(2) = 1, w(2) = 2$ )

$$\mathcal{Z}(0) = \{3\}$$

$$\mathcal{Z}'(0) = \{3\}$$

$$\mathcal{Z}(1) = \{1\}$$

$$\mathcal{Z}'(1) = \{1\}$$

Abbildung A.74 zeigt den Ausschnitt der Automatengraphen, der für die Auswertung des E/A-Paares relevant ist.

4.  $k_e = 3$ : E/A-Paar ( $v(3) = 1, w(3) = 1$ )

$$\mathcal{Z}(0) = \{3\}$$

$$\mathcal{Z}'(0) = \{3\}$$

$$\mathcal{Z}(1) = \{1\}$$

$$\mathcal{Z}'(1) = \{1\}.$$

Das E/A-Paar ist konsistent mit beiden Automaten. Auf Grund dieser Messungen kann also weder der fehlerfreie Fall noch der Fehlerfall ausgeschlossen werden und die Menge  $\mathcal{F}(3)$  der Fehlerkandidaten enthält beide Fehler. Das beobachtete Verhalten können beide Automaten zeigen, so dass die Diagnose mit dieser Messung zu keinem Ergebnis kommt.

### Aufgabe 16.2 Diagnose zweier gekoppelter Reaktoren

Die Lösung der Aufgabe mit Hilfe des Algorithmus 16.2 erfolgt ähnlich wie in Aufgabe 16.1. Es werden hier deshalb nur die nacheinander berechneten Mengen angegeben.

- (a) fehlerfreier Fall      Fehlerfall
- |                                |                            |
|--------------------------------|----------------------------|
| $\mathcal{Z}(1 0) = \{6, 9\}$  | $\mathcal{Z}(1 0) = \{6\}$ |
| $\mathcal{Z}(2 1) = \{6, 9\}$  | $\mathcal{Z}(2 1) = \{3\}$ |
| $\mathcal{Z}(3 2) = \emptyset$ | $\mathcal{Z}(3 2) = \{3\}$ |

Das erste E/A-Paar ist mit dem Modell des fehlerfreien Reaktorsystems inkonsistent. Also ist der Fehler (Blockierung der Verbindungsleitung) aufgetreten.

- (b) fehlerfreier Fall      Fehlerfall
- |                                  |                            |
|----------------------------------|----------------------------|
| $\mathcal{Z}(1 0) = \{2, 5, 6\}$ | $\mathcal{Z}(1 0) = \{3\}$ |
| $\mathcal{Z}(2 1) = \{2, 3, 6\}$ | $\mathcal{Z}(2 1) = \{3\}$ |
| $\mathcal{Z}(3 2) = \{2, 3, 6\}$ | $\mathcal{Z}(3 2) = \{3\}$ |
| $\mathcal{Z}(4 3) = \{2, 3, 6\}$ | $\mathcal{Z}(4 3) = \{3\}$ |

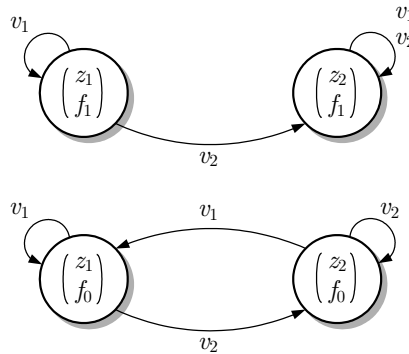
Es kann für das zweite E/A-Paar kein Fehler detektiert werden.

- (c) fehlerfreier Fall      Fehlerfall
- |                                |                            |
|--------------------------------|----------------------------|
| $\mathcal{Z}(1 0) = \{6, 9\}$  | $\mathcal{Z}(1 0) = \{6\}$ |
| $\mathcal{Z}(2 1) = \{6\}$     | $\mathcal{Z}(2 1) = \{3\}$ |
| $\mathcal{Z}(3 2) = \emptyset$ | $\mathcal{Z}(3 2) = \{3\}$ |
| $\mathcal{Z}(4 3) = \emptyset$ | $\mathcal{Z}(4 3) = \{3\}$ |

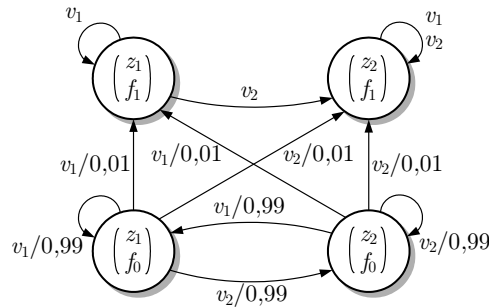
Schon bei  $k_e = 2$  ist das E/A-Paar mit dem Modell des fehlerfreien Systems inkonsistent, so dass ein Fehler detektiert wird. Die Messwerte sind mit dem Modell des Reaktorsystems bei blockiertem Verbindungsrohr konsistent.

### Aufgabe 16.6 Modellierung eines Batchreaktors für die Fehlerdiagnose

- Der untere Teil von Abb. A.75 zeigt den deterministischen Automaten, der den Reaktor im fehlerfreien Betrieb (Fehler  $f_0$ ) beschreibt. Durch die Eingaben  $v_1$  und  $v_2$  werden Zustandswechsel erzwungen. Andererseits bleibt der Reaktor im Zustand  $z_1$  bzw.  $z_2$ , wenn die Eingabe  $v_1$  die Pumpe eingeschaltet lässt (wobei die Sicherheitsabschaltung wirkt) bzw. die Eingabe  $v_2$  das Ventil weiterhin öffnet.
- Im Fehlerzustand kann der Reaktor nicht mehr vom Zustand  $z_2$  in den Zustand  $z_1$  übergehen, so dass die entsprechende Kante entfällt und die Schlinge am Zustand  $z_2$  jetzt auch für die Eingabe  $v_1$  gilt. Der Zustand des Reaktors ist jetzt durch den Füllstand und den Fehlerzustand beschrieben (Abb. A.75).



**Abb. A.75:** Beschreibung des Batchreaktors durch einen deterministischen Automaten



**Abb. A.76:** Erweiterung des Modells zum stochastischen Automaten

3. Die in Abb. A.75 für den fehlerfreien Betrieb gezeigten Zustandsübergänge treten jetzt nur noch mit 99%-iger Wahrscheinlichkeit ein, was in Abb. A.76 durch die Zahlenwerte hinter der Eingabe kenntlich gemacht ist. Mit 1%-iger Wahrscheinlichkeit treten Zustandsübergänge auf, bei denen sich nicht nur der Füllstand wie bisher, sondern zusätzlich der Fehlerzustand ändert. Der Automatengraph gilt für den Reaktor, der aus dem Fehlerzustand nicht wieder in den fehlerfreien Zustand zurückkehren kann.

## Anhang 2

### Fachwörter deutsch – englisch

In diesem Anhang sind die wichtigsten englischen und deutschen Begriffe der Automatisierungstechnik einander gegenübergestellt, wobei gleichzeitig auf die Seite verwiesen wird, auf der der deutsche Begriff eingeführt wird. Damit soll dem Leser der Zugriff auf die umfangreiche englischsprachige Literatur erleichtert werden.

#### Allgemeine Begriffe der Automatisierungstechnik

Aktor, 5	<i>actuator, actuating device</i>	Fehleridentifikation, 290	<i>fault identification</i>
Ausgang, 47	<i>output</i>	Fehlerlokalisierung, 290	<i>fault isolation</i>
Automatisierungspyramide, 36	<i>automation pyramid</i>	Fehlermodell, 317	<i>fault model</i>
Automatisierungstechnik, 3	<i>automation</i>	fehlertolerante Steuerung/Regelung, 328	<i>fault-tolerant control</i>
Arbeitspunkt, 112	<i>operation point</i>	geschlossener Wirkungskreis, 33	<i>closed-loop system</i>
Beobachtbarkeit, 155	<i>observability</i>	Güteforderung, 211	<i>performance specification</i>
Beobachter, 261	<i>observer</i>	Identifikation, 28	<i>identification</i>
Beobachtung, 30	<i>observation</i>	Industrielle Informationstechnik, 40	<i>industrial information technology</i>
Blockschaltbild, 55	<i>block diagram</i>	Kreisstruktur, 66	<i>loop structure</i>
Dekomposition, 64	<i>decomposition</i>	Kybernetik, 25	<i>cybernetics</i>
Diagnosesystem, 291	<i>diagnostic unit, diagnoser</i>	kompositionale Modellbildung, 116	<i>compositional modelling</i>
Eingang, 47	<i>input</i>	Mechatronik, 24	<i>mechatronics</i>
eingebettetes System, 23	<i>embedded system</i>	Mensch-Maschine-Kommunikation, 10	<i>human-machine interface</i>
Fehler, 288	<i>fault</i>	Modell, 26	<i>model</i>
Fehlerdiagnose, 31	<i>fault diagnosis</i>		
Fehlererkennung, 290	<i>fault detection</i>		

modellbasierte Diagnose, 292	<i>model-based diagnosis</i>	Sensor, 4	<i>sensor, measurement device</i>
Modellbildung, 27	<i>modelling</i>	Signal, 46	<i>signal</i>
Modellunsicherheit, Modellunbestimmtheit, 230	<i>model uncertainty</i>	Signalflussgraph, 62	<i>signal flow graph</i>
offene Wirkungskette, 33	<i>open-loop system</i>	Speicherprogrammierbare Steuerung (SPS), 40	<i>programmable logic controller (PLC)</i>
Parallelschaltung, 117	<i>parallel connection, parallel structure</i>	stark zusammenhängende Komponenten, 66	<i>strongly connected components</i>
Prozess, 46	<i>process</i>	Stellglied, 5	<i>actuator, actuating device</i>
Prozessinformatik, 43	<i>computer control</i>	Stellgröße, 20	<i>control input</i>
Prozessleitsystem, 10	<i>distributed control system, process control system</i>	Steuerbarkeit, 155	<i>controllability</i>
Prozessregelung, 43	<i>process control</i>	Steuerung (allg.), 32	<i>control</i>
Prozessüberwachung, 4	<i>process supervision</i>	Steuerung (in der offenen Wirkungskette), 33	<i>feedforward control</i>
reaktives System, 23	<i>reactive system</i>	Steuerung (diskret), 32	<i>logic control, sequential control, binary control</i>
rechnergestützter Entwurf, 260	<i>computer-aided design</i>	Störung, 26	<i>disturbance</i>
Redundanz, 294	<i>redundancy</i>	Struktur, 54	<i>structure</i>
Regelabweichung, 7	<i>control error</i>	System, 46	<i>system</i>
Regelkreis, 211	<i>control loop</i>	Systemtechnik, 40	<i>systems engineering</i>
Regelstrecke, 20	<i>plant, system to be controlled</i>	Totzeit, 244	<i>dead time</i>
Regelung, 33	<i>feedback control</i>	Verhalten, 123	<i>behaviour</i>
Regler, 72	<i>controller, regulator</i>	vernetztes Regelungs-system, 23	<i>networked control system</i>
Reglerentwurf, 241	<i>controller design</i>	Vorhersage, 123	<i>prediction</i>
Reglergesetz, 33	<i>control law</i>	Wirkungsrichtung, 56	<i>direction of action</i>
Reihenschaltung, 117	<i>series connection</i>	Zustand, 26	<i>state</i>
Rekonfiguration, 329	<i>reconfiguration</i>	Zustandsbeobachter, 30	<i>state observer</i>
Rückführschaltung, 118	<i>feedback connection</i>	Zustandsraum, 94	<i>state space</i>
Rückkopplung, 21	<i>feedback</i>	Zustandsraummodell, 350	<i>state space model</i>
Ruhelage, 182	<i>equilibrium state</i>	Zuverlässigkeit, 19	<i>reliability</i>
technische Sicherheit, 15	<i>safety</i>		



## Wertkontinuierliche Systeme

Anstiegszeit, 213	<i>rise time</i>	PID-Regler, 234	<i>PID controller, three-term controller</i>
Arbeitspunkt, 112	<i>operation point</i>	Regelgröße, 7	<i>controlled variable</i>
Ausgabegleichung, 91	<i>output equation</i>	Residuenauswertung, 297	<i>residual evaluation</i>
Ausgangsmatrix, 99	<i>output matrix</i>	Residuenberechnung, 297	<i>residual generation</i>
Beobachtbarkeitsmatrix, 163	<i>observability matrix</i>	Residuum, 296	<i>residual</i>
Beruhigungszeit, 213	<i>settling time</i>	robuste Regelung, 230	<i>robust control</i>
charakteristisches Polynom, 187	<i>characteristic polynomial</i>	Robustheit, 207	<i>robustness</i>
dezentrale Regelung, 8	<i>decentralised control</i>	Sollwert, 7	<i>set point, desired value</i>
Durchgangsmatrix, 99	<i>direct input-output matrix</i>	Sollwertfolge, 219	<i>set-point following</i>
Eigenbewegung, 126	<i>free motion</i>	Sprungfunktion, 141	<i>step function</i>
Eingangsmatrix, 99	<i>input matrix</i>	Stabilisierbarkeit, 209	<i>stabilisability</i>
Einstellfaktor, 255	<i>tuning factor</i>	Stabilität, 181	<i>stability</i>
Einstellregel, 243	<i>tuning rule</i>	statische Verstärkung, 142	<i>static reinforcement, DC gain</i>
erzwungene Bewegung, 126	<i>forced motion</i>	Steuerbarkeitsmatrix, 157	<i>controllability matrix</i>
freie Bewegung, 126	<i>free motion</i>	Störkompensation, 219	<i>disturbance attenuation</i>
Führungsgröße, 212	<i>command signal, reference signal</i>	Strukturgraph, 169	<i>structure graph</i>
Führungsübergangsfunktion, 212	<i>command step response</i>	Systemmatrix, 91	<i>system matrix</i>
Gegenkopplung, 250	<i>negative feedback</i>	Trajektorie, 94	<i>trajectory</i>
Gleichgewichtszustand, 100	<i>equilibrium</i>	Transitionsmatrix, 126	<i>transition matrix</i>
Kreisverstärkung, 222	<i>loop gain</i>	Vorfilter, 225	<i>prefilter</i>
Linearisierung, 111	<i>linearisation</i>	Übergangsfunktion, 141	<i>step response</i>
Ljapunowfunktion, 196	<i>Lyapunov function</i>	Überschwingweite, 213	<i>overshoot</i>
Mehrgrößensystem, 99	<i>multivariable system</i>	Verstärkungsfaktor, 100	<i>gain</i>
proportionale Rückführung, 236	<i>proportional feedback</i>	Zeitkonstante, 138	<i>time constant</i>
		Zustandsgleichung, 91	<i>state equation</i>
		Zustandsvariable, 94	<i>state variable</i>

## Ereignisdiskrete Systeme

Ablaufsteuerung, 459	<i>sequential control</i>	Lebendigkeit, 447	<i>liveliness</i>
aktivierte Transition, 391	<i>enabled transition</i>	Marke, 389	<i>token</i>
Ausgabefunktion, 350	<i>output function</i>	Markoveigenschaft, 385	<i>Markov property</i>
Ausgabealphabet, 339	<i>output alphabet</i>	Markovkette, 376	<i>Markov chain</i>
Ausgangsfolge, 335	<i>output sequence</i>	nichtdeterministischer Automat, 355	<i>nondeterministic automaton</i>
Automat, 344	<i>automaton</i>	Petrinetz, 389	<i>Petri net</i>
Automatengraph, 345	<i>automaton graph, state transition graph</i>	Platz, 389	<i>place</i>
Automatentabelle, 345	<i>automaton table, state transition table</i>	Takt, 347	<i>clock</i>
Bedingungs/Ereignis-Netz, 393	<i>condition-event net</i>	Transition, 389	<i>transition</i>
deterministischer Automat, 344	<i>deterministic automaton</i>	Stelle, 389	<i>place</i>
Eingabealphabet, 339	<i>input alphabet</i>	stochastischer Automat, 372	<i>stochastic automaton</i>
Eingangsfolge, 335	<i>input sequence</i>	Synchronisationsgraph, 398	<i>timed event graph</i>
endlicher Automat, 344	<i>finite automaton, finite-state machine</i>	Verhaltensrelation, 367	<i>behavioural relation</i>
Ereignis, 336	<i>event</i>	Verifikation, 480	<i>verification</i>
Ereignisfolge, 336	<i>event sequence</i>	Verklemmung, 447	<i>deadlock</i>
Erreichbarkeit, 429	<i>reachability</i>	Wahrscheinlichkeit, 373	<i>probability</i>
Erreichbarkeitsgraph, 447	<i>reachability graph</i>	Zeitplansteuerung, 465	<i>time scheduled control</i>
Funktionsplan, 474	<i>function chart</i>	Zustandsalphabet, 339	<i>state alphabet, state space</i>
hybrider Automat, 558	<i>hybrid automaton</i>	Zustandsfolge, 335	<i>state sequence</i>
Invariante, 448	<i>invariant</i>	Zustandsmaschine, 344	<i>state machine</i>
Konflikt, 394	<i>conflict</i>	Zustandsmenge, 339	<i>state set</i>
Kontakt, 392	<i>contact</i>	Zustandsübergangsfunktion, 344	<i>state transition function</i>
		Zustandsübergangsrelation, 358	<i>state transition relation</i>

# Sachwortverzeichnis

- Ablaufsteuerung, 459, 462
  - A. mit Zähler, 469
  - zeitgeführte A., 466
- Abstandsregeltempomat, 569
- Abstraktion, 554
- Aggregation, 64
- aktivierte Transition, 391
- Aktor, 5
- Alternative, 394
- Anstiegszeit, 213
- Antiblockiersystem, 15, 207
- Anweisungsliste, 474
- Arbeitspunkt, 112, 182
- Architektur, 25
- Aufspaltung, 395
- Ausgabe, 339
- Ausgabealphabet, 339
- Ausgabefunktion, 350
- Ausgabegleichung, 91
- Ausgang, 47, 339
- Ausgangsereignis, 336
- Ausgangsfolge, 335
- Ausgangsgröße, 26
- Ausgangsmatrix, 99
- Ausgangstransition, 391
- Ausgangsvektor, 99
- ausgangsverbunden, 171
- Automat, 344
  - abstrakter A., 425
  - deterministischer A., 344
  - E/A-A., 349
  - endlicher A., 344
  - hybrider A., 558
  - irreduzibler A., 431
  - nichtdeterministischer A., 355
  - reduzierter A., 481
  - $\Sigma$ -Automat, 425
  - Standardautomat, 425
  - stochastischer A., 372
- Automatengraph, 345
- Automatentabelle, 345
- automatische Steuerung, 40
- Automatisierung, 5
  - Notwendigkeit der A., 18
- Automatisierungsaufgaben, 25
  - Lösungsweg für A., 41
- Automatisierungseinrichtung, 19
- Automatisierungsgerät, 40
- Automatisierungshierarchie, 36
- Automatisierungspyramide, 36
- Automatisierungssystem, 20
- Automatisierungstechnik, 3
  - Aufgaben der A., 25
  - Methoden der A., 40
  - Ziele der A., 18
- autonomes Fahren, 569
- Batchprozess, 8
- Bayesformel, 541
- Bayesnetz, 540
- Bedingungs/Ereignis-Netz, 393
- Begegnung, 395
- Beispiele**, xvii, 6
- Beispiel
  - Abfüllanlage, 59, 67, 79
  - aktive Federung, 22
  - Ampelsteuerung, 461, 470
  - Antiblockiersystem, 15, 107, 114, 207, 238, 423
  - Batchprozess, 8, 111, 337, 400, 439, 452, 478, 531, 534, 550, 552, 561
  - Batchreaktor, 49, 51, 75
  - Digitalkamera, 404
  - Eisenbahn, 404
  - Elbehochwasser, 59
  - Elektroenergienetz, 13, 70
  - fahrerloses Fahrzeug, 561

- Fahrkartenautomat, 24  
 Fensterheber, 179, 299  
 Fertigungszelle, 38, 363, 366, 381, 397, 403, 414, 438, 451, 499, 508, 512  
 Flugregelung, 16  
 gekoppelte Reaktoren, 272, 535  
 Getränkeautomat, 462  
 Gleichstrommotor, 58, 65, 86, 90, 94, 96, 98, 100, 215, 222, 228, 234, 284  
 Heizung, 24  
 Industrieofen, 158, 166, 172, 270, 275, 284  
 intelligenter Sensor, 313  
 invertiertes Pendel, 73, 102, 193, 206, 208, 236  
 Kühlsystem, 542  
 Kartentelefon, 372  
 Kopiergerät, 24  
 Kraftfahrzeug, 6, 14, 61, 114, 129, 147, 150, 178, 194, 231, 288, 469  
 Paketlager, 49, 51  
 Pendel, 192, 198, 303  
 Personenaufzug, 422, 483, 496  
 pH-Sensor, 308  
 Qualitätskontrolle, 544  
 Rührkesselreaktor, 7, 140, 206, 246, 303  
 Radioteleskop, 110, 218  
 Raumbeleuchtung, 372, 536  
 Raumtemperaturregelung, 12, 79  
 Rechnerkommunikation, 411  
 Regensensor, 368, 385, 446  
 Roboter, 11, 78  
 Rolltreppe, 412, 418, 466  
 Satellit, 180  
 Schiffsschleuse, 496  
 Screening-Anlage, 406  
 Servoventil, 320, 327  
 Sicherheitsschleuse, 493  
 Stanze, 348, 360, 380, 442, 499  
 Temperaturregelung, 256  
 Transportsystem, 11  
 Treppenhausbeleuchtung, 470  
 U-Bahn, 471, 555  
 Unterwasserfahrzeug, 103, 114, 194  
 Verbrennungsmotor, 14, 62, 406  
 Vorratsbehälter, 424  
 Wärmeübertrager, 114, 152, 189  
 Waschautomat, 5, 24  
 Beobachtbarkeit, 155, 156, 163  
   strukturelle B., 170  
 Beobachtbarkeitskriterium, 163  
 Beobachtbarkeitsmatrix, 163  
   gramsche B., 166  
 Beobachter, 261, 501, 525  
   dedizierter B., 306  
 Beobachtung, 30, 261  
   B. deterministischer Automaten, 511  
   B. diskreter Systeme, 501  
   B. kontinuierlicher Systeme, 261  
   B. nichtdeterministischer Automaten, 502  
   B. stochastischer Automaten, 515  
 Beobachtungsmatrix, 99  
 Beobachtungsnormalform, 269  
 Beruhigungszeit, 213  
 Beschreibung diskreter Systeme, 333  
 Beschreibung kontinuierlicher Systeme, 83  
 Betriebsleitebene, 37  
 Bewegungsgleichung, 126  
 Blockschaltbild, 55  
  
 CAD, 40  
 CAE, 40  
 charakteristisches Polynom, 187  
 CIM, 11  
*computer-aided design*, 40  
*computer-aided engineering*, 40  
*computer-integrated manufacturing*, 11  
*Cyber-physical system*, 23  
  
 Datenbank, 43  
 Dekomposition, 64  
 deterministischer Automat, 344  
 dezentrale Regelung, 8  
 Diagnose, 31  
   D. diskreter Systeme, 525  
   D. kontinuierlicher Systeme, 287  
   D. nichtdeterministischer Automaten, 526  
   D. stochastischer Automaten, 538  
   konsistenzbasierte D., 296  
   modellbasierte D., 291  
   Sensordiagnose, 305  
   signalbasierte D., 290  
 Diagnosesystem, 291  
 Dimensionalität, 53  
 dominierender Eigenwert, 138  
 Durchgangsmatrix, 99  
 Dynamik, 53  
 dynamische Ordnung, 92  
  
 E/A-Automat, 349

- E/A-Paar, 502
  - Konsistenz des E. mit einem Automaten, 513
- E/A-Verhalten, 127
- Echtzeitprogrammierung, 42
- Eigenbewegung, 126
- Eigenvorgang, 137
- Eigenwert
  - Beobachtereigenwert, 269
  - dominierender E., 138
  - E. der Systemmatrix, 136
- Eingabe, 339
- Eingabealphabet, 339
- Eingang, 47, 339
- Eingangs-Ausgangsrelation, 47
- Eingangs-Ausgangspaar, 502
- Eingangs-Ausgangsverhalten, 127
- Eingangsereignis, 336
- Eingangsfolge, 335
- Eingangsmatrix, 99
- Eingangssignal, 26
- Eingangstransition, 391
- Eingangsvektor, 99
- eingangsverbunden, 171
- eingebettetes System, 23
- Einstellfaktor, 255
- Einstellregel, 243
  - E. für robuste I-Regler, 254
- endlicher Automat, 344
- Ereignis, 336
  - spontanes E., 351
  - steuerbares E., 354, 402
- Ereignisfolge, 336
- ereignisgetrieben, 340
- Erreichbarkeit, 429
- Erreichbarkeitanalyse, 429
- Erreichbarkeitsgraph, 447
- Erreichbarkeitsmenge, 447
- erzwungene Bewegung, 126
- Eulerverfahren, 145
- Fahrerassistenzsystem, 569
- FDI, *siehe* Diagnose
- Fehler, 26, 288
  - additiver F., 294
  - Aktorfehler, 289
  - externer F., 288
  - interner F., 288
  - multiplikativer F., 294
  - Sensorfehler, 289
  - Systemfehler, 289
- Fehlerdetektion, *siehe* Fehlererkennung
- Fehlerdiagnose, 31, *siehe* Diagnose
- Fehlererkennung, 290, 528
- Fehleridentifikation, 290, 529
  - F. mit Beobachter, 316
- Fehlerkandidat, 296, 526
- Fehlerlokalisierung, 290
- Fehlermodell, 317, 527
- Fehlerraum, 526
- fehlertolerante Steuerung, 328, 483
- Feldebene, 36
- Fernautomatisierung, 43
- Flugregelung, 16
- freie Bewegung, 126
- Frequenzbereich, 209
- Führungsgröße, 7, 212
  - impulsförmige F., 219
  - sprungförmige F., 220
- Führungsübergangsfunktion, 212
- Fundamentalmatrix, 126
- Funktionalmatrix, 113
- Funktionsplan, 474
- Gebäudeautomatisierung, 12
- Gegenkopplung, 250
- geschlossener Wirkungskreis, 33
- Gleichgewichtszustand, 100, 182
- Gleitbeiwert, 108
- global asymptotisch stabil, 197
- Gradient, 196
- GRAF CET, 425
- Grundstruktur automatisierter Systeme, 19
- Güteforderung, 211
- Haftbeiwert, 108
- harmonische Balance, 209
- Hidden Markov model*, 425
- HiGraph*, 425
- Hilfsregelgröße, 239
- Hilfsstellgröße, 239
- HURWITZ, A., 187
- Hurwitzdeterminante, 188
- Hurwitzkriterium, 188
- hybrider Automat, 558
- hybrides System, 555
- I-Regler, 225
- Identifikation, 28

- Industrie 4.0, 11
- Industrielle Informationstechnik, 40
- Informatik, 42
- Informationstechnik, 17
- Injektor, 557
- Integrationsverfahren, 145
  - Eulerverfahren, 145
  - Runge-Kutta-Verfahren, 146
- Invariante, 448
- Invarianzprinzip, 197
- Inzidenzmatrix, 396
  
- Jacobimatrix, 113
  
- KALMAN, R. E., 157
- Kalmanfilter, 30
- Kalmanzerlegung, 178
- Kaskadenregelung, 239
- Kausalität, 53, 393
- Kennwertermittlung, 151
- Kommunikationsnetz, 43
- Kommunikationstechnik, 17
- Komplexität, 42
- komponentenorientierte Modellbildung, *siehe*
  - kompositionale Modellbildung
- kompositionale Modellbildung, 64, 116
  - Blockschaltbild, 56
  - k. M. kontinuierlicher Systeme, 116
  - k. M. mit Automaten, 407
- Kondensation, 67
- Konflikt, 394
- Konsistenz, 513
- Kontakt, 392
- Kontaktplan, 474
- Koordination, 38
- Kopplung
  - schwache K., 69
  - strenge K., 66
- Kopplungsanalyse, 66
- Kopplungsgraph, 66
- Kraftschlussbeanspruchung, 108
- Kreisstruktur, 66
- Kreisverstärkung, 222
- Kybernetik, 25
  - Technische K., 25
  
- LASALLE, 197
- Lebendigkeit, 447
- leeres Symbol, 340, 351
  
- leiten, 10
- Linearisierung, 111
- Linearität, 54
- Lipschitzbedingung, 144
- LJAPUNOW, A. M., 184
- Ljapunowfunktion, 196
- Ljapunowgleichung, 201
- LUENBERGER, D. G., 265
- Luenbergerbeobachter, 263
  
- Marke, 389, 391
- Markierung, 389
  - tote M., 447
- Markierungsvektor, 396
- Markoveigenschaft, 385
- Markovkette, 376
- Materialflusssystem, 43
- Matrixexponentialfunktion, 124
- Mealy-Automat, 352
- Mechanisierung, 5
- Mechatronik, 24
- Medizintechnik, 44
- Mehrgrößensystem, 99
- Mensch-Maschine-Kommunikation, 10
- Messfehler, 228
- Messglied, 57
- Messstelle, *siehe* Messglied
- Messung, 261
- Mode, 137
- Model checking, 500
- Modell, 26
- modellbasierte Diagnose, 292
- Modellbildung, 27
  - hierarchische M., 65
  - komponentenorientierte M., *siehe* kompositionale Modellbildung
  - kompositionale M., 64, 407
  - M. diskreter Systeme, 340
  - M. kontinuierlicher Systeme, 83
  - prozessorientierte M., 393
- Modellierung, *siehe* Modellbildung
- Modellunbestimmtheit, 230
- Modellunsicherheit, *siehe* Modellunbestimmtheit
- Moore-Automat, 352
  
- Nachrichtentechnik, 43
- Nachstellzeit, 235
- NICHOLS, N. B., 243
- Nichtdeterminismus, 393

- nichtdeterministischer Automat, 355  
Notbremsassistent, 569
- offene Wirkungskette, 33  
Onboard-Diagnose, 288
- P-Regler, 215  
Parallelschaltung, 117  
Petrinetz, 389  
    Free-choice-Netz, 395  
    Steuerungstechnische Interpretation d. P., 400  
PID-Regler, 234  
Planung, 29  
Platz, 389, *siehe* Stelle  
Poststelle, 391  
Prädiktion, *siehe* Vorhersage  
Prästelle, 391  
Produktionsleitebene, 37  
Produktivitätsanforderung, 458  
Prozess, 46  
    nebenläufige Prozesse, 389  
    stochastischer P., 372  
    zyklischer P., 431  
Prozessüberwachung, 4  
Prozessautomatisierung, 6  
Prozessdiagnose, *siehe* Diagnose  
Prozessinformatik, 43  
Prozessleitebene, 36  
Prozessleitsystem, 10  
Prozessleittechnik, 10  
Prozessregelung, 43
- Quantisierer, 557
- R&I-Fließbild, 8  
Radlast, 107  
reaktives System, 23  
rechnergestützter Reglerentwurf, 260  
Rechnernetz, 17  
Rechnersystem, 43  
Redundanz, 294  
    analytische R., 294, 305, 313  
    physikalische R., 305  
reduzierter Automat, 481  
Regelabweichung, 7  
Regleinrichtung, *siehe* Regler  
Regelgüte  
    erreichbare R., 228  
Regelgröße, 7  
Regelgrößen-Stellgrößen-Zuordnung, 241  
Regelgüte, 212  
Regelkreis, 33, 211  
    Eigenschaften d. R., 219  
    Standardregelkreis, 214  
Regelkreisstruktur, 241  
regeln und steuern, 32  
Regelstrecke, 20, 72  
Regelung, 33, 72, 458  
    diskrete R., 32  
    ereignisbasierte R., 32  
    R. mit Zustandsbeobachter, 35, 283  
    regeln und steuern, 32  
Regelungsaufgabe, 211  
Regelungstechnik, 32  
Regelungstheorie, 40  
Regler, 20, 33, 72  
    dezentraler R., 8  
    diskreter R., 32  
    I-Regler, 225  
    P-Regler, 215  
    PID-Regler, 234  
    robuster R., 207, 230  
Reglerentwurf, 241  
    R. für Automaten, 480  
    R. für Petrinetze, 488  
Reglergesetz, 33  
Reglerkomponenten, 237  
Reglerparameter, 242  
Reglerstruktur, 241  
Reglertyp, 234  
Reihenschaltung, 117, 413  
Rekonfiguration, 329  
Repräsentation, 85  
Residuenauswertung, 297  
Residuenberechnung, 297  
Residuum, 293, 296  
Rezept, 478  
Robotersteuerung, 11  
robuste Regelung, 230, 250  
robuster Regler, 207  
Robustheit, 207, 230  
Rückführautomat, 417  
Rückführschaltung, 118  
Rückkopplung, 66  
Rückkopplungsprinzip, 21  
rückwirkendes System, 23  
Ruhelage, 182  
Runge-Kutte-Verfahren, 146

- S-Invariante, 450
- s-Rang, 171
- Schaltnetz, 460
- Schaltregel, 344
- Schaltvektor, 396
- Schlupf, 108
- Screening, 406
- Senke, 447
- Sensor, 4
- Sensorfehler, 305
- Sicherheit, 15
- Sicherheitsanforderung, 458
- Signal, 46
- Signalanzeige, 57
- Signalflussgraph, 62
- Signalgenerator, 57
- Signalverzweigung, 56
- Sollwert, 7
- Sollwertfolge, 219
- Speicherprogrammierbare Steuerung, 40, 473
- Spezifikation, 458
- Sprungfunktion, 141
- SPS, *siehe* Speicherprogrammierbare Steuerung
- Spurhalteassistent, 569
- Störgrößenaufschaltung, 78, 239
- Störkompensation, 219
- Stabilisierbarkeit, 177, 209
- Stabilität, 138, 181
  - asymptotische S., 183
  - globale asymptotische S., 197
  - Ljapunowstabilität, 183
  - robuste S., 207
  - S. d. Regelkreises, 203
  - S. nichtlinearer Systeme, 190
  - Zustandsstabilität, 183
- Standardautomat, 425
- Standardregelkreis, 214
- stark verbunden, *siehe* stark zusammenhängend
- stark zusammenhängend
  - s. z. Komponenten, 66
  - s. z. Zustände, 431
- Stateflow, 425
- Statemate, 425
- statische Verstärkung, 142
- Stelle, 389
  - Poststelle, 391
  - Prästelle, 391
- Stellglied, 5
- Stellgröße, 7
- Stellgrößenbeschränkung, 229
- Stellgröße, 26
- Steuerbarkeit, 155
  - S. von Automaten, 430
  - S. von deterministischen Automaten, 433
  - S. von kontinuierlichen Systemen, 156
  - S. von linearen Systemen, 157
  - S. von nichtdeterministischen Automaten, 438
  - strukturelle S., 170
- Steuerbarkeitskriterium, 157
- Steuerbarkeitsmatrix, 157
  - gramsche S., 158
- Steuereinrichtung, 72
- Steuergerät, 20
- Steuerkreis, 459
- Steuermatrix, 99
- Steuerstrecke, 20, 72
- Steuerung, 4, 20, 72
  - diskrete S., 33, 458
  - fehlertolerante S., 483
  - Realisierung d. S., 472
  - regeln und steuern, 32
  - S. im geschlossenen Wirkungskreis, 33, 72, 457
  - S. in der offenen Wirkungskette, 33, 72
  - Verriegelungssteuerung, 500
- Steuerung (allg.), 32
- Steuerungsaufgabe, 457
- Steuerungsentwurf, 475
  - S. für Petrinetze, 488
- Steuerungsgesetz, 34, *siehe* Reglergesetz
- Steuerungsstruktur, 55
- Steuerungstechnik, 32, 40
- Steuerungstheorie, 40
- stochastischer Automat, 372
- stochastischer Prozess, 372
- Störgröße, 7, 26
  - impulsförmige S., 219
  - sprungförmige S., 220
- Störübergangsfunktion, 213
- Störung (im Sinne eines Fehlverhaltens), 289
- Strecke, *siehe* Regelstrecke, Steuerstrecke
- Struktur, 54
  - Steuerungsstruktur, 55
- strukturelle Analyse, 168
- strukturelle Beobachtbarkeit, 170
- strukturelle Steuerbarkeit, 170
- struktureller Rang, 171
- Strukturgraph, 169
- Summationsstelle, 56



- SYLVESTER, J. J., 143  
Symbol  
  leeres S., 340  
Synchronisation, 395  
Synchronisationsgraph, 398  
System, 46, 64  
  affines S., 114  
  autonomes S., 93  
  deterministisches S., 355  
  dissipatives S., 200  
  ereignisdiskretes S., 47, 553  
  gesteuertes S., 52  
  hybrides S., 555  
  kontinuierliches S., 47, 553  
  kooperierendes S., 13  
  lineares S., 54, 84  
  nichtdeterministisches S., 54, 355  
  nichtlineares S., 102  
  rückwirkendes S., 23  
  sprungfähiges S., 143  
  stückweise lineares S., 561  
  technisch realisierbares S., 85  
  zeitinvariantes S., 53  
Systemfunktion, 47  
Systemmatrix, 91, 99  
Systemoperator, 47  
Systemordnung, 92  
Systemstruktur, 54  
Systemtechnik, 40  
Systemverhalten, *siehe* Verhalten  
Szenarienanalyse, 281  
  
T-Invariante, 449  
Takt, 347, 354  
Technische Kybernetik, 25  
technische Sicherheit, 15  
Temperaturregelung, 246, 256  
Totzeit, 244  
Trajektorie, 94  
Transition, 389  
  Ausgangstransition, 391  
  Eingangstransition, 391  
  steuerbare T., 402  
  tote T., 447  
Transitionsmatrix, 126  
Transitionsvektor, 396  
Trellis-Diagramm, 436  
Übergangsfunktion, 141  
Übergangsmatrix, 126  
Überschwingweite, 213  
Überschwingzeit, 213  
Übertragungsfunktion, 209  
Überwachung, 4  
Unternehmensleitebene, 37  
  
Vektornorm, 183  
Verhalten, 123  
  E/A-Verhalten, 127  
  V. deterministischer Automaten, 428  
  V. kontinuierlicher Systeme, 123  
  V. linearer Systeme, 124  
  V. nichtdeterministischer Automaten, 434  
  V. nichtlinearer Systeme, 144  
  V. stochastischer Automaten, 441  
  V. von Petrinetzen, 446  
Verhaltensrelation, 367  
Verifikation, 480  
Verklemmung, 447  
Verknüpfungssteuerung, 458, 460  
vernetztes Regelungssystem, 23  
Verriegelungssteuerung, 500  
Verstärkung, *siehe* statische Verstärkung  
Verstärkungsfaktor, 100  
Vorfilter, 225  
Vorhaltezeit, 235  
Vorhersage, 28, 123, 281  
Vorsteuerung, 35, 78  
  
Wahrscheinlichkeit, 373  
Wahrscheinlichkeitsautomat, 376  
Weiterschaltbedingung, 465  
Werkstattdiagnose, 288  
WIENER, N., 25  
Wirkungsrichtung, 56  
Wirkungsschema, 55  
  
Zeitinvarianz, 53  
Zeitkonstante, 138, 151  
Zeitplansteuerung, 465  
ZIEGLER/NICHOLS-Einstellregeln, 243  
Zustand, 26, 339  
  absorbierender Z., 432  
  rekurrenter Z., 431  
  transienter Z., 432  
  Z. eines Petrinetzes, 393  
  Z. ereignisdiskreter Systeme, 351  
  Z. kontinuierlicher Systeme, 94

- Z. nichtdeterministischer Automaten, 360
- Zustandsalphabet, 339
- Zustandsbeobachter, 30
- Zustandsbeobachtung, *siehe* Beobachtung
- Zustandsereignis, 336
- Zustandsfolge, 335
- Zustandsgleichung, 91
  - Lösung der Z., 124
- Zustandsgröße, 94
- Zustandsmaschine, 344, 400
- Zustandsmenge, 339
- Zustandsraum, 94, 393
- Zustandsraummodell, 50, 350
  - Kalmanzerlegung d. Z., 178
  - lineares Z., 90
  - nichtlineares Z., 102
- Zustandsschätzung, 30
- Zustandsstabilität, 183
- Zustandstrajektorie, 336
- Zustandsüberföhrungsfunktion, 344
- Zustandsübergangsfunktion, 344
- Zustandsübergangsrelation, 358
- Zustandsübergangswahrscheinlichkeit, 374
- Zustandsvariable, 94
- Zustandsvektor, 94, 99
- Zustandswahrscheinlichkeitsverteilung, 373
- Zuverlässigkeit, 19
- zyklischer Prozess, 431