

Lehrstuhl für Elektrische Antriebssysteme
Technische Universität München

Nichtlineare modellbasierte prädiktive Regelung auf Basis lernfähiger Zustandsraummodelle

Martin Rau

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Alexander W. Koch

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing., Dr.-Ing. h.c. Dierk Schröder
2. Univ.-Prof. Dr.-Ing. Dirk Abel, Rheinisch-Westfälische
Technische Hochschule Aachen

Die Dissertation wurde am 15.01.2003 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 30.05.2003 angenommen.

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als Wissenschaftlicher Assistent am Lehrstuhl für Elektrische Antriebssysteme der Technischen Universität München.

Mein besonderer Dank gilt dem Leiter des Lehrstuhls, Herrn Prof. Dr.–Ing. Dr.–Ing. h.c. Dierk Schröder, für die Betreuung dieser Arbeit sowie für die fortwährende und anregende Unterstützung, die einen großen Teil zum Gelingen beigetragen hat. Er gab mir die Möglichkeit, auf einem sehr interessanten Forschungsgebiet unter hervorragenden Arbeitsbedingungen tätig zu sein.

Für die Übernahme des Korreferats und das entgegengebrachte Interesse an dieser Arbeit danke ich Herrn Prof. Dr.–Ing. Dirk Abel, für die Übernahme des Prüfungsvorsitzes gebührt mein Dank Herrn Prof. Dr.–Ing. Alexander W. Koch.

Mein Dank gilt auch allen Kollegen und Kolleginnen des Lehrstuhls, die durch das gute Betriebsklima eine immer angenehme und fruchtbare Arbeitsumgebung geschaffen haben. Besonders hervorheben möchte ich zahlreiche offene Diskussionen mit Herrn Dipl.–Ing. Matthias Feiler, Frau Dipl.–Ing. Anne Angermann und Herrn Dipl.–Ing. Christian Hintz.

Nicht vergessen will ich alle Diplomanden und wiss. Hilfskräfte, die stets engagiert zum Gelingen der Arbeit beigetragen haben. Für die Erstellung zahlreicher Programme zur prädiktiven Regelung danke ich besonders Herrn Dipl.–Ing. Hans Schuster und Herrn Dipl.–Ing. Franz Berchtenbreiter.

Nicht zuletzt bedanke ich mich bei meiner Frau Eva für die uneingeschränkte Unterstützung und Geduld, die mir immer ein großer Rückhalt waren.

Meinen Eltern gebührt mein besonderer Dank für die langjährige Förderung und Ermöglichung des Studiums. Ihnen widme ich diese Arbeit.

Waakirchen, 18.07.2003

Martin Rau

Kurzzusammenfassung

Diese Dissertation liefert einen Beitrag zur nichtlinearen Systemidentifikation und zum Entwurf einer nichtlinearen prädiktiven Regelung im Zustandsraum. Nichtlinearitäten werden innerhalb eines Beobachters durch neuronale Netze approximiert, die durch stabile Lerngesetze online trainiert werden. Das Ergebnis ist die Schätzung nichtlinearer Kennlinien und nicht messbarer Zustandsgrößen. Der Entwurf der darauf aufbauenden prädiktiven Regelung erfolgt im Zustandsraum durch Linearisierung der Nichtlinearität um die Referenztrajektorie. Unter Berücksichtigung von Begrenzungen ergibt sich ein quadratisches Programm als Optimierungsaufgabe, das online gelöst wird. Für die Regelung ist Stabilität nach Lyapunov gewährleistet. Neben den theoretischen Ableitungen erfolgt die simulative und praktische Validierung von Identifikation und Regelung an einem Zweimassensystem, einer kontinuierlichen Fertigungsanlage und einem Radioteleskop.

Abstract

The objective of this thesis is the identification and model predictive control design for nonlinear systems with a state space approach. Nonlinearities are approximated by neural networks included in a dynamic observer. The training of the networks is guaranteed stable. The result is the approximation of nonlinear characteristics and non measurable state variables. The predictive control design is based on a state space model and the linearization of the nonlinearity around the reference trajectory. The consideration of bounds in the control law results in a quadratic program, which is solved online. The controllers's stability is guaranteed in accordance with Lyapunov's theory. Besides the necessary theory, simulations and practical implementations of identification and control are shown for a two-mass system, a continuous processing plant and a radio telescope.

Inhaltsverzeichnis

1	Einführung	1
1.1	Allgemeines	1
1.2	Motivation und Ziel der Arbeit	3
2	Lineare Modellbildung	5
2.1	Modellbildung und Identifikation	5
2.1.1	Modellbildung	5
2.1.2	Identifikation	7
2.2	Lineare Prozessmodelle	8
2.2.1	Lineare Ein- Ausgangsmodelle	11
2.2.2	Lineare Differenzgleichung	13
2.2.3	Faltungssumme	14
2.2.4	Methode der kleinsten Quadrate	16
2.2.5	Modelle zur Fehlerminimierung	19
3	Nichtlineare Modellbildung	22
3.1	Klassifikation nichtlinearer dynamischer Systeme	23
3.1.1	Blockorientierte nichtlineare Modelle	23
3.1.2	Allgemeine nichtlineare Modelle	24
3.1.3	Zustandsdarstellung mit isolierter Nichtlinearität	24
3.2	Approximation nichtlinearer Funktionen	27
3.2.1	Methoden der Funktionsapproximation	28
3.2.2	Kriterien zur Beurteilung künstlicher neuronaler Netze	29
3.2.3	Funktionsapproximation mit lokalen Basisfunktionen	30
3.2.4	Radial Basis Function (RBF) Netz	32
3.2.5	General Regression Neural Network (GRNN)	33
3.2.6	Lerngesetz	35
3.2.6.1	Stabilität nach Lyapunov	36
3.2.6.2	Asymptotische Stabilität und Parameterkonvergenz	37
3.2.6.3	Automatische Einstellung der Lernschrittweite	38

3.2.6.4	Lernstruktur und Fehlermodelle	41
3.2.6.5	Approximation einer eindimensionalen Nichtlinearität	46
3.3	Lernfähiger Beobachter	49
3.3.1	Strecken mit isolierter Nichtlinearität	49
3.3.2	Beobachterentwurf bei messbarem Eingangsraum	50
3.3.3	Beobachterentwurf bei nicht messbarem Eingangsraum	53
3.3.4	Identifikation mehrerer Nichtlinearitäten	56
3.4	Experimentelle Validierung	59
3.5	Zusammenfassung	63
4	Selbstlernender Regler für periodische Nichtlinearitäten	64
4.1	Einführung	64
4.2	Entwurf des selbstlernenden Reglers	65
4.2.1	Streckenbeschreibung	65
4.2.2	Selbstlernende Kompensation	66
4.2.2.1	Approximation des Kompensationssignals durch ein neuronales Netz . . .	69
4.2.2.2	Stabiles Lerngesetz	71
4.2.3	Zusammenfassung	73
4.3	Elektrischer Antrieb mit Stellglied	74
4.3.1	Kompensation bei konstantem Sollwert	75
4.3.2	Kompensation bei Arbeitspunktwechseln	78
4.4	Unrundheitskompensation in kontinuierlichen Fertigungsanlagen	80
4.4.1	Modellierung	80
4.4.2	Unrundheitskompensation durch den selbstlernenden Regler	86
5	Modellgestützte prädiktive Regelungen	93
5.1	Prinzipielle Funktionsweise	94
5.2	Prädiktive Regelverfahren auf Basis linearer Prozessmodelle	99
5.2.1	Dynamic Matrix Control	100
5.2.2	Generalized Predictive Control	101
5.3	Nichtlineare prädiktive Regelungen	103
6	Prädiktive Regelung mit nichtlinearen Zustandsmodellen	105
6.1	Abhängigkeit der Nichtlinearität vom Prozessausgang	107
6.1.1	Linearisierung nullter Ordnung	107

6.1.2	Linearisierung erster Ordnung	110
6.2	Abhängigkeit der Nichtlinearität vom Zustandsvektor	112
6.2.1	Sollverlauf des Zustandsvektors	115
6.2.2	Aktueller Zustand als Referenzpunkt	117
6.2.3	Solltrajektorie bei Prozessen mit Relativgrad eins	117
6.2.4	Verwendung des Stellgrößenvektors des vorhergehenden Zeitschrittes . . .	120
6.3	Regelgesetz ohne Begrenzungen	122
6.3.1	Stationäre Genauigkeit bei Störgrößen	125
6.4	Prädiktive Regelung mit Integralanteil	126
6.4.1	Regelgesetz ohne Begrenzungen	129
6.5	Optimierung unter Berücksichtigung von Beschränkungen	130
6.5.1	Stellgrößenbeschränkungen	133
6.5.2	Stellgrößenänderungsbeschränkungen	135
6.5.3	Zustandsgrößenbeschränkungen	135
6.5.4	Ausgangsgrößenbeschränkungen	137
6.5.5	Resultierendes Optimierungsproblem	138
6.6	Erweiterung auf mehrere Nichtlinearitäten	139
7	Stabilität prädiktiver Regelungen	142
7.1	Stabilitätsbeweis durch monotone Abnahme der Kostenfunktion	143
7.2	Endzustandsbeschränkung	145
7.3	Quasi-Infinite Horizon Regelungskonzept	146
7.4	Stabilität der entwickelten Regelung	148
8	Experimentelle Validierung der prädiktiven Regelung	152
8.1	Prädiktive Regelung ohne Begrenzungen	152
8.1.1	MPC ohne Integralanteil	152
8.1.2	MPC mit Integralanteil	155
8.2	Prädiktive Regelung mit Begrenzungen	157
8.2.1	Stellgrößenbegrenzungen	157
8.2.2	Zustandsbegrenzungen	159
8.2.3	Regelgrößenbegrenzungen	160
8.3	Untersuchung der Einstellparameter	162
8.4	Robustheitsuntersuchung	165
8.5	Reglervergleich	167

8.6	Prädiktive Regelung mit lernfähigem Beobachter	170
8.6.1	Besonderheit des lernfähigen Beobachters mit prädiktivem Regler	171
8.6.2	Ergebnisse	172
9	Mehrgrößenprozesse	176
9.1	Erweiterung auf Mehrgrößenprozesse	176
9.2	Anwendung der prädiktiven Regelung auf ein Radioteleskop	179
9.2.1	Beschreibung des Systems und Modellierung	179
9.2.2	Gegenüberstellung von prädiktivem Regler und PI-Regler am Radioteleskop	183
10	Zusammenfassung und Ausblick	189
A	Prüfstand Zweimassensystem	192
A.1	Prüfstand: Elektrische Daten	192
A.2	Anlagendaten	194
A.3	Messtechnische Bestimmung der Reibkennlinie	194
A.4	Lineare Parameteridentifikation	195
A.5	Systembeschreibung im Zustandsraum	196
B	Daten und Aufbau der MODAM	199
B.1	Allgemeine Daten	199
B.2	Daten zum Bahnsystem und Papier	199
B.3	Daten des Wickler-Antriebssystems	200
B.4	Daten des Zweimassenschwingers	200
B.5	Daten des Regelsystems dSpace	200
B.6	Teilkomponenten des Versuchsstandes	200
	Bezeichnungen	203
	Literaturverzeichnis	207

1 Einführung

1.1 Allgemeines

Eine wichtige Aufgabe der Regelungstechnik ist die gezielte Einflussnahme auf technische Prozesse zur Verbesserung des dynamischen und statischen Verhaltens. Dabei sind wichtige Kriterien der sichere, ökonomische und ökologische Betrieb von Automatisierungssystemen. Fortschritte in diesen Bereichen werden im Wesentlichen durch folgende Weiterentwicklungen getragen:

- Regelungstheorie/Prozessidentifikation
- Sensorik/Aktorik
- Mikroprozessor-Technologie und Kommunikationsnetze

Dadurch ergeben sich Aufgaben im Bereich der Gestaltung von Mensch-Maschine-Schnittstellen, der Entwicklung von Steuerungen und Regelungen und der Überwachung und Fehlerdiagnose. Die Bewältigung derartiger Aufgaben erfordert ein reichhaltiges Instrumentarium an Methoden. Die Entwicklung neuartiger Verfahren wird von der kontinuierlichen Leistungssteigerung von Mikroprozessoren begleitet bzw. vorangetrieben. Auf heutigen digitalen Regelungssystemen sind heute problemlos Algorithmen implementierbar, die noch vor 10 Jahren undenkbar waren. Auch die Art der Software-Erstellung hat sich gewandelt. Funktionen fortgeschrittener Regelalgorithmen können häufig in einer grafischen Hochsprache erstellt und mittels automatischer Codegenerierung in das Regelsystem integriert werden. Die Intelligenz eines Regelsystems ist meist nicht in einem zentralen Leitrechner konzentriert, sondern Vorverarbeitungs- und Diagnosefunktionen sind in modernen Sensoren und Aktoren integriert. Diese verteilte Intelligenz wird durch leistungsfähige Bussysteme unterstützt.

Durch die immer stärkere Verzahnung von Elektronik und Mechanik hat sich die Disziplin *Mechatronik* entwickelt, die das Ziel einer Gesamtsystembetrachtung hat. Dabei verschwinden klassische Grenzen von Teilsystemen. Wenn früher die Regelung einer elektrischen Maschine an der Motorwelle endete, so rückt heute immer mehr die Regelung des gesamten angetriebenen Systems in den Vordergrund. Um diese komplexen Systeme zu beschreiben und gezielt zu beeinflussen, kommen meist rechnergestützte Identifikations- und Entwurfsverfahren zum Einsatz [4]. Die genauere Prozessmodellierung ebnet gleichzeitig den Weg für modellgestützte Regelungsverfahren.

Die meisten industriellen Regelungen beruhen immer noch auf PID-ähnlichen Regelalgorithmen. Durch steigende Anforderungen an Qualität und Wirtschaftlichkeit der Anlagen, können nicht mehr alle Randbedingungen eingehalten werden. Ein seit längerem in der Praxis eingesetztes modellbasiertes Regelungsverfahren stellt die *modellbasierte prädiktive Regelung* dar. Dieses Verfahren konzentrierte sich in der Anfangszeit auf lineare Mehrgrößensysteme der Verfahrenstechnik mit *langsamer* Dynamik [66]. Das Prinzip beruht

auf der Nutzung eines Prozessmodells zur Vorhersage zukünftiger Regelgrößen, um daraus Rückschlüsse auf optimale aktuelle Stelleingriffe zu ziehen. Dieses Prinzip ist sehr universell einsetzbar und nicht auf eine bestimmte Systemklasse begrenzt. Insbesondere ist das Verfahren für lineare und nichtlineare Prozesse anwendbar. Bemerkenswert ist auch die Tatsache, dass dieses universell einsetzbare Verfahren aus der industriellen Praxis den Weg in die Universitäten gefunden hat.

Dieses intuitive Konzept, sich in einer konkreten Entscheidungssituation so zu verhalten, dass in der Zukunft ein bestmögliches Ergebnis zu erwarten ist, hat sicherlich wegen seiner bestechenden Einfachheit eine weite Verbreitung in Forschung und Industrie gefunden. Die ersten prädiktiven Regelverfahren basierten alle auf linearen Prozessmodellen. Besonderer Wert wurde auf die Einbeziehung von Prozessbeschränkungen gelegt, was bis heute einen entscheidenden Vorteil gegenüber den meisten anderen Regelungsverfahren darstellt. Die Unterschiede der Verfahren sind in verschiedenen zugrunde liegenden Prozessmodellen zu suchen. So verwendet das *Dynamic Matrix Control* (DMC) Verfahren ein Gewichtsfolgenmodell zur Vorhersage [66]. Das später entwickelte *Generalized Predictive Control* (GPC) Verfahren setzt dagegen ein zeitdiskretes Übertragungsfunktions-Modell ein [64, 65], das in [44] in den Zustandsraum übertragen wird und ein Kalman-Filter zur Zustandsschätzung benutzt.

In [69] wird die Einbindung nichtlinearer Prozessbeschreibungen aufgezeigt. Der entwickelte *Nonlinear Quadratic Dynamic Matrix Control* (NLQDMC) Algorithmus basiert auf einer nichtlinearen Modellbeschreibung, die zur Prädiktion linearisiert wird. Inzwischen ist die Verwendung nichtlinearer Modellbeschreibungen zur Prädiktion in den Mittelpunkt des Forschungsinteresses gerückt. Durch den Einsatz nichtlinearer Modelle werden sowohl neue Einsatzgebiete für die prädiktive Regelung geschaffen, die bisher nicht zufriedenstellend gelöst werden konnten, als auch bestehende Regelungen verbessert, indem sie näher an den Grenzen des zulässigen Arbeitsbereiches betrieben werden können. Dadurch lassen sich z.B. die Produktqualität oder die Effizienz einer Anlage steigern. Eine Auswahl von Verfahren zur nichtlinearen prädiktiven Regelung sind in [39, 58, 79, 80] zu finden.

Bei allen prädiktiven Regelungsverfahren erweist sich der Nachweis der Stabilität als sehr schwierig, selbst bei linearen Verfahren. Dies resultiert aus der Tatsache, dass zur Prädiktion des Prozessverhaltens der Regelkreis aufgetrennt wird. Alleine anhand des Prozessmodells blickt der prädiktive Regler in die Zukunft. Zur Bestimmung der Stellgröße geht man demnach vom offenen Regelkreis aus. Dadurch scheiden viele übliche Stabilitätskriterien aus. In den meisten veröffentlichten Beweisen zur Stabilität wird daher auf die sehr allgemeine Methode von Alexandr Mikhailovich Lyapunov aus dem Jahre 1892, die auf einer Energiebetrachtung für stabile Systeme beruht, zurückgegriffen [77].

Neben dem Reglerentwurf an sich ist die wichtigste Grundlage der prädiktiven Regelung ein möglichst realitätsnahes Prozessmodell. Eine Regelung kann nur so gut wie das zugrunde liegende Modell sein. Deshalb werden in dieser Arbeit die Kap. 2 und 3 der Modellbildung gewidmet. Bei linearen Prozessmodellen kann auf einen reichen Fundus an Methoden zur Identifikation zurückgegriffen werden [13, 34]. Bei nichtlinearen Modellen existieren keine allgemeingültigen Identifikationsverfahren, so dass in den meisten Fällen von einer theoretischen Prozessanalyse ausgegangen wird. Modellverfeinerungen können dann mittels Identifikationsverfahren erfolgen. Für diese, meist sehr speziellen, Prozessmodelle wird

dann mittels Rechnersimulationen ein geeigneter prädiktiver Regler ausgelegt. Da die Vielfalt nichtlinearer Systeme unendlich groß ist, sind auch die möglichen Prozessmodelle sehr unterschiedlich und immer in gewisser Weise auf die geforderte Anwendung zugeschnitten. Ein allgemeines prädiktives Regelungsverfahren für nichtlineare Systeme kann daher nicht erwartet werden. In [58] wurden nichtlineare FIR-Modelle, in [39] nichtlineare Expertenmodelle zur prädiktiven Regelung eingesetzt.

1.2 Motivation und Ziel der Arbeit

Durch die interdisziplinäre Betrachtungsweise von Elektrotechnik und Mechanik in einem System (Mechatronik) entstehen gerade für Antriebs- und Regelungstechniker neue Problemfelder, welche oftmals mit konventionellen Methoden nicht zufriedenstellend gelöst werden können. Ein Grund sind die durch die Systemerweiterung hinzukommenden, teilweise unbekannten, Parameter, Störgrößen und Nichtlinearitäten. Dies führt aus regelungstechnischer Sicht zu erheblichen Problemen. Es steigt der Wunsch nach Verfahren, die einen zusätzlichen Informationsgewinn bereitstellen. Somit gewinnt der Begriff **Identifikation** stärker an Bedeutung, da eine geeignete Modellbildung zu verbesserten Regelungsverfahren führen kann. Die meisten technischen Systeme weisen nichtlinearen Charakter auf, wodurch für die Erstellung genauer Modelle nichtlineare Identifikationsmethoden unverzichtbar sind. Diese aufwändigeren nichtlinearen Streckenmodelle können dann zur gezielten Verbesserung vorhandener Regelungen oder zur Entwicklung neuartiger Regelungen eingesetzt werden.

In den Arbeiten [46, 56, 57] wurde ein nichtlineares Identifikationsverfahren für eine Klasse nichtlinearer Prozesse entwickelt und praktisch erprobt. Dabei lag der Schwerpunkt auf der Identifikation. Die Verbesserung der Regelung erfolgte stark auf die jeweilige Anwendung zugeschnitten. Diese Arbeit setzt auf den nichtlinearen Identifikationsverfahren auf, verfeinert und verbessert diese und schließt einen allgemeinen, modellbasierten Regelungsentwurf für eine breite Klasse nichtlinearer Systeme an. Der Schwerpunkt dieser Arbeit liegt demnach auf einem systematischen, möglichst allgemeingültigen, modellbasierten Regelungsentwurf, der die verfeinerte nichtlineare Modellierung voll nutzt. Durch die zusätzliche Forderung nach Einbeziehung von Begrenzungen bietet sich die modellbasierte prädiktive Regelung als Grundkonzept an. Dieses wird auf die Anforderung nach Berücksichtigung von Nichtlinearitäten erweitert. Um die Praxistauglichkeit der entwickelten Verfahren unter Beweis zu stellen, erfolgt eine Validierung an ausgewählten mechatronischen Antriebssystemen.

Die Arbeit gliedert sich in vier Hauptpunkte: die Modellierung und Identifikation nichtlinearer Systeme, den Entwurf eines selbst lernenden Reglers zur Kompensation periodischer Nichtlinearitäten, dem theoretischen Entwurf einer nichtlinearen modellbasierten prädiktiven Regelung im Zustandsraum und der praktischen Anwendung auf mechatronische Systeme.

Die Grundvoraussetzung für eine qualitativ hochwertige prädiktive Regelung ist ein genaues Prozessmodell. Deshalb werden in **Kapitel 2** verschiedene lineare Prozessbeschreibungen eingeführt und Identifikationsverfahren vorgestellt. Diese linearen Modelle stellen gleichzei-

tig die Basis der nichtlinearen Prozessmodelle in **Kapitel 3** dar. Dabei wird ein bekannter linearer Systemteil vorausgesetzt, der um isolierte Nichtlinearitäten ergänzt wird. Bei der Identifikation werden diese Nichtlinearitäten durch neuronale Netze (*General Regression Neural Network*) nachgebildet und durch stabile Lerngesetze online trainiert. Das Ergebnis ist die Schätzung nichtlinearer Kennlinien und die Beobachtung aller nicht messbaren Systemzustände.

In **Kapitel 4** wird das neuronale Netz zur direkten Regelung eingesetzt. Es wird der Spezialfall einer periodisch wirkenden Nichtlinearität untersucht, wie er häufig bei rotierenden Maschinen auftritt. Das neuronale Netz übernimmt die Kompensation der Nichtlinearität, während alle anderen Regelungsaufgaben weiterhin von einem konventionellen Regler übernommen werden. Die Leistungsfähigkeit wird durch die Unrundheitskompensation an einer kontinuierlichen Fertigungsanlage demonstriert.

Die **Kapitel 5** und **6** setzen auf der Modellbildung aus Kapitel 3 auf und zeigen den systematischen Entwurf einer modellbasierten prädiktiven Regelung mit nichtlinearen Zustandsraummodellen. Neben dem Regelgesetz ohne Beschränkungen erfolgt auch die Einbeziehung von Stell-, Zustands- und Ausgangsbegrenzungen. Die effiziente Lösung der entstehenden Optimierungsprobleme als quadratisches Programm wird durch eine Linearisierung entlang der Referenztrajektorie sichergestellt. Da Stabilität einer Regelung eine wesentliche Eigenschaft darstellt, erfolgt in **Kapitel 7** eine eingehende Stabilitätsbetrachtung der entwickelten Regelung.

Um die Praxistauglichkeit unter Beweis zu stellen, wird in **Kapitel 8** die Regelung an einem nichtlinearen Zweimassensystem implementiert, das aus zwei elastisch gekoppelten Synchronmaschinen besteht. Neben der Einbeziehung von Begrenzungen wird auch die gleichzeitige Identifikation nach Kapitel 3 und Regelung nach Kapitel 6 durchgeführt. Es entsteht ein adaptiver Regler. In **Kapitel 9** wird das nichtlineare prädiktive Regelverfahren auf Mehrgrößenprozesse erweitert und simulativ auf ein großes Radioteleskop angewendet. Dies stellt eine typische Anwendung für prädiktive Regelungen dar, da Sollwerttrajektorien über einen langen Zeitraum im voraus bekannt sind.

In Bild 1.1 ist die Strukturierung der Arbeit nochmals dargestellt.

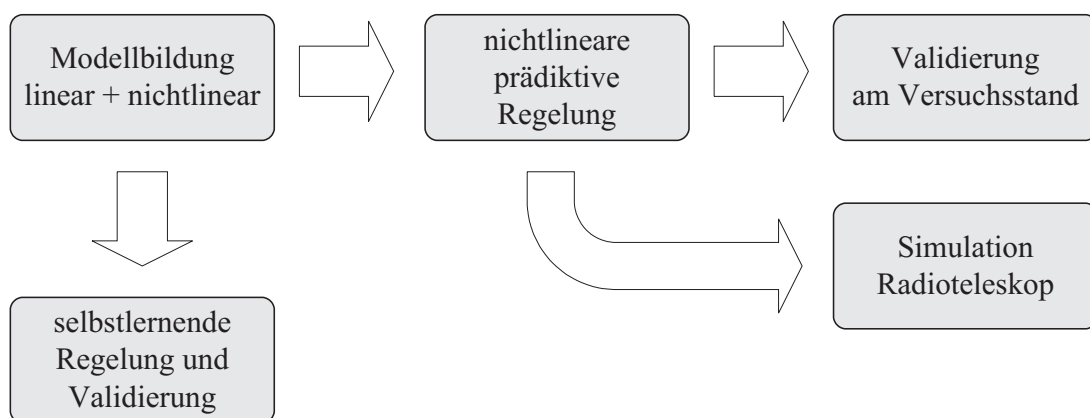


Bild 1.1: Strukturierung der Arbeit

2 Lineare Modellbildung

Bei der Entwicklung moderner Regelungsverfahren gewinnt die Kenntnis eines genauen Prozessmodells zunehmend an Bedeutung. Dies wird durch den Einsatz von Simulationen mechatronischer Systeme verstärkt und durch die Entwicklung immer leistungsfähigerer Rechner und Simulationswerkzeuge gefördert. Die Simulation ermöglicht eine gezielte Analyse der Regelungsaufgabe im Vorfeld und vermeidet aufwändige und teure Experimente. Grundlage jeder Simulation ist ein Modell, das die physikalische Realität vereinfacht, aber genügend genau abbildet.

In diesem Kapitel werden zunächst einige Begriffe erläutert und abgegrenzt, die im Zusammenhang mit der Modellierung und Identifikation eine wichtige Rolle spielen. Anschließend werden Modellbeschreibungen für lineare dynamische Systeme zusammengefasst und an zwei Varianten genauer diskutiert. Da das Ziel dieser Arbeit die prädiktive Regelung *nicht-linearer* Strecken ist, erfolgt in Kap. 3 die nichtlineare Modellierung mittels lernfähigem Zustandsbeobachter. Hierfür ist jedoch die Vorabkenntnis des linearen Streckenanteils nötig, so dass zuerst immer eine lineare Modellbildung erfolgen muss. Die lineare Modellierung erfolgt anhand zeitdiskreter Ein- Ausgangsmodelle, die jedoch in Zustandsraummodelle umgerechnet werden können [15]. Die nichtlineare Modellbildung erfolgt im Zeitkontinuierlichen, weil die dazu benötigte Theorie zu Stabilität (Lyapunov) und Konvergenz einfacher darstellbar ist [17, 33] und der für die prädiktive Regelung notwendige Übergang auf ein zeitdiskretes Zustandsraummodell leicht durchzuführen ist.

2.1 Modellbildung und Identifikation

2.1.1 Modellbildung

Grundlage für eine Systemidentifikation ist immer eine modellhafte Vorstellung der physikalischen Realität. Bei der Modellbildung kann prinzipiell unterschieden werden zwischen der Modellstruktur und den Modellparametern. Während die Modellstruktur das generelle Verhalten und die Komplexität des Modells bestimmt (qualitative Modellbeschreibung), bestimmen die Parameter das spezielle Verhalten eines Modells mit gegebener Struktur (quantitative Modellbeschreibung). Im Allgemeinen können folgende Ansätze zur Modellbildung, abhängig vom Vorwissen über das zu modellierende System, unterschieden werden:

- **White-Box-Modelle:** White-Box-Modelle resultieren aus einer genauen theoretischen Analyse des Systemverhaltens. Diese Analyse erfolgt durch das Aufstellen von physikalischen und geometrischen Gleichungen. Charakteristisch für White-Box-Modelle ist, dass die Modellstruktur genau bekannt ist und die Modellparameter physikalischen Parametern entsprechen. Die Modellparameter können durch Messungen abgeglichen werden. White-Box-Modelle weisen eine hohe Genauigkeit auf, setzen aber voraus, dass das Systemverhalten sehr genau analysiert wurde, was unter Umständen sehr zeitaufwändig sein kann. Man spricht in diesem Fall von einem

parametrischen Modell.

- **Black-Box-Modelle:** Möchte man eine aufwändige theoretische Analyse vermeiden oder hat man nur unzureichende Kenntnisse über das Systemverhalten, bedient man sich der experimentellen Analyse oder Identifikation. Das Resultat einer Identifikation ist ein so genanntes Black-Box-Modell oder ein Grey-Box-Modell. Unterschiede zwischen beiden ergeben sich aus dem Grad des Vorwissens, das in das Modell eingebracht wird. Charakteristisch für Black-Box-Modelle ist, dass keine oder nur sehr wenige Vorkenntnisse über das Systemverhalten bzw. die Systemstruktur bekannt sind und die Modellparameter keine physikalische Bedeutung haben. Man spricht in diesem Fall auch von einem nichtparametrischen Modell, da das Modell nur das Ein-/Ausgangsverhalten abbildet und die physikalischen Parameter z.B. implizit in Form von Gewichtungsfunktionen enthalten sind.
- **Grey-Box-Modelle:** Grey-Box-Modelle sind eine Mischung aus White-Box- und Black-Box-Modellen. Sie beinhalten im Allgemeinen Informationen aus physikalischen Gleichungen und Messdaten sowie qualitative Informationen in Form von Regeln. Grey-Box-Modellen liegt häufig eine Annahme über die Struktur des Prozesses zugrunde. Man spricht in diesem Fall auch von einem parametrischen Modell, da die Parameter einer gewissen Modellvorstellung zugeordnet werden können, was nicht gleichzeitig bedeutet, dass sie den physikalischen Parametern entsprechen.

In Bild 2.1 sind die erläuterten Begriffe noch einmal grafisch gegeneinander abgegrenzt. Die

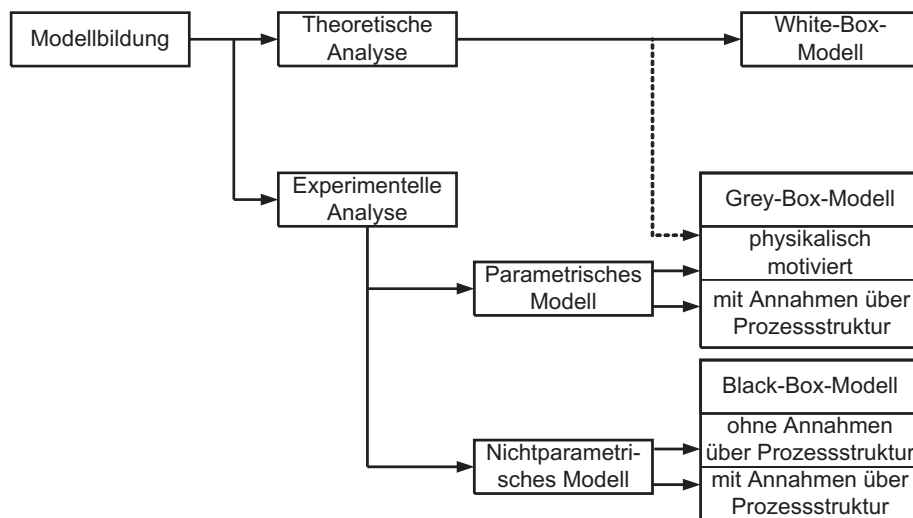


Bild 2.1: Systemmodellierung

Wahl der Systemanalyse (theoretisch oder experimentell) hängt sehr stark vom Vorwissen über das System und vom Verwendungszweck des gewonnen Modells ab. Benötigt man interne Systemzustände zum Aufbau einer Regelung, ist im Allgemeinen die theoretische Analyse vorzuziehen (vgl. Kap. 3.3). Benötigt man lediglich das Ein-/Ausgangsverhalten eines Modells zur Prädiktion, Simulation oder Diagnose, kann man auf die experimentelle Analyse zurückgreifen. In dieser Arbeit wird sowohl auf die experimentelle Analyse mittels Identifikation (Kap. 2.2) als auch auf Prozessmodelle mit Vorwissen eingegangen (Kap. 3.3).

2.1.2 Identifikation

Ziel der experimentellen Analyse bzw. der Identifikation ist es, mit Hilfe gemessener Ein- und Ausgangssignale des Prozesses ein Modell zu bestimmen, welches das statische und dynamische Verhalten möglichst gut nachbildet. Dabei wird angenommen, dass ein eindeutiger Zusammenhang zwischen dem Eingangssignal $u(t)$, der Anregung des Prozesses, und dem Prozessausgangssignal $y(t)$ existiert. Da auf jeden Prozess Störungen, wie z.B. Messrauschen einwirken, kann nur das gestörte Prozessausgangssignal $y_r(t)$ zur Identifikation genutzt werden, welches aus der Summe des ungestörten Prozessausgangs $y(t)$ und einem Störsignal $z(t)$ entsteht. Bild 2.2 zeigt die grundsätzliche Struktur für die Identifikation dynamischer Prozesse. Physikalische und zeitliche Abhängigkeiten innerhalb des

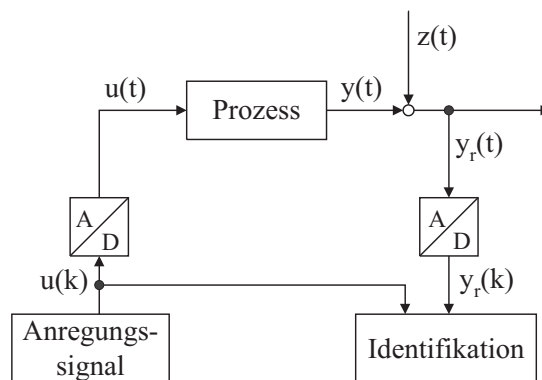


Bild 2.2: Identifikationsstruktur eines dynamischen Prozesses

Prozesses werden bei der Ein- Ausgangsidentifikation nicht analysiert. Wird eine innere Prozessgröße benötigt, so muss das dynamische Verhalten zwischen Prozesseingang und der inneren Größe durch eine zweite Identifikation untersucht werden. Jede Identifikation setzt sich aus zwei grundlegenden Schritten zusammen:

- Strukturauswahl zur Beschreibung des Prozessverhaltens
- Optimierung der freien Parameter

Zunächst muss für das Modell eine Struktur festgelegt werden. Die Strukturauswahl bestimmt, welche funktionalen Zusammenhänge beschrieben werden können. So können durch eine lineare Modellstruktur auch nur lineare Prozesseigenschaften abgebildet werden. Auf die unterschiedlichen Arten der Modellstrukturen und ihre Eignung für die Identifikation wird in Kap. 2.2 genauer eingegangen. Je nach gewählter Struktur unterscheidet man nichtparametrische und parametrische Modelle und Identifikationsverfahren [11, 13]. Nichtparametrische Modelle zeichnen sich dadurch aus, dass wenig Vorwissen eingebracht werden muss (Black-Box-Modell). Allerdings sind die zugehörigen Identifikationsverfahren sehr rechenintensiv, da eine hohe Parameterzahl optimiert werden muss. Parametrische Identifikationsverfahren haben den Vorteil, dass die Modelleigenschaften durch eine geringe Parameteranzahl beschrieben werden. Aufgrund des geringen Speicher- und Rechenbedarfs eignen sich diese Verfahren auch für den Online-Einsatz. Die am häufigsten verwendeten mathematischen Modelle beschreiben den Prozess als lineares, zeitinvariantes Übertragungssystem

(LTI), da dies von vielen regelungstechnischen Verfahren vorausgesetzt wird und zu einfach handhabbaren Identifikationsverfahren führt.

Im zweiten Schritt müssen die Parameter der gewählten Modellstruktur so angepasst werden, dass ein zwischen Prozess und Modell gebildetes Fehlersignal $e(t)$ möglichst klein wird. Dieser Schritt wird deshalb als Parameteroptimierung bezeichnet. Die Optimierung erfolgt bei den meisten Identifikationsverfahren dadurch, dass mit Hilfe gemessener Ein- und Ausgangssignale ein überbestimmtes Gleichungssystem erzeugt wird, welches durch Ausgleichsverfahren näherungsweise gelöst wird.

Die Prozessanregung hat einen entscheidenden Einfluss auf die Modellgüte. Die einfache Vorstellung einer Messung, bei der der Prozess während der gesamten Messzeit in Ruhe verweilt, macht deutlich, dass aus einer solchen Messung keine Informationen über das dynamische Verhalten des Systems gewonnen werden können [45]. Bei der Identifikation muss also stets darauf geachtet werden, dass der gesamte Eingangsraum, wie auch die relevante Dynamik des Prozesses durch eine entsprechende Wahl des Eingangssignals angeregt werden.

In Kap. 2.2.2 wird ein parametrisches Ein- Ausgangsmodell genauer untersucht und ein Verfahren zur Parameteroptimierung angegeben. Dieses parametrische lineare Modell ist die Basis für das in Kap. 3.3 erweiterte nichtlineare Prozessmodell, das die Grundlage des prädiktiven Regelverfahrens in Kap. 6 darstellt. Ein Beispiel eines nichtparametrischen Modells mit höherer Parameterzahl wird in Kap. 2.2.3 untersucht.

2.2 Lineare Prozessmodelle

Für lineare dynamische Systeme existiert eine einheitliche Beschreibungstheorie, die sowohl für den zeitkontinuierlichen als auch den zeitdiskreten Fall entwickelt wurde. Die Systemidentifikation linearer Systeme hat sich in der Regelungstechnik bereits seit langem durchgesetzt und findet zahlreiche Anwendungsmöglichkeiten [13]. Bild 2.3 gibt einen Überblick über die Möglichkeiten zur Beschreibung linearer dynamischer Systeme. In der Literatur werden sowohl Identifikationsverfahren im Zeit- als auch im Frequenzbereich untersucht. In dieser Arbeit werden ausschließlich Verfahren im Zeitbereich betrachtet, da die resultierenden Modelle zur prädiktiven Regelung in Kap. 6 eingesetzt werden.

Aufgrund der höheren Flexibilität und der immer weiter fortschreitenden Mikroprozessortechnik werden zur Regelung zeitkontinuierlicher Prozesse fast ausschließlich Digitalrechner oder Steuergeräte verwendet. Digitale Regelungen unterscheiden sich von zeitkontinuierlich arbeitenden in der zeitlichen Abarbeitung (zeitdiskret) und der Signaldarstellung (wertdiskret). Das Messen der Regelgröße $y(t)$ und die Ausgabe der Stellgröße $u(t)$ erfolgt über Analog-Digital-Umsetzer (AD) bzw. Digital-Analog-Umsetzer (DA) zu diskreten Zeitpunkten. Die Abarbeitung des Regelalgorithmus geschieht meistens zu äquidistanten Zeitpunkten, wobei die Zeitspanne zwischen zwei Abarbeitungen als Abtastzeit T_{ab} bezeichnet wird. Die eingelesenen Signale werden im Rechner als Zahl mit einer maximalen Bitbreite verarbeitet, wodurch sich eine wertdiskrete Darstellung ergibt. Durch die heute üblicherweise verwendeten AD-Wandler mit einer Bitbreite von 12 oder 16 kann die wertdiskrete Zahlendarstellung vernachlässigt werden. Die Werte im Digitalrechner bestehen nun aus der Pro-

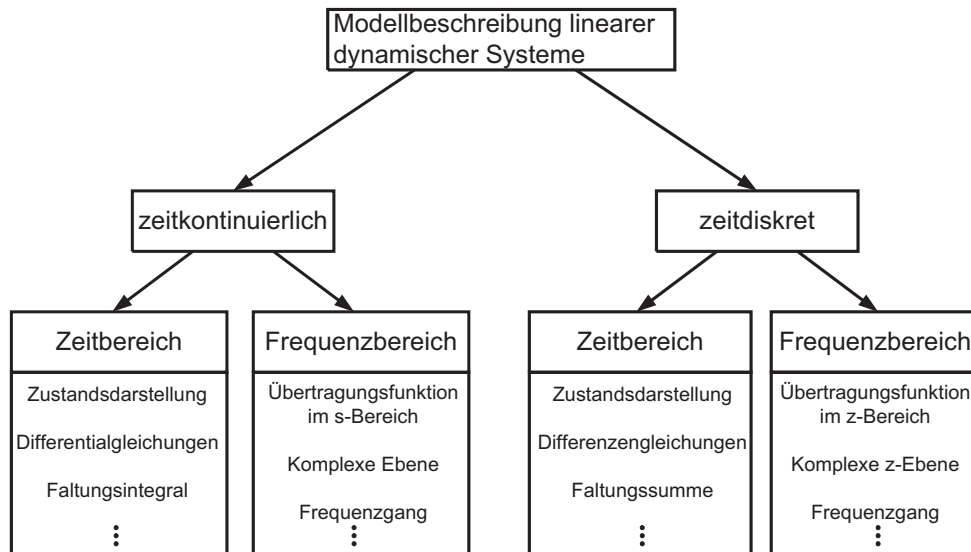


Bild 2.3: Beschreibungsformen linearer dynamischer Systeme

zesseingangssequenz $\{u(k)\}$ und der Prozessausgangssequenz $\{y(k)\}$ mit $u(k) = u(k \cdot T_{ab})$ und $y(k) = y(k \cdot T_{ab})$. Bild 2.4 zeigt das Prinzip einer digitalen Regelung. Wegen der

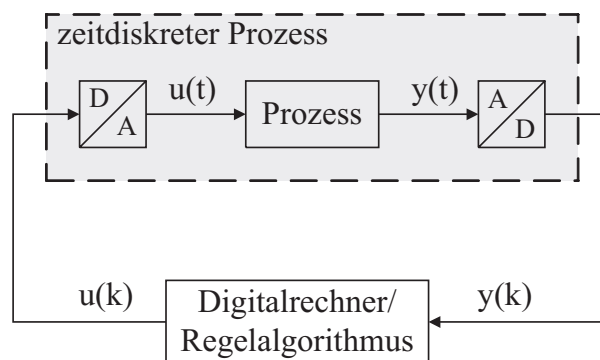


Bild 2.4: Digital geregelter Prozess

zeitdiskreten Abarbeitung in Digitalrechnern kommt auch zeitdiskreten Prozessmodellen eine wichtige Bedeutung zu. Die Aufgabe einer zeitdiskreten Modellbildung ist es, den dynamischen Zusammenhang zwischen Eingangs- und Ausgangssequenz zu beschreiben. Ein zeitdiskretes Prozessmodell kann als stroboskopisches Modell angesehen werden, das den Zusammenhang zwischen Eingang und Ausgang jeweils nur zu den Abtastzeitpunkten wiedergibt.

Da der Übergang von einem bekannten linearen zeitkontinuierlichen Prozessmodell auf ein zeitdiskretes Modell sehr wichtig ist und auch bei der prädiktiven Regelung in Kap. 6 Verwendung findet, wird nun die Diskretisierung eines linearen zeitinvarianten Prozessmodells (LTI) in Zustandsdarstellung durchgeführt. Eine Beschreibung als Übertragungsfunktion kann vorab in eine Zustandsdarstellung umgerechnet werden [15]. Ferner ist es durch die Umrechnung möglich, ein durch zeitkontinuierliche Identifikation (siehe Bild 2.3) erhaltenes

Modell in ein zeitdiskretes Modell ohne erneute Identifikation umzurechnen. Der umgekehrte Weg ist auch möglich. Für das durch theoretische Analyse ermittelte zeitkontinuierliche Modell stellt Gleichung (2.1)

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_c \mathbf{x}(t) + \mathbf{B}_c \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_c \mathbf{x}(t) + \mathbf{D}_c \mathbf{u}(t)\end{aligned}\quad (2.1)$$

mit den Prozesseingängen $\mathbf{u}(t) \in \mathbb{R}^{o \times 1}$, den Prozessausgängen $\mathbf{y}(t) \in \mathbb{R}^{m \times 1}$, der Zustandsmatrix $\mathbf{A}_c \in \mathbb{R}^{N \times N}$, der Einkoppelmatrix $\mathbf{B}_c \in \mathbb{R}^{N \times m}$, der Auskoppelmatrix $\mathbf{C}_c \in \mathbb{R}^{m \times N}$, dem Durchgriff $\mathbf{D}_c \in \mathbb{R}^{m \times o}$ und dem Zustandsvektor $\mathbf{x} \in \mathbb{R}^{N \times 1}$ eine geeignete Beschreibung dar. Die Lösung des Differentialgleichungssystems lautet:

$$\mathbf{x}(t) = \exp(\mathbf{A}_c \cdot (t - t_0)) \cdot \mathbf{x}(t_0) + \int_{t_0}^t \exp(\mathbf{A}_c \cdot (t - \tau)) \cdot \mathbf{B}_c \cdot \mathbf{u}(\tau) d\tau \quad (2.2)$$

Ein zeitdiskretes Modell erhält man durch Lösung des Differentialgleichungssystems (2.1) zwischen zwei Abtastzeitpunkten.

$$\mathbf{x}(k+1) = \exp(\mathbf{A}_c \cdot T_{ab}) \cdot \mathbf{x}(k) + \int_{kT_{ab}}^{(k+1)T_{ab}} \exp(\mathbf{A}_c \cdot ((k+1) \cdot T_{ab} - \tau)) \cdot \mathbf{B}_c \cdot \mathbf{u}(\tau) d\tau \quad (2.3)$$

Zur Berechnung der partikulären Lösung von (2.3) benötigt man den Verlauf der Eingangsgröße $u(t)$ zwischen den Abtastzeitpunkten kT_{ab} und $(k+1)T_{ab}$. Dazu geht man davon aus, dass der DA-Wandler am Eingang des kontinuierlichen Prozesses zwischen zwei Abtastzeitpunkten ein konstantes Signal liefert. Der DA-Wandler besitzt intern ein Halteglied nullter Ordnung, das das aktuelle Eingangssignal solange konstant hält, bis ein neues Eingangssignal zur Verfügung steht. Diese Methode bezeichnet man auch als Diskretisierung mittels *zero-order-hold*. Berücksichtigt man diese Annahme, so erhält man folgende diskrete Zustandsraumbeschreibung

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_d \mathbf{x}(k) + \mathbf{D}_d \mathbf{u}(k)\end{aligned}\quad (2.4)$$

mit den Matrizen

$$\mathbf{A}_d = \exp(\mathbf{A}_c \cdot T_{ab}) \quad \mathbf{B}_d = \int_0^{T_{ab}} \exp(\mathbf{A}_c \cdot (T_{ab} - \tau)) \cdot \mathbf{B}_c d\tau \quad \mathbf{C}_d = \mathbf{C}_c \quad \mathbf{D}_d = \mathbf{D}_c \quad (2.5)$$

Wenn der Prozess nur einen Eingang und einen Ausgang besitzt (SISO), so lassen sich die Zustandsvariablen $\mathbf{x}$ eliminieren und man gelangt zu einer Ein- Ausgangsbeschreibung in Form einer linearen Differenzengleichung N -ter Ordnung mit konstanten Koeffizienten (Kap. 2.2.2 und [15]).

$$y(k) = b_0 u(k) + b_1 u(k-1) + \dots + b_N u(k-N) - a_1 y(k-1) - \dots - a_N y(k-N) \quad (2.6)$$

Die Differenzengleichung (2.6) lässt sich durch Definition der Zustandsgrößen $x_1 = y(k-N)$, $x_2 = y(k-N+1)$, $\dots$, $x_N = y(k-1)$ wieder in Zustandsdarstellung umformen [10].

Dazu bietet sich die Regelungsnormalform nach Gleichung (2.7) an.

$$\mathbf{x}(k+1) = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & \ddots & & \vdots \\ 0 & \cdots & & & 1 & 0 \\ 0 & \cdots & & & 0 & 1 \\ -a_N & -a_{N-1} & \cdots & & & -a_1 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k) \quad (2.7)$$

$$y(k) = [b_N - b_0 a_N \quad b_{N-1} - b_0 a_{N-1} \quad \cdots \quad b_1 - b_0 a_1] \mathbf{x}(k) + b_0 u(k)$$

Generell ist bei Zustandsdarstellungen zu beachten, dass diese bzgl. eines bestimmten Ein-Ausgangsverhaltens nicht eindeutig bestimmt sind. Das bedeutet, zu einem gegebenen Ein- Ausgangsverhalten existieren immer unendlich viele Zustandsdarstellungen, die jeweils durch nichtsinguläre Ähnlichkeitstransformationen ineinander übergehen [15]. Das heißt insbesondere, dass die Zustandsmatrizen in den Gleichungen (2.4) und (2.7) im Allgemeinen nicht identisch sind.

Die Umrechnung eines mit der Abtastzeit T_{ab} diskretisierten Modells in das zugehörige zeitkontinuierliche Modell erfolgt ebenfalls mit Gleichung (2.5). Bei gegebenen Matrizen $\mathbf{A}_d$ und $\mathbf{B}_d$ berechnen sich die zugehörigen Matrizen $\mathbf{A}_c$ und $\mathbf{B}_c$ mit dem natürlichen Logarithmus $\ln$ nach Gleichung (2.8).

$$\mathbf{A}_c = \frac{\ln(\mathbf{A}_d)}{T_{ab}} \quad \mathbf{B}_c = \left[\int_0^{T_{ab}} \exp(\mathbf{A}_c \cdot (T_{ab} - \tau)) d\tau \right]^{-1} \cdot \mathbf{B}_d \quad (2.8)$$

In den folgenden Abschnitten werden Verfahren zur Identifikation der Parameter von Ein-Ausgangsmodellen dargestellt. Die Identifikation einer kompletten Zustandsdarstellung ist wegen ihrer Mehrdeutigkeit im Allgemeinen nicht möglich, jedoch ist die Umrechnung einer Ein- Ausgangsdarstellung auf eine Zustandsdarstellung mittels Gleichung (2.7) möglich, so dass sie als Basis für die lernfähigen Zustandsraummodelle in Kap. 3.3 dienen kann. Für die Identifikation ist die Wahl der Modellstruktur von entscheidender Bedeutung. Deshalb werden nun einige Modellstrukturen vorgestellt, die in der Literatur zur Identifikation linearer dynamischer Systeme häufig verwendet werden [11, 12, 13].

2.2.1 Lineare Ein- Ausgangsmodelle

Der Ausgangspunkt für zeitdiskrete lineare Ein- Ausgangsmodelle ist eine Modellvorstellung entsprechend Bild 2.5 mit dem Ein-Schritt Schiebeoperator z . Das Modell ist charakterisiert durch folgende Gleichung:

$$A(z^{-1}) \cdot y(k) = B(z^{-1}) \cdot u(k) + C(z^{-1}) \cdot z(k) \quad (2.9)$$

mit den Polynomen in z^{-1}

$$A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_{na} z^{-na} \quad (2.10)$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_{nb} z^{-nb} \quad (2.11)$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + \cdots + c_{nc} z^{-nc} \quad (2.12)$$

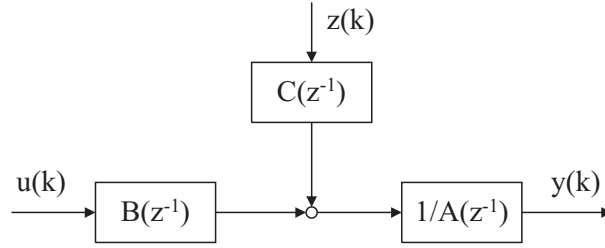


Bild 2.5: Allgemeine Modellstruktur

Dabei bezeichnet $u(k)$ die Eingangsgröße und $y(k)$ die Ausgangsgröße des Systems. Zusätzlich wird das System durch ein Rauschsignal $z(k)$ angeregt. Für das Rauschsignal wird angenommen, dass es sich um **weißes Rauschen** handelt, d.h. alle aufeinander folgenden Signalwerte sind statistisch unabhängig. Der Mittelwert des Rauschsignals ist null und die Leistungsdichte ist über den gesamten Frequenzbereich konstant. Formuliert man Gleichung (2.9) zu einer Differenzengleichung um, erhält man Gleichung (2.13).

$$\begin{aligned} y(k) = & -a_1 y(k-1) - \dots - a_{na} y(k-na) + b_0 u(k) + b_1 u(k-1) + \\ & + b_{nb} u(k-nb) + z(k) + c_1 z(k-1) + \dots + c_{nc} z(k-nc) \end{aligned} \quad (2.13)$$

Schreibt man Gleichung (2.13) übersichtlich in Vektorform ergibt sich:

$$y(k) = \mathbf{m}^T(k) \cdot \boldsymbol{\Theta} + z(k) \quad (2.14)$$

mit dem Vektor

$$\mathbf{m}^T(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-na) & u(k) & \dots & u(k-nb) & z(k-1) & \dots & z(k-nc) \end{bmatrix} \quad (2.15)$$

und dem Parametervektor

$$\boldsymbol{\Theta}^T = [a_1 \dots a_{na} \ b_0 \dots b_{nb} \ c_1 \dots c_{nc}] \quad (2.16)$$

Ausgehend von dieser allgemeinen Form werden verschiedene Modelle definiert, die sich aus unterschiedlichen Annahmen über das Eingangs- und Störsignal sowie die Polynome A , B und C ergeben. Folgende Spezialfälle können unterschieden werden:

- **AR-Modell (autoregressive model):** Ein AR-Modell erhält man, wenn $nb = nc = 0$ und $u(k) = 0 \ \forall k$. In diesem Fall gilt:

$$A(z^{-1}) \cdot y(k) = z(k) \quad (2.17)$$

Der Parametervektor ergibt sich zu:

$$\boldsymbol{\Theta}^T = [a_1 \dots a_{na}] \quad (2.18)$$

- **MA-Modell (moving average model):** Für das MA-Modell wird angenommen, dass $na = nb = 0$ und $u(k) = 0 \ \forall k$ ist. Es gilt dann:

$$y(k) = C(z^{-1}) \cdot z(k) \quad (2.19)$$

Als Parametervektor erhält man:

$$\boldsymbol{\Theta}^T = [c_1 \dots c_{nc}] \quad (2.20)$$

- **ARMA-Modell (autoregressive moving average model):** Von einem ARMA-Modell spricht man, wenn $nb = 0$ und $u(k) = 0 \forall k$.

$$A(z^{-1}) \cdot y(k) = C(z^{-1}) \cdot z(k) \quad (2.21)$$

Für den Parametervektor erhält man:

$$\Theta^T = [a_1 \dots a_{na} \ c_1 \dots c_{nc}] \quad (2.22)$$

- **FIR-Modell (finite impulse response):** Ein FIR-Modell erhält man, wenn $na = nc = 0$ ist.

$$y(k) = B(z^{-1}) \cdot u(k) + z(k) \quad (2.23)$$

Der Parametervektor ergibt sich zu:

$$\Theta^T = [b_0 \dots b_{nb}] \quad (2.24)$$

- **ARX-Modell (autoregressive model with exogenous input):** Für das ARX-Modell wird angenommen, dass $nc = 0$ ist.

$$A(z^{-1}) \cdot y(k) = B(z^{-1}) \cdot u(k) + z(k) \quad (2.25)$$

Als Parametervektor erhält man:

$$\Theta^T = [a_1 \dots a_{na} \ b_0 \dots b_{nb}] \quad (2.26)$$

- **ARMAX-Modell (autoregressive moving average model with exogenous input):** Für das ARMAX-Modell gibt es keine speziellen Annahmen, d.h. es ist durch die allgemeinen Gleichungen (2.9) bis (2.16) gegeben.

Alle vorgestellten parametrischen Modelle resultieren letztendlich in einem Ausgleichsproblem zur Bestimmung des Parametervektors Θ . Auf die verschiedenen Ausgleichsverfahren wird in Kap. 2.2.4 noch genauer eingegangen. In Software-Tools, wie z.B. der *System Identification Toolbox* aus Matlab/Simulink, existieren Standardlösungen für lineare Identifikationsprobleme.

Wenn keine genauen Informationen über den Verlauf des Störsignals $z(k)$ bekannt sind, so ist das ARX-Modell eine geeignete Wahl. Dieses entspricht einer linearen Differenzengleichung bzw. einer Übertragungsfunktion in der komplexen z -Ebene [2, 22]. Sollen zur Identifikation keine vergangenen Ausgangssignale verwendet werden (nicht rekursiv), so stellt das FIR-Modell eine mögliche Wahl dar. Das FIR-Modell entspricht einer linearen Faltungssumme. An diesen beiden Beispielen werden grundlegende Unterschiede zwischen rekursiven und nicht-rekursiven Modellen deutlich.

2.2.2 Lineare Differenzengleichung

Die lineare Differenzengleichung beschreibt ein dynamisches System als IIR-Filter (*infinite impulse response*). Charakteristisch ist, dass das aktuelle Ausgangssignal $y(k)$ sowohl

aus vergangenen Eingangs- als auch Ausgangswerten gebildet wird. Damit können, im Gegensatz zur Faltungssumme in Kap. 2.2.3, auch instabile Modelle gebildet werden. Die allgemeine lineare Differenzengleichung lautet:

$$\begin{aligned} a_0 y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = \\ b_0 u(k) + b_1 u(k-1) + \dots + b_m u(k-m) \end{aligned} \quad (2.27)$$

Sie gibt die Beziehung zwischen den zeitdiskreten Größen $u(k) = u(k \cdot T_{ab})$ und $y(k) = y(k \cdot T_{ab})$ an. Bei der Verwendung der Differenzengleichung für die Identifikation, wird meist die Ordnung $m = n = N$ gleich der Systemordnung N gesetzt und die Gleichung auf $a_0 = 1$ normiert. Durch Auflösen der Gleichung (2.27) nach $y(k)$ entsteht Gleichung (2.28), die es erlaubt das aktuelle Ausgangssignal $y(k)$ aus vergangenen Ein- und Ausgangsgrößen zu berechnen.

$$y(k) = -a_1 y(k-1) - \dots - a_N y(k-N) + b_0 u(k) + b_1 u(k-1) + \dots + b_N u(k-N) \quad (2.28)$$

Deshalb wird dieser Ansatz auch oft *rekursive Differenzengleichung* genannt. Diese Darstellung entspricht genau dem ARX-Modell. Ein weiteres Merkmal ist die Linearität in den Parametern a_i und b_i , weshalb Gleichung (2.28) übersichtlich in Vektorform geschrieben werden kann.

$$y(k) = \mathbf{m}^T(k) \cdot \Theta \quad (2.29)$$

mit dem Daten- oder Messvektor

$$\mathbf{m}^T(k) = [-y(k-1) \dots -y(k-N) \quad u(k) \quad u(k-1) \dots u(k-N)] \quad (2.30)$$

und dem Parametervektor

$$\Theta^T = [a_1 \dots a_N \quad b_0 \quad b_1 \dots b_N] \quad (2.31)$$

Durch Einführung des Ein-Schritt Rückwärtsschiebeoperators z^{-1} lässt sich die Differenzengleichung (2.28) als zeitdiskrete Übertragungsfunktion $G(z^{-1})$ im komplexen z -Bereich darstellen [2].

$$\frac{y(k)}{u(k)} = G(z^{-1}) = \frac{b_0 + b_1 z^{-1} \dots + b_N z^{-N}}{1 + a_1 z^{-1} \dots + a_N z^{-N}} \quad (2.32)$$

Bei der linearen Differenzengleichung als Systemmodell ist allein durch Vorgabe der Systemordnung die Modellstruktur festgelegt. Sie ist als Systemmodell sehr weit verbreitet, da sich mit einer geringen Parameteranzahl jedes lineare zeitinvariante System (LTI-System) beschreiben lässt. Die Bestimmung der Parameter a_1 bis b_N aus Messdaten erfolgt in Kap. 2.2.4.

2.2.3 Faltungssumme

Eine weitere Beschreibung linearer dynamischer Systeme ist die Impulsantwort $h(t)$. Sie ist die Antwort eines dynamischen Systems auf eine Anregung mit dem δ -Impuls. Durch

sie wird das dynamische Verhalten des Systems vollständig charakterisiert [23]. Ist die Impulsantwort bekannt, so kann über das Faltungsintegral

$$y(t) = \int_{-\infty}^{+\infty} h(\tau) u(t - \tau) d\tau \quad (2.33)$$

der Modellausgang zu jedem Zeitpunkt eindeutig bestimmt werden. Da reale Prozesse das Kausalitätsprinzip erfüllen, gilt für die Impulsantwort:

$$h(\tau) = 0 \quad \text{für} \quad \tau < 0 \quad (2.34)$$

Dies bedeutet, dass eine Veränderung am Ausgang erst nach einer Anregung am Eingang erfolgt. Aus diesem Grund kann im Faltungsintegral die untere Integrationsgrenze zu null gesetzt werden [45]. Für $y(t)$ gilt dann:

$$y(t) = \int_0^{\infty} h(\tau) u(t - \tau) d\tau \quad (2.35)$$

Bei zeitdiskreten Systemen sind Eingangssignal, Ausgangssignal und Impulsantwort nur zu den Abtastzeitpunkten kT_{ab} verfügbar. Die zeitdiskrete Impulsantwort $h(k)$ ist die Antwort des Prozesses an den Abtastzeitpunkten auf eine zeitdiskrete Impulsanregung $\delta(k)$. In der zeitdiskreten Darstellung geht das Faltungsintegral in die Faltungssumme über und das Ausgangssignal ergibt sich zu:

$$y(k) = \sum_{i=0}^{\infty} h(i) u(k - i) \quad (2.36)$$

Für stabile Systeme streben die Koeffizienten der Impulsantwort $h(k)$ gegen null.

$$\lim_{k \rightarrow \infty} h(k) = 0 \quad (2.37)$$

Die Faltungssumme kann deshalb unter Vernachlässigung eines Restfehlers bei einer oberen Grenze a abgeschnitten werden. Sie wird im Folgenden als Antwortlänge a bezeichnet und hängt von der Systemdynamik und der Abtastzeit ab. Bei einer angemessenen Wahl der oberen Grenze sind die entstehenden Ungenauigkeiten für das Modell gering. Das Ausgangssignal berechnet sich unter Berücksichtigung der Antwortlänge a zu:

$$y(k) = \sum_{i=0}^a h(i) u(k - i) \quad (2.38)$$

bzw.

$$y(k) = h(0) \cdot u(k) + h(1) \cdot u(k - 1) + \dots + h(a) \cdot u(k - a) \quad (2.39)$$

Diese Darstellung entspricht dem FIR-Modell. Die Gewichtsfolgenwerte $h(k)$ stellen die zu bestimmenden Parameter dar. In der Darstellungsform der Faltungssumme (2.38) ist zu erkennen, dass sich der aktuelle Ausgang *nur* aus dem aktuellen und zurückliegenden gewichteten *Eingangssignalen* ergibt. Da bei der Faltungssumme alle Parameter linear in

die Berechnung des aktuellen Ausgangssignals $y(k)$ eingehen, kann auch dieser Ansatz übersichtlich in Vektorform angegeben werden.

$$y(k) = \mathbf{m}^T(k) \cdot \Theta \quad (2.40)$$

Diese Gleichung ist formal äquivalent zu Gleichung (2.29). Unterschiede zeigen sich jedoch im Messvektor $\mathbf{m}^T(k)$,

$$\mathbf{m}^T(k) = \begin{bmatrix} u(k) & u(k-1) & \dots & u(k-a) \end{bmatrix} \quad (2.41)$$

der nur Eingangssignale enthält, und im Parametervektor Θ ,

$$\Theta = \begin{bmatrix} h(0) & h(1) & \dots & h(a) \end{bmatrix} \quad (2.42)$$

in dem die einzelnen Gewichtsfolgenwerte auftreten. Man erkennt, dass die Anzahl der Elemente im Parametervektor allein von der oberen Grenze der Faltungssumme abhängt. Die sich ergebende Parameterzahl $p = a + 1$ ist im Allgemeinen deutlich größer, als die Anzahl der Parameter der Differenzengleichung.

Der wesentliche Unterschied zwischen Faltungssumme und Differenzengleichung besteht darin, dass bei der Differenzengleichung das Ausgangssignal aus vergangenen *Ein- und Ausgangswerten* berechnet wird, während bei der Faltungssumme *nur Eingangswerte* zur Berechnung des aktuellen Ausgangssignals herangezogen werden. Beide Darstellungsarten haben Vor- und Nachteile. Beim nichtrekursiven Ansatz der Faltungssumme bleibt das geschätzte Modell immer stabil, solange die Eingangssignale begrenzt bleiben. Dies ist beim rekursiven Ansatz der Differenzengleichung nicht in jedem Fall gewährleistet. Dies bedeutet aber auch im Umkehrschluss, dass die Darstellung instabiler Systeme mittels Faltungssumme nicht möglich ist, da in diesem Fall die Antwortlänge a unendlich wäre. Bei stark gestörten Prozessen können die Parameter der Differenzengleichung nur fehlerbehaftet bestimmt werden, was in Kap. 2.2.5 näher erläutert wird. Im Gegensatz dazu können die Gewichtsfolgenwerte der Faltungssumme strukturbedingt annähernd fehlerfrei bestimmt werden [13]. Ein Nachteil des Ansatzes der Faltungssumme ist die wesentlich höhere Parameterzahl, als beim rekursiven Ansatz der Differenzengleichung. Bei linearen Systemen stellt dies zwar noch kein Problem dar, hingegen bei einem erweiterten Ansatz für nichtlineare Systeme (Volterra-Reihe) werden für die praktische Anwendung Vereinfachungen nötig [45].

2.2.4 Methode der kleinsten Quadrate

Hat man sich bei der Identifikation für ein mathematisches Modell entschieden, so müssen in einem zweiten Schritt die jeweiligen Parameter so an den Prozess angepasst werden, dass das Modell das Verhalten des Prozesses möglichst gut wiedergibt. Dazu wird ein zwischen Prozess und Modell gebildetes Fehlersignal e minimiert. Dieser Vorgang wird als Parameteroptimierung bezeichnet. Ausgehend von der Messgleichung (2.29) bzw. (2.40) wird ein Gleichungssystem durch Auswertung der Messgleichung zu m Zeitpunkten aufgebaut.

$$y(k) = \mathbf{m}^T(k) \cdot \hat{\Theta} + e(k) \quad (2.43)$$

$$y(k+1) = \mathbf{m}^T(k+1) \cdot \hat{\boldsymbol{\Theta}} + e(k+1) \quad (2.44)$$

$$\vdots$$

$$y(k+m-1) = \mathbf{m}^T(k+m-1) \cdot \hat{\boldsymbol{\Theta}} + e(k+m-1) \quad (2.45)$$

Dabei wird für den Parametervektor die Bezeichnung $\hat{\boldsymbol{\Theta}}$ eingeführt, um ihn vom optimalen Parametervektor $\boldsymbol{\Theta}$ zu unterscheiden. $\boldsymbol{\Theta}$ beschreibt den Prozess exakt, wenn keine Störeinflüsse vorhanden wären. Der Vektor $\hat{\boldsymbol{\Theta}}$ wird aus gemessenen und deshalb auch verauschten Signalen berechnet. Ist die Messreihe zur Schätzung der im Vektor $\hat{\boldsymbol{\Theta}}$ zusammengefassten Parameter groß genug, können die durch Messrauschen verursachten Fehler im Sinne eines Ausgleichs minimiert werden, so dass im besten Fall der geschätzte Parametervektor $\hat{\boldsymbol{\Theta}}$ mit dem optimalen Parametervektor $\boldsymbol{\Theta}$ übereinstimmt. Überführt man das Gleichungssystem in Matrixschreibweise, erhält man:

$$\mathbf{y} = \mathbf{M} \cdot \hat{\boldsymbol{\Theta}} + \mathbf{e} \quad \text{mit} \quad \mathbf{M} = [\mathbf{m}^T(k) \dots \mathbf{m}^T(k+m-1)]^T \quad \hat{\mathbf{y}} = \mathbf{M} \cdot \hat{\boldsymbol{\Theta}} \quad (2.46)$$

Die Anzahl der aufgestellten Messgleichungen muss größer oder zumindest gleich der Anzahl zu schätzender Parameter sein, da sonst das Gleichungssystem unterbestimmt wäre. Im ungestörten Fall ließe sich aus Gleichung (2.46) unter der Voraussetzung, dass die Eingangsgröße den Prozess ausreichend anregt, der Parametervektor eindeutig bestimmen. Da $\mathbf{y}$ und $\mathbf{M}$ gestörte Messwerte enthalten, stellt Gleichung (2.46) ein nicht lösbares Gleichungssystem dar. Es kann also nur versucht werden, das Gleichungssystem möglichst gut zu lösen. Alle Störungen werden wegen der Gültigkeit des Superpositionsprinzips in Gleichung (2.46) in einem Störsignal am Ausgang zusammengefasst. Zur Ausmittlung der Messfehler wird in der Praxis die Anzahl der Messgleichungen erheblich größer als die Parameteranzahl gewählt. Grundsätzlich lässt sich feststellen, dass eine größere Anzahl von Messgleichungen zu besseren Schätzergebnissen führt. Löst man Gleichung (2.46) nach dem Fehlervektor $\mathbf{e}$ auf, erhält man:

$$\mathbf{e} = \mathbf{y} - \mathbf{M} \cdot \hat{\boldsymbol{\Theta}} = \mathbf{y} - \hat{\mathbf{y}} \quad (2.47)$$

Der Fehler $\mathbf{e}$ beschreibt die Abweichung zwischen den gemessenen Ausgangswerten $\mathbf{y}$ und den Ausgangswerten $\hat{\mathbf{y}}$ des Modells. Aufgabe der Parameteroptimierung ist es, die Parameterwerte so zu bestimmen, dass die im Fehlervektor $\mathbf{e}$ zusammengefassten Fehlersignale $e(k), \dots, e(k+m-1)$ in ihrer Summe ein Minimum ergeben. Als Maß für die Größe der Abweichung wird die quadratische euklidische Vektornorm verwendet, was der im folgenden vorgestellten Methode auch ihren Namen „Least-Squares-Verfahren“ (kurz: LS-Verfahren) oder „Methode der kleinsten Quadrate“ (kurz: MKQ) gegeben hat.

$$\|\mathbf{e}\|_2^2 = \sum_{i=0}^{m-1} e^2(k+i) \quad (2.48)$$

Das Gütekriterium J zur Minimierung des Fehlers wird wie folgt definiert:

$$J(\hat{\boldsymbol{\Theta}}) = \|\mathbf{e}\|_2^2 = \mathbf{e}^T \cdot \mathbf{e} = \left(\mathbf{y} - \mathbf{M} \cdot \hat{\boldsymbol{\Theta}} \right)^T \cdot \left(\mathbf{y} - \mathbf{M} \cdot \hat{\boldsymbol{\Theta}} \right) \quad (2.49)$$

Die Wahl des quadratischen Fehlers zur Minimierung des Gütefunktionalen eignet sich aus folgenden Gründen:

- Das Gütefunktional J besitzt als einziges Extremum ein eindeutiges globales Minimum [24].
- Es ist keine Fallunterscheidung für die Vorzeichen des Fehlersignals $e(k)$ zu treffen.
- Starke Abweichungen zwischen gemessenem und geschätztem Ausgangssignal gehen mit mehr Gewicht in die Berechnung ein als kleine Abweichungen.

Das Minimum des Gütefunktionals (2.49) wird durch Differentiation nach dem Parametervektor und anschließendes Nullsetzen berechnet.

$$\frac{\partial J}{\partial \hat{\Theta}} = -2\mathbf{M}^T \cdot (\mathbf{y} - \mathbf{M} \cdot \hat{\Theta}) \stackrel{!}{=} 0 \quad (2.50)$$

Durch das Auflösen dieser Gleichung nach $\hat{\Theta}$ erhält man als Lösung die optimalen Parameterwerte.

$$\hat{\Theta} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{y} \quad (2.51)$$

Das in den Gleichungen (2.43) bis (2.51) vorgeführte Verfahren stellt den nichtrekursiven Least-Squares-Algorithmus dar [11]. In Gleichung (2.51) kommt der Matrix $\mathbf{M}^T \mathbf{M}$ – auch Kovarianzmatrix genannt – eine besondere Bedeutung zu. Solange diese Matrix invertierbar ist, existiert eine Lösung des Gleichungssystems. Für die Praxis folgt aus dieser Tatsache, dass der Prozess genügend angeregt werden muss, um vollen Rang der Kovarianzmatrix zu garantieren.

Da die Messwerte in Gleichung (2.46) sukzessive anfallen, liegt es nahe, zunächst eine Schätzung des Parametervektors durchzuführen und diese bei jeder weiteren Messung zu verbessern. Dies führt auf die rekursive Methode der kleinsten Quadrate [11]. Bei der rekursiven Methode muss die Matrixinversion der Kovarianzmatrix ($\mathbf{M}^T \mathbf{M}$) nicht durchgeführt werden. Die Matrixinversion zerfällt in eine Division durch eine skalare Größe, was den Rechenaufwand deutlich reduziert.

Im Laufe der letzten Jahrzehnte wurde eine Fülle verschiedener linearer Ausgleichsverfahren entwickelt, wobei die meisten auf der Idee des Least-Squares-Algorithmus beruhen. Sie haben zum Ziel, das Konvergenzverhalten und die Güte der Modelle bei stark verrauschten Signalen zu verbessern. Dazu werden Filtertechniken für die Eingangs- und Ausgangssignale angewendet [13, 11, 34]. Es muss dabei immer bedacht werden, dass bei einer Filterung der Eingangssignale auch die Anregung des Modells verringert wird.

Neben der Modellordnung und der Abtastzeit ist die Wahl des Eingangssignals sehr wichtig. Es muss den Prozess ausreichend anregen, so dass alle dynamischen Effekte während der Identifikation auch auftreten. Neben der Leistung ist auch das Leistungsdichtespektrum des Anregungssignals von Bedeutung. So macht es z.B. wenig Sinn, ein System mit Tiefpasscharakter mit einem Anregungssignal identifizieren zu wollen, das die meiste Leistungsdichte bei hohen Frequenzen enthält. Diese hohen Frequenzen werden dann durch das System selbst nahezu ausgelöscht.

2.2.5 Modelle zur Fehlerminimierung

In diesem Abschnitt werden die Unterschiede zwischen Faltungssumme und Differenzengleichung noch einmal aufgegriffen. Ausgangspunkt ist die Messgleichung, die zu vielen Zeitpunkten aufgestellt wird, so dass sich ein überbestimmtes Gleichungssystem ergibt. Mittels des Least-Squares-Verfahrens können die Parameterwerte bestimmt werden. Für beide Ansätze ergibt sich Gleichung (2.52) für das Ausgangssignal $\hat{y}$ des Modells.

$$\hat{y}(k) = \mathbf{m}^T(k) \cdot \hat{\Theta} \quad (2.52)$$

Allerdings sind die Inhalte der Vektoren $\mathbf{m}$ und $\hat{\Theta}$ sehr wohl verschieden. Beim Ansatz der Differenzengleichung enthält der Messvektor sowohl Eingangs- als auch Ausgangswerte.

$$\mathbf{m}^T(k) = \begin{bmatrix} -y(k-1), \dots, -y(k-N), u(k), u(k-1), \dots, u(k-N) \end{bmatrix} \quad (2.53)$$

Bei der Faltungssumme als Identifikationsansatz kann der Modellausgang allein aus dem aktuellen und zurückliegenden Eingangssignalen berechnet werden. Der Prozessausgang wird ausschließlich zur Bildung des Modellfehlers e verwendet, der als Maß für die Parameterbestimmung dient. Dies spiegelt sich im Messvektor der Faltungssumme in Gleichung (2.54) wieder.

$$\mathbf{m}^T(k) = \begin{bmatrix} u(k), u(k-1), \dots, u(k-a) \end{bmatrix} \quad (2.54)$$

Bild 2.6 veranschaulicht die strukturellen Unterschiede der beiden Ansätze. Die linke Iden-

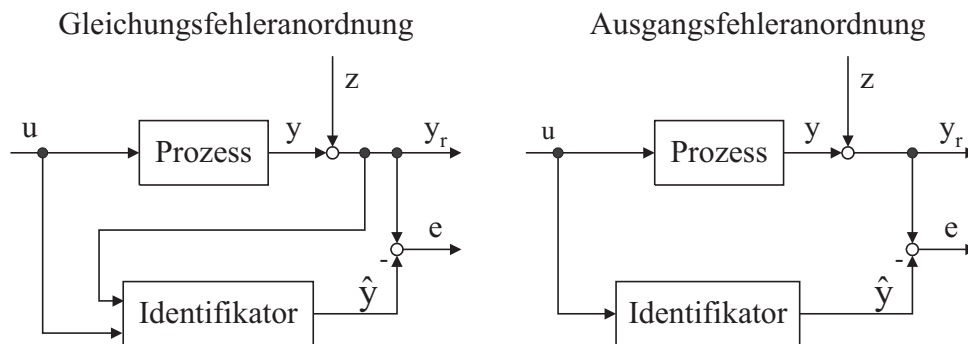


Bild 2.6: Gleichungsfehleranordnung – Ausgangsfehleranordnung

tifikationsstruktur wird als Gleichungsfehleranordnung bezeichnet und entsteht bei Verwendung der Differenzengleichung als Modellansatz. Man erkennt, dass zur Berechnung des geschätzten Ausgangssignals vom Identifikator neben den Eingangssignalen auch *gemessene* Ausgangssignale herangezogen werden. Bei dieser Struktur wird der so genannte Gleichungsfehler (oder auch 1-Schritt-Prädiktionsfehler) gebildet und minimiert. Die Bezeichnung Gleichungsfehler oder 1-Schritt-Prädiktionsfehler wird verwendet, weil der Identifikator das aktuelle Ausgangssignal nicht eigenständig, sondern nur mit Hilfe der zuletzt *gemessenen* Ausgangssignale schätzt. Somit wird vom Identifikator nur eine 1-Schritt-Prädiktion zum neuen Ausgangssignal ausgeführt.

Die rechte Identifikationsstruktur wird als Ausgangsfehleranordnung bezeichnet und ergibt sich bei Verwendung der Faltungssumme als Systemmodell. Zur Berechnung des Modellausgangs werden *nur Eingangssignale* verwendet. Die Modellausgangsgröße wird durch

den Identifikator eigenständig, allein mit dem Wissen der Anregung erzeugt. Das gemessene Ausgangssignal y_r wird nur zur Fehlerbildung mit dem geschätzten Ausgangssignal $\hat{y}$ verwendet. Deshalb wird bei dieser Identifikationsanordnung der tatsächliche Ausgangsfehler des Modells gebildet und im Verlauf der Identifikation minimiert. Es entsteht ein *echt paralleles Modell*.

Im Gegensatz dazu entsteht bei der Gleichungsfehleranordnung ein *seriell-paralleles Modell*. Bei genauer Betrachtung erkennt man, dass Prozess und Modell bezüglich des Eingangs $u(k)$ parallel, aber bezüglich des Prozessausgangs $y(k)$ seriell verbunden sind. Bei nicht oder wenig verrauschten Messsignalen stellt der Gleichungsfehler eine gute Näherung für den Ausgangsfehler dar [11], so dass das geschätzte Modell nach erfolgreicher Identifikation auch parallel betrieben werden kann. Dies gilt jedoch nicht mehr, wenn die Störungen auf den Prozess zunehmen. Denn anders als bei der Ausgangsfehleranordnung, bei der sich Messstörungen nur auf das gebildete Fehlersignal e auswirken, wird bei der Gleichungsfehleranordnung zusätzlich der Messvektor verfälscht. Dem Identifikator werden die gemessenen *verrauschten* Ausgangssignale y_r zugeführt. Es ergibt sich folgender Messvektor:

$$\mathbf{m}^T(k) = \left[-y_r(k-1), \dots, -y_r(k-N), u(k), u(k-1), \dots, u(k-N) \right] \quad (2.55)$$

Bei der Gleichungsfehleranordnung ist die Parameterschätzung unter Einfluss von Messrauschen im Allgemeinen mit einem systematischen Fehler behaftet (biasbehaftet) [18]. Aus diesem Grund wurden aufwändige Ausgleichsverfahren entwickelt, wie z.B. die Methode der verallgemeinerten kleinsten Quadrate, die Methode der Hilfsvariablen und die Methode der totalen kleinsten Quadrate, die alle zum Ziel haben, die Genauigkeit der Parameterschätzung bei gestörtem Messvektor $\mathbf{m}$ zu erhöhen. Die Praxis hat jedoch gezeigt, dass auch mit diesen aufwändigen Methoden keine vollständig biasfreien Schätzungen erzielt werden, da reales Messrauschen in der Regel nicht die restriktiven Bedingungen erfüllt, die zur Herleitung dieser Verfahren idealisiert angenommen werden.

Prinzipiell wäre es möglich, auch bei der rekursiven Differenzengleichung den Ausgangsfehler zu minimieren. Dafür müsste der Modellausgang $\hat{y}$ statt des gemessenen Ausgangssignals y_r auf den Modelleingang zurückgeführt werden, was in Bild 2.7 dargestellt ist. Man spricht in diesem Fall auch von einem OE-Modell (*output error model*). In der Ausgangsfehleranor-

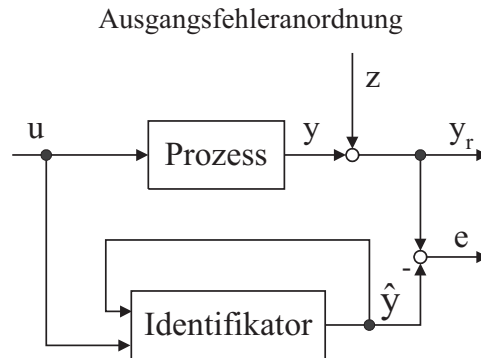


Bild 2.7: Ausgangsfehleranordnung bei rekursivem Modellansatz

nung für rekursive Modellansätze liegt Parallelität zwischen Prozess und Modell vor. Das

bedeutet, die Parameter können auch bei Störungen auf den Prozess biasfrei geschätzt werden. Allerdings können während der Identifikation, bedingt durch die modelleigene Ausgangsrückführung, Stabilitätsprobleme auftreten. Außerdem führt die Identifikationsstruktur nach Bild 2.7 dazu, dass der Modellausgang *nicht mehr linear* von den Parametern abhängig ist. Das bedeutet, es können keine linearen Ausgleichsverfahren angewandt werden. Vielmehr muss auf nichtlineare Optimierungsverfahren zurückgegriffen werden. Dies wird klar, wenn man sich die Modellgleichungen zu Bild 2.7 für ein System erster Ordnung für zwei aufeinander folgende Zeitschritte betrachtet.

$$\begin{aligned}\hat{y}(k) &= -a_1\hat{y}(k-1) + b_1u(k-1) \\ \hat{y}(k+1) &= -a_1(-a_1\hat{y}(k-1) + b_1u(k-1)) + b_1u(k) \\ &= a_1^2\hat{y}(k-1) - a_1b_1u(k-1) + b_1u(k)\end{aligned}\tag{2.56}$$

Die zurückliegenden Modellausgangssignale müssen wieder in die Differenzengleichung eingesetzt werden, was dazu führt, dass sogar bei diesem einfachen Modell das Ausgangssignal nicht mehr linear in den Parametern ist. Den Einsatz von aufwändigeren nichtlinearen Optimierungsalgorithmen möchte man in der Regel vermeiden, weswegen der Ausgangsfehleranordnung nach Bild 2.7 zur Identifikation linearer Systeme praktisch kaum Bedeutung zukommt.

3 Nichtlineare Modellbildung

Zur verbesserten Regelung technischer Prozesse ist es notwendig neben der linearen Dynamik auch nichtlineare Effekte zu berücksichtigen. Dieser Gedanke wird bei der prädiktiven Regelung in Kap. 6 weiter verfolgt. Die meisten Prozesse weisen nichtlinearen Charakter auf, was in der Vergangenheit häufig dadurch behandelt wurde, dass eine Linearisierung um einen Arbeitspunkt durchgeführt wurde. Wenn diese Linearisierung nicht die erforderliche Genauigkeit liefert oder häufige Arbeitspunktwechsel auftreten, werden zur Modellierung nichtlineare Methoden notwendig.

Bei nichtlinearen dynamischen Systemen nimmt die Komplexität und Vielfalt, verglichen mit linearen dynamischen Systemen, deutlich zu. Bis heute gibt es keine einheitliche, in sich geschlossene, mathematische Theorie zur Beschreibung und Identifikation solcher Systeme. Im Rahmen von Forschungsarbeiten wurden deshalb immer Identifikations- und Regelverfahren für bestimmte *Klassen von nichtlinearen dynamischen Systemen* entwickelt. Der andauernde Forschungsaufwand kann dadurch gerechtfertigt werden, dass eine Herausforderung der Regelungstechnik darin besteht, die Genauigkeit und die dynamischen Eigenschaften von Systemen zu verbessern bzw. durch den Einsatz nichtlinearer Verfahren deren Regelung überhaupt erst möglich zu machen. Dadurch eröffnen sich neue Anwendungsgebiete für die Regelungstechnik. Als Beispiele sind der verstärkte Einsatz elektrischer Aktoren in der Automobiltechnik (X-by-Wire, vollvariable Ventilverstellung), Erhöhung der Effizienz von Produktionsanlagen (Erhöhung der Produktionsgeschwindigkeit bei gleichbleibender Qualität) oder der zunehmende Einsatz von Simulationstools bei der Produktentwicklung zu nennen.

Im Folgenden wird ein kurzer Überblick über mögliche Modellbeschreibungen für nichtlineare Systeme gegeben. Als Basis für die prädiktive Regelung in Kap. 6 eignen sich Zustandsraummodelle besonders gut. Lernfähige Zustandsraummodelle mit neuronalen Netzen werden deshalb eingehend behandelt und durch Messergebnisse untermauert. Neuronale Beobachter wurden bereits in [40, 46, 56, 57] zur Systemidentifikation eingesetzt. In dieser Arbeit wird ein durchgängiges Regelungsverfahren auf Basis der gewonnenen nichtlinearen Zustandsmodelle hergeleitet, wobei der Fokus auf der breiten Anwendbarkeit liegt und keine anwendungsspezifischen Lösungen im Vordergrund stehen. Der Bereich **Modellierung** nimmt in dieser Arbeit viel Raum ein, da er die **Basis für eine prädiktive Regelung** erst schafft. Es werden Modelle benötigt, die einerseits in der Lage sind den Prozess genau zu beschreiben, deren Parameter aber andererseits durch Identifikation leicht bestimmbar sind. Diese Voraussetzung wird vom lernfähigen Zustandsraummodell in Kap. 3.3 erfüllt. In Kap. 8.4 wird gezeigt, dass sich ungenaue Modellparameter nachteilig auf das Regelergebnis auswirken können. Aus diesem Grund kommt die Online-Fähigkeit des lernfähigen Zustandsraummodells positiv zum Tragen. Bei einer Variation der berücksichtigten Nichtlinearität kann diese online im Modell adaptiert werden und es steht stets ein genaues Prozessmodell zur Verfügung (Kap. 8.6).

3.1 Klassifikation nichtlinearer dynamischer Systeme

Die Vielfalt an nichtlinearen dynamischen Systemen lässt eine vollständige Klassifikation aller vorstellbaren nichtlinearen Systeme nicht zu. Im Folgenden wird daher ein Überblick über die häufigsten Darstellungsformen gegeben. Diese stellen in der Regel die Grundlage für nichtlineare Identifikationsverfahren dar.

3.1.1 Blockorientierte nichtlineare Modelle

In Bild 3.1 sind einige blockorientierte nichtlineare Systeme dargestellt. Sie alle entstehen durch die Kombination aus statischen nichtlinearen Funktionen $\mathcal{NL}_i$ mit linearen dynamischen Systemen $F_i(s)$. Die dargestellten nichtlinearen Modellstrukturen, besonders das

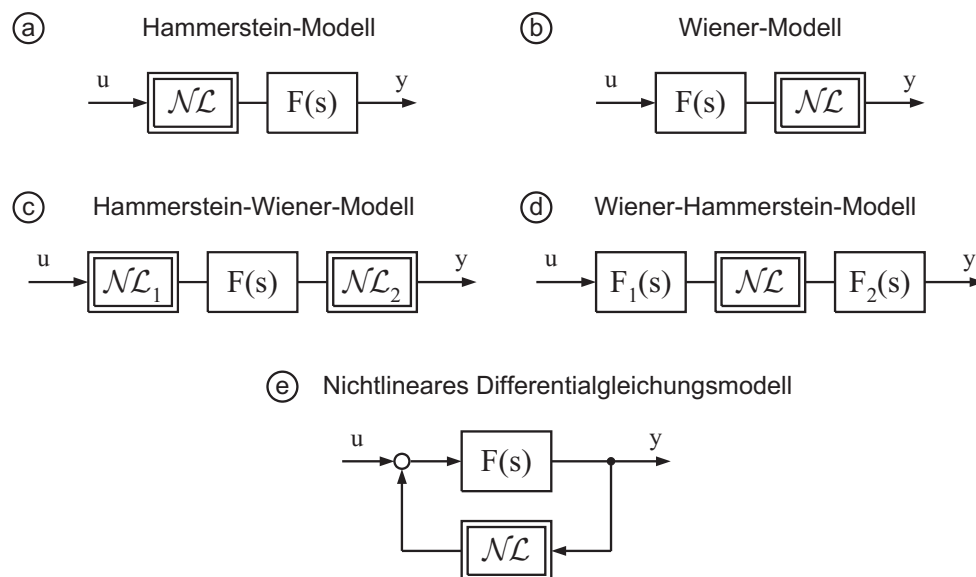


Bild 3.1: Blockorientierte nichtlineare Modelle

Hammerstein- und das Wiener-Modell, stellen in der Literatur häufig verwendete nichtlineare Testprozesse dar [45, 58]. Das Hammerstein-Modell (Bild 3.1 a) setzt sich aus einer statischen nichtlinearen Funktion $\mathcal{NL}$, gefolgt von einem linearen dynamischen System $F(s)$, zusammen. Im Gegensatz dazu, wird beim Wiener-Modell (Bild 3.1 b) die statische Nichtlinearität $\mathcal{NL}$ durch das vorgeschaltete lineare dynamische System angeregt. Das Wiener-Modell stellt im Allgemeinen den komplexeren nichtlinearen Testprozess dar, weil das Eingangssignal der nichtlinearen Funktion $\mathcal{NL}$ nicht als Messgröße zur Verfügung steht. Weiterhin sind auch Mischmodelle beider Ansätze denkbar, wie z.B. das in Bild 3.1 c gezeigte Hammerstein-Wiener-Modell oder das in Bild 3.1 d dargestellte Wiener-Hammerstein-Modell. Eine weitere wichtige Struktur ist das nichtlineare Differentialgleichungsmodell entsprechend Bild 3.1 e. Charakteristisch ist die statische Nichtlinearität $\mathcal{NL}$ in der Rückkopplung. Im Allgemeinen wird angenommen, dass es sich bei den statischen Nichtlinearitäten $\mathcal{NL}$ um stetige differenzierbare Funktionen handelt.

3.1.2 Allgemeine nichtlineare Modelle

Nicht alle nichtlinearen Systeme lassen sich durch blockorientierte Modelle darstellen. Fast alle regelungstechnischen Systeme mit konzentrierten Parametern lassen sich hingegen als nichtlineares Differentialgleichungssystem erster Ordnung darstellen. Bild 3.2 zeigt die Beschreibung eines nichtlinearen dynamischen Systems mit mehreren Eingängen und einem Ausgang (MISO) in seiner allgemeinsten Form. Das System lässt sich durch die folgenden

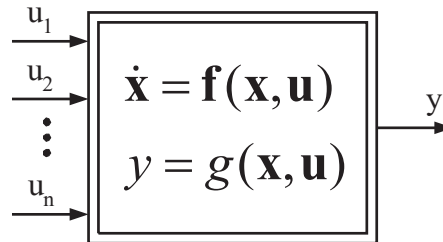


Bild 3.2: Allgemeines nichtlineares Modell

Gleichungen beschreiben:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ y &= g(\mathbf{x}, \mathbf{u})\end{aligned}\tag{3.1}$$

Ein Identifikationsverfahren könnte nun mittels mehrdimensionalem Funktionsapproximator (z.B. neuronales Netz, Splines, Polynome) versuchen die nichtlinearen Funktionen $\mathbf{f}$ und g nachzubilden. Dazu müsste nach Gleichung (3.1) der gesamte Zustandsvektor $\mathbf{x}$ bekannt und messbar sein. Eine Identifikation basierend nur auf Ein- Ausgangsdaten ist mit dem Modell in Bild 3.2 nicht möglich, da Zustandsdarstellungen nie eindeutig sind und bei gleichem Ein- Ausgangsverhalten durch invertierbare Transformationen beliebig ineinander umwandelbar sind [33, 15]. Die allgemeine Gültigkeit des Modells in Gleichung (3.1) erkauft man sich durch hohe Anforderungen an Messbarkeit interner Zustandsgrößen. Dies ist der Grund, warum man versucht die allgemeine Systemklasse zu spezialisieren um dadurch zu leichter realisierbaren Identifikationsstrukturen zu gelangen. Besonders für elektrische Antriebssysteme und mechatronische Systeme ist eine Darstellung als dynamisches System mit isolierter Nichtlinearität besonders geeignet. Man macht sich dabei zu Nutze, dass Vorwissen über Struktur und lineare Dynamik eingebracht werden kann und nur statische Nichtlinearitäten approximiert bzw. identifiziert werden müssen. Dies entspricht einem ingenieurmäßigen Ansatz, der nicht von einer Black-Box ausgeht, sondern eine gewisse Grundvorstellung des Systems voraussetzt. Diese Modellstruktur wird in den folgenden Kapiteln genau untersucht und bildet auch die Modellbasis für die prädiktive Regelung in Kap. 6. Es handelt sich um lernfähige Zustandsraummodelle.

3.1.3 Zustandsdarstellung mit isolierter Nichtlinearität

Ist der nichtlineare Einfluss innerhalb einer Strecke genau bekannt, ist unter gewissen Voraussetzungen eine Zustandsdarstellung mit isolierter Nichtlinearität der folgenden Form

möglich:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A} \cdot \mathbf{x} + \mathbf{b} \cdot u + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}, u) \\ y &= \mathbf{c}^T \mathbf{x} + d \cdot u\end{aligned}\tag{3.2}$$

Gleichung (3.2) beschreibt ein dynamisches System mit einem Eingang und einem Ausgang (SISO). Eine Erweiterung auf Systeme mit mehreren Ein- und Ausgängen ist ohne Einschränkung möglich, soll aber hier zum leichteren Verständnis nicht durchgeführt werden. Die Systemgleichung unterteilt das gesamte nichtlineare System in einen linearen Anteil mit den Systemmatrizen $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ und d und einen nichtlinearen Anteil mit der statischen nichtlinearen Funktion $\mathcal{NL}$ und dem zugehörigen Einkoppelvektor $\mathbf{k}$. Diese Darstellung bietet sich an, wenn das wesentliche Systemverhalten durch ein lineares Differentialgleichungssystem beschreibbar ist, aber zusätzlich nichtlineare Effekte auftreten. Die nichtlineare Funktion $\mathcal{NL}$ kann allgemein den gesamten Zustandsvektor $\mathbf{x}$ und das Eingangssignal u als Argumente erhalten, es ist aber auch nur eine Untermenge dieser Signale als Argumente möglich. Eine grafische Darstellung des Zustandsraummodells mit isolierter Nichtlinearität ist in Bild 3.3 gegeben. Vergleicht man diese Darstellung mit den blockorientierten Modellen aus

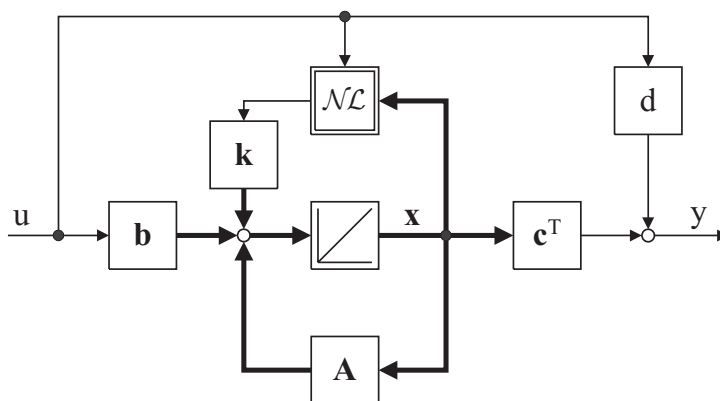


Bild 3.3: Zustandsraummodell mit isolierter Nichtlinearität

Bild 3.1, so fällt auf, dass eine eindeutige Zuordnung der Nichtlinearität zu Vorwärtspfad oder Rückwärtspfad nicht möglich ist. Vielmehr handelt es sich um eine Obermenge, die einige der blockorientierten Modelle aus Bild 3.1 enthält. Das Hammerstein-Modell erhält man, wenn die Nichtlinearität nur vom Eingang u abhängt und für den Einkoppelvektor $\mathbf{b} = \mathbf{0}$ gilt. Das nichtlineare Differentialgleichungsmodell ergibt sich für $\mathbf{k} = \mathbf{b}$ und einer Abhängigkeit der Nichtlinearität vom Ausgangssignal, d.h. $\mathcal{NL} = \mathcal{NL}(y) = \mathcal{NL}(\mathbf{c}^T \mathbf{x} + du)$. Für die Darstellung eines Wiener-Hammerstein-Modells ist die Aufteilung des Zustandsvektors in zwei Untermengen nötig. Die Nichtlinearität ist nur von den Zustandssignalen der Teilübertragungsfunktion $F_1(s)$ und dem Eingangssignal u abhängig. Der Einkoppelvektor $\mathbf{k}$ ist so aufgebaut, dass er nur auf Ableitungen der Zustandsgrößen aus $F_2(s)$ wirkt. Ferner darf keine lineare Kopplung der Zustandsgrößen aus $F_1(s)$ und $F_2(s)$ bestehen, d.h. die Matrix $\mathbf{A}$ besitzt an den entsprechenden Stellen eine Null. Das Wiener-Modell und das Hammerstein-Wiener-Modell sind mit der Zustandsdarstellung mit isolierter Nichtlinearität dagegen nicht darstellbar. Sie erfordern eine nichtlineare Funktion in der Ausgangsgleichung, was in Gleichung (3.2) nicht gegeben ist. Es besteht damit die allgemeine

Einschränkung, dass ein linearer Zusammenhang zwischen Zustandsgrößen und Ausgangssignal bestehen muss. Alle Prozesse, bei denen sich die relevante Regelgröße als nichtlineare Funktion der Zustandsgrößen ergibt und Prozesse mit nichtlinearen Messgliedern sind somit ausgeschlossen.

Zur Identifikation eines Prozesses nach Gleichung (3.2) werden folgende Voraussetzungen getroffen:

- Der lineare Prozessanteil ($\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d) ist bekannt und konstant.
- Die Eingriffspunkte der Nichtlinearität in den Prozess sind bekannt (Einkoppelvektor $\mathbf{k}$).
- Die Argumente der nichtlinearen Funktion sind bekannt, d.h. man weiß von welchen Zustandsvariablen die Nichtlinearität $\mathcal{NL}$ abhängt.
- Die Nichtlinearität $\mathcal{NL}$ ist unbekannt und kann auch langsam zeitvariant sein. $\mathcal{NL}$ ist eine skalare stetige Funktion, die über den gesamten Definitionsbereich differenzierbar ist.

Der lineare Prozessanteil muss vorab entweder durch theoretische Analyse oder eine lineare Identifikation ermittelt werden (siehe Kap. 2). Bei einer linearen Identifikation muss dazu der Prozess ohne Wirkung der Nichtlinearität betrieben werden. Dies kann dadurch geschehen, dass nicht alle Teilkomponenten bei der Identifikation angekoppelt sind (z.B. bei elektrischen Antrieben). Wenn dies nicht möglich ist, können einzelne Parameter auch gleichzeitig mit der nichtlinearen Funktion $\mathcal{NL}$ identifiziert werden [57]. Die aus der Identifikation erhaltene Differenzengleichung kann dann wieder in die benötigte Zustandsform umgewandelt werden [15], wofür auch zahlreiche numerische Tools in der *Control System Toolbox* aus Matlab zur Verfügung stehen [4, 47].

Die Eingriffspunkte der Nichtlinearität sind im Vektor $\mathbf{k}$ zusammengefasst. Jede Komponente ungleich null kennzeichnet einen Eingriffspunkt. Man benötigt also Kenntnis darüber, auf welche Zustände die Nichtlinearität wirkt. Diese Voraussetzung ist nicht sehr einschränkend, wie das Beispiel eines elektrischen Antriebs zeigt. Wenn etwa als Nichtlinearität Reibungseffekte betrachtet werden, so ist die Erkenntnis wichtig, dass das Reibmoment eines Antriebs wieder auf diesen selbst wirkt. Damit ist der Angriffspunkt bestimmt. Als Parameter ist im Vektor $\mathbf{k}$ das Trägheitsmoment enthalten, also ein Parameter, der schon vorher im linearen Teil bestimmt wurde.

Die Argumente der Nichtlinearität sind durch eine vorausgehende Prozessanalyse ebenfalls leicht bestimmbar. Dazu wird das Beispiel des elektrischen Antriebs nochmals aufgegriffen. Wenn Unwuchteffekte auftreten, lässt sich leicht feststellen, dass diese eine Funktion der Winkelstellung des Rotors sind. Hier wird wieder klar, dass es sich nicht um ein Black-Box Modell handelt. Man braucht vor der Identifikation eine gewisse Prozesskenntnis, die ein ingenieurmäßiges Vorgehen erfordert. Im Gegenzug liefert die Identifikation interpretierbare und für den anschließenden Regelungsentwurf relativ leicht handhabbare Modelle. Es handelt sich um physikalisch motivierte Modelle, die die Vorstellung des Ingenieurs wiedergeben und kein rein mathematisches Gebilde darstellen.

Die Nichtlinearität selbst wird als unbekannt angenommen. Außerdem darf sie langsam (im Vergleich zu Prozesszeitkonstanten) zeitvariant sein. Diese Annahme entspringt der Erfahrung, dass nichtlineare Effekte (Reibung, Unwuchten, Federkennlinien, Dämpfungscharakteristika) häufig von Betriebsbedingungen und Betriebsdauer abhängen. So kann sich Reibung mit der Lagerbelastung ändern oder Federkennlinien können durch Alterungseffekte variieren. In der Praxis sind jedoch grobe Vorkenntnisse über die nichtlinearen Funktionen vorhanden. Ziel muss es demnach sein, ein Identifikationsverfahren zu entwickeln, das während des Betriebs (online) die Nichtlinearität schätzt und das Vorwissen sinnvoll aufnehmen kann. Ferner muss der Identifikationsvorgang garantiert stabil ablaufen, da das Identifikationsergebnis für Regelungszwecke weiterverwendet wird.

Unabhängig vom genauen Aufbau der kompletten Identifikation muss ein nichtlinearer Funktionsapproximator zur Verfügung stehen. Neuronale Netze stellen universelle Funktionsapproximatoren dar, die jedoch je nach Netztyp sehr unterschiedliche Eigenschaften besitzen. In Kap. 3.2 werden neuronale Netze mit lokalen Basisfunktionen zur Funktionsapproximation genauer betrachtet. Es wird gezeigt, dass sie die geforderten Eigenschaften sehr gut erfüllen und in die Systemidentifikation in Kap. 3.3 integrierbar sind. Ausführliche Untersuchungen zu diesem Netzwerktyp sind in [54] enthalten.

3.2 Approximation nichtlinearer Funktionen

Für anspruchsvolle Regelungs- und Diagnoseaufgaben werden zunehmend Methoden zur Approximation von Funktionen benötigt, die nicht analytisch beschreibbar oder vorab nicht bekannt sind. Als Grundlage wird zunächst die Klasse der darstellbaren nichtlinearen Funktionen definiert.

Definition 3.1: Nichtlinearität

Eine kontinuierliche, begrenzte und zeitinvariante Funktion $\mathcal{NL} : \mathbb{R}^N \rightarrow \mathbb{R}$, die einen N -dimensionalen Eingangsvektor $\mathbf{x}$ auf einen skalaren Ausgangswert y abbildet, sei eine Nichtlinearität $\mathcal{NL}$ mit $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$.

$$y = \mathcal{NL}(\mathbf{x})$$

Für den Fall mehrdimensionaler Nichtlinearitäten wird entsprechend der Dimension der Ausgangsgröße y die entsprechende Anzahl skalarer Nichtlinearitäten kaskadiert. Daher soll es im Folgenden genügen, lediglich den skalaren Fall zu betrachten. Bei der Nachbildung nichtlinearer Funktionen durch neuronale Netze, werden die Begriffe *Lernen*, *Adaption* und *Identifikation* im Weiteren synonym verwendet und beschreiben unterschiedliche Aspekte der Anwendung. Dabei kann der Schwerpunkt auf die Analogie zu ihren biologischen Vorbildern gelegt werden oder auf ihre technische Funktion als Algorithmen zur Nachbildung funktionaler Zusammenhänge.

3.2.1 Methoden der Funktionsapproximation

Zur Approximation einer Nichtlinearität stehen verschiedene Methoden zur Verfügung, die sich jeweils in ihren Einsatzmöglichkeiten und Randbedingungen unterscheiden. Im Folgenden werden nach [26] einige Möglichkeiten aufgeführt und anschließend näher erläutert (siehe auch Beispiele in Bild 3.4):

- Algebraische Darstellung mit Funktionsreihen
- Tabellarische Darstellung mit Stützstellen
 - Interpolation
 - Approximation
- Konnektionistische Darstellung

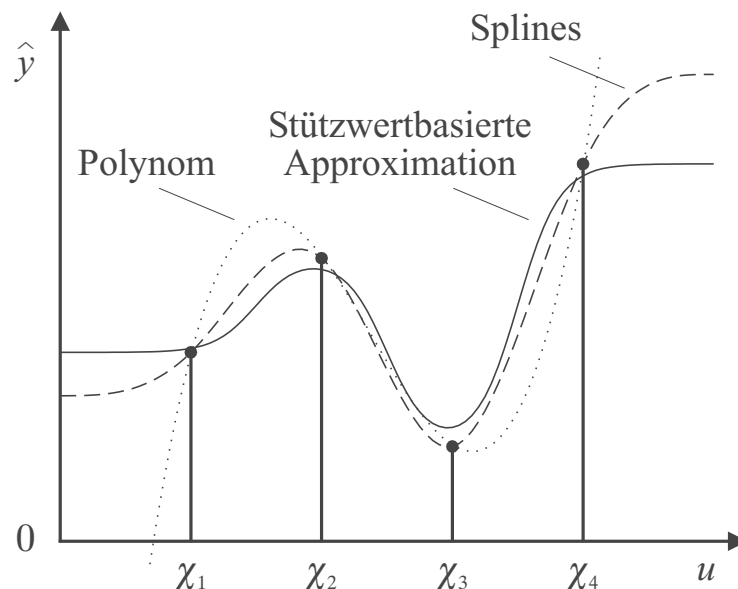


Bild 3.4: Beispiele zur Funktionsapproximation

Bekannte Beispiele für eine **algebraische Darstellung** sind Polynome (wie z.B. die Taylor-Reihe) und Reihenentwicklungen (wie z.B. die Fourier-Reihe). In der Regel hängt die Ausgangsgröße linear von einer endlichen Anzahl an Koeffizienten ab. Nachteilig für den Einsatz in adaptiven Verfahren erweist sich dabei meist, dass jeder Koeffizient auf weite Bereiche des Eingangsraums wirkt, also keine lokale Zuordnung zu bestimmten Eingangswerten möglich ist.

Eine **tabellarische Darstellung** kommt häufig zum Einsatz, wenn eine Funktion bereits rasterförmig vermessen vorliegt. Zwischen diesen meist in einer Tabelle (*Lookup Table*) abgelegten Messwerten wird dann geeignet interpoliert bzw. approximiert. Somit wirken alle Parameter lokal und nachvollziehbar. Ein Sonderfall der Interpolation sind dabei Splines. Diese stellen eine Zwischenform dar, in der durch die Messwerte eine globale tabellarische Repräsentation vorliegt, die aber lokal durch den algebraischen Zusammenhang der

Splines ausgewertet wird. Da eine durch Interpolation nachgebildete Funktion durch alle Messwerte verläuft, wirken sich Messfehler empfindlich aus.

Um diesen Einfluss zu verringern, kann statt der Interpolation auch eine so genannte stützwertbasierte Approximation erfolgen. Im Gegensatz zur Interpolation stehen bei einer Approximation nicht genügend freie Parameter zur Verfügung, um alle Randbedingungen (z.B. in Form vorgegebener Messwerte) zu erfüllen. Stattdessen wird die Abweichung der approximierten Funktion von diesen Vorgaben, der so genannte Approximationsfehler, minimiert. Der Vorteil liegt dabei in der effizienten Berechnung und in der ausgleichenden Wirkung der Approximation, die insbesondere Störeinflüsse reduziert. Beispiele für stützwertbasierte Approximation sind die im weiteren Verlauf behandelten neuronalen RBF-Netze. Die dort verwendeten Stützwerte besitzen zwar einen quantitativen Zusammenhang mit dem Wert der Funktion für den zugehörigen Eingangswert, liegen aber nicht notwendigerweise auf dem approximierten Funktionsverlauf. Durch die glättende Wirkung der Approximation können Störungen und Messfehler wirkungsvoll herausgefiltert werden. Gleichzeitig erlaubt die lokale Zuordnung der Stützwerte zu Bereichen des Eingangsraums eine lokale und schnelle Adaption. In der Regel ist damit auch die Eindeutigkeit der adaptierten Parameter und somit eine Parameterkonvergenz verbunden, die wesentlich für eine Interpretierbarkeit der adaptierten Funktion ist.

Mehrschichtige neuronale Netze, wie z.B. das Multi Layer Perceptron (MLP) Netz, gehören zur Gruppe mit **konnektionistischer Darstellung** und können auch Funktionen mit einer Eingangsgröße hoher Dimension nachbilden. Allerdings ist die Auslegung der Neuronenzahl in den Zwischenschichten (*Hidden Layers*) problematisch; ebenso lässt sich der Nachweis einer optimalen Konvergenz nur empirisch erbringen. Eine Deutung der Parameter ist in aller Regel nicht möglich.

3.2.2 Kriterien zur Beurteilung künstlicher neuronaler Netze

Zur Beurteilung künstlicher neuronaler Netze müssen zunächst einige Kriterien definiert werden, die für die theoretischen Analysen und den praktischen Einsatz dieser Netze von Bedeutung sind. Im einzelnen sind folgende Punkte von Bedeutung [26]:

- **Repräsentationsfähigkeit:** Repräsentationsfähigkeit bedeutet, dass der gewählte Netztyp mit seiner Topologie in der Lage ist, die Abbildungsaufgabe zu erfüllen.
- **Lernfähigkeit:** Der Vorgang der Parameteradaption eines Netzes wird als Lernen bezeichnet. Lernfähigkeit bedeutet, dass die gewünschte Abbildung mittels eines geeigneten Lernverfahrens in das künstliche neuronale Netz eintrainiert werden kann. In der Praxis spielen dabei Kriterien, wie Lerngeschwindigkeit, Stabilität der Lernergebnisse und Online-Fähigkeit des Lernverfahrens, eine wichtige Rolle.
- **Verallgemeinerungsfähigkeit:** Die Verallgemeinerungsfähigkeit beschreibt die Fähigkeit des trainierten Netzes während der Reproduktionsphase auch Wertepaare, die nicht in der Trainingsmenge enthalten sind, hinreichend genau zu repräsentieren. Hierbei ist speziell zwischen dem Interpolations- und Extrapolationsverhalten zu unterscheiden, welche die Approximationsfähigkeit des Netzes innerhalb bzw. außerhalb des Eingangsbereichs der Trainingsdaten charakterisieren.

Ein wichtiges Ziel ist die gute Verallgemeinerungsfähigkeit. Die Überprüfung der Verallgemeinerungsfähigkeit darf nicht mit dem Trainingsdatensatz erfolgen, sondern dazu muss ein neuer Datensatz verwendet werden. Der Grund dafür ist, dass bei zu langem Training mit dem gleichen Datensatz der Effekt des Übertrainierens (Overfitting) eintritt, d.h. das neuronale Netz spezialisiert sich durch *Auswendiglernen* auf die optimale Reproduktion des Trainingsdatensatzes, zeigt aber eine schlechte Verallgemeinerungsfähigkeit bei einem anderen Datensatz.

3.2.3 Funktionsapproximation mit lokalen Basisfunktionen

Beim Einsatz neuronaler Netze für regelungstechnische Aufgaben werden in der Regel kurze Adaptionzeiten sowie eine nachweisbare Stabilität und Konvergenz auch unter Störeinflüssen gefordert. Dies wird am besten von neuronalen Netzen erfüllt, die nach der Methode der stützwertbasierten Approximation arbeiten. Ein wesentliches Merkmal dieser Netze ist die Verwendung lokaler Basisfunktionen, die hier wie folgt definiert werden:

Definition 3.2: Lokale Basisfunktion

Eine zusammenhängende begrenzte und nicht-negative Funktion $\mathcal{B} : \mathbb{R}^P \rightarrow \mathbb{R}_{0+}$ sei eine lokale Basisfunktion, wenn sie ein globales Maximum bei $\mathbf{x} = \chi$ besitzt und wenn für alle Elemente x_i des n -dimensionalen Eingangsvektors $\mathbf{x}$ gilt

$$\frac{\partial \mathcal{B}(\mathbf{x})}{\partial x_i} \quad \begin{cases} \geq 0 & \text{für } x_i < \chi_i \\ \leq 0 & \text{für } x_i > \chi_i \end{cases}$$

und

$$\lim_{\|\mathbf{x}\| \rightarrow \infty} \mathcal{B}(\mathbf{x}) = 0$$

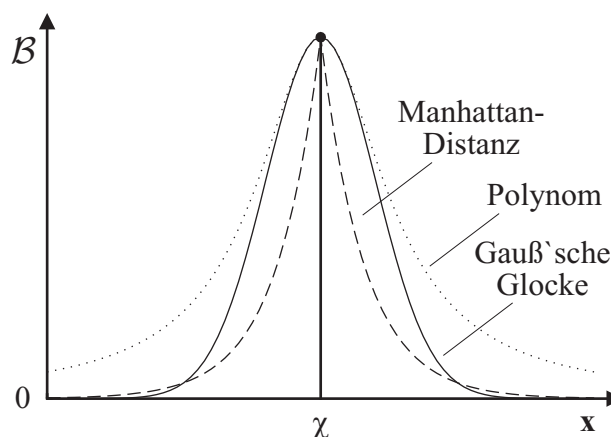


Bild 3.5: Beispiele lokaler Basisfunktionen

Beispiele lokaler Basisfunktionen sind die Gauß'sche Glockenkurve, die Manhattan-Distanz, Polynome wie z.B. $1/(1+x^2)$ oder ein Dreieckfenster; dabei bezeichnet χ das Zentrum der

Basisfunktion (siehe Bild 3.5). In [5] wird sogar das Rechteckfenster zu den Basisfunktionen gezählt. Ebenso können auch die im Rahmen der Fuzzy-Logik verwendeten Aktivierungsgrade formal als Basisfunktionen behandelt werden.

Um die oben definierten Basisfunktionen zur Funktionsapproximation mit neuronalen Netzen einsetzen zu können, müssen die universelle Einsetzbarkeit zur Nachbildung *beliebiger* Funktionen und eine konvergente Adaption gewährleistet sein.

Im Zusammenhang neuronaler Netze werden lokale Basisfunktionen als Aktivierungsfunktionen der Neuronen bzw. ihrer Gewichte eingesetzt. Eine Aktivierungsfunktion $\mathcal{A}_i$ sei allgemein eine lokale Basisfunktion $\mathcal{B}$ der vektoriellen Eingangsgröße $\mathbf{x}$ und eines Vektors χ_i , der die Lage des Zentrums und damit des Maximums der Aktivierungsfunktion im Eingangsraum angibt. Die einzelnen Aktivierungsfunktionen $\mathcal{A}_i$ werden nun zu einem Vektor $\mathcal{A}$ zusammengefasst

$$\mathcal{A}(\mathbf{x}) = [\mathcal{B}(\mathbf{x}, \chi_1) \ \mathcal{B}(\mathbf{x}, \chi_2) \ \dots \ \mathcal{B}(\mathbf{x}, \chi_p)]^T \quad (3.3)$$

Wird nun ein Gewichtsvektor Θ derselben Länge p aufgestellt,

$$\Theta = [\Theta_1 \ \Theta_2 \ \dots \ \Theta_p]^T \quad (3.4)$$

kann eine nichtlineare Funktion $\mathcal{NL}$ nach obiger Definition als Skalarprodukt aus Gewichts- und Aktivierungsvektor dargestellt werden. Diese Darstellung erscheint zunächst willkürlich und ohne physikalische Entsprechung gewählt. Sie dient aber im weiteren Verlauf der anschaulichen Darstellung der Adaption neuronaler Netze:

$$\mathcal{NL}: \quad y(\mathbf{x}) = \Theta^T \mathcal{A}(\mathbf{x}) + d(\mathbf{x}) \quad (3.5)$$

Die Auswahl der Aktivierungsfunktionen, ihrer Anzahl und Parameter muss dabei so möglich sein, dass der inhärente Approximationsfehler $d(\mathbf{x})$ eine beliebig kleine Schranke nicht überschreitet. Bei Verwendung von lokalen Basisfunktionen steht damit ein universeller Funktionsapproximator zur Verfügung. Der Approximationsfehler d wird im Folgenden vernachlässigt.

Die gleiche Darstellung kann nun auch für die durch ein neuronales Netz nachgebildete (d.h. „geschätzte“) Funktion $\widehat{\mathcal{NL}}$ verwendet werden.

$$\widehat{\mathcal{NL}}: \quad \hat{y}(\mathbf{x}) = \hat{\Theta}^T \mathcal{A}(\mathbf{x}) \quad (3.6)$$

Dabei wird angenommen, dass der Vektor $\mathcal{A}$ der Aktivierungsfunktionen mit dem Aktivierungsvektor bei der oben eingeführten Darstellung der betrachteten Nichtlinearität identisch ist. Damit kann ein Adaptionsfehler e eingeführt werden, der im weiteren auch als Lernfehler bezeichnet wird.

$$e(\mathbf{x}) = \hat{y}(\mathbf{x}) - y(\mathbf{x}) = \hat{\Theta}^T \mathcal{A}(\mathbf{x}) - \Theta^T \mathcal{A}(\mathbf{x}) = (\hat{\Theta}^T - \Theta^T) \mathcal{A}(\mathbf{x}) \quad (3.7)$$

Die Aufgabe der Adaption lässt sich so auf eine Anpassung der Gewichte reduzieren. Optimale Adaption bedeutet dann, dass der Gewichtsvektor $\hat{\Theta}$ des neuronalen Netzes gleich

dem Gewichtsvektor Θ der Nichtlinearität ist. Der Gewichtsvektor der Nichtlinearität entspricht einem optimal trainierten neuronalen Netz mit festen Stützwerten. Dies ist gleichbedeutend mit einem Verschwinden des Parameterfehlers Φ , der bei optimaler Adaption zu null wird.

$$\Phi = \hat{\Theta} - \Theta \quad (3.8)$$

Der Ausgangsfehler kann nun dargestellt werden als

$$e(\mathbf{x}) = \Phi^T \mathcal{A}(\mathbf{x}) = \mathcal{A}(\mathbf{x})^T \Phi \quad (3.9)$$

Bei ausreichender Variation des Eingangswerts $\mathbf{x}$ ermöglicht die lokale Wirksamkeit der Aktivierungsfunktionen im Eingangsraum die Eindeutigkeit der Adaption. Für Gauß'sche Radiale Basisfunktionen wurde die Eindeutigkeit der Funktionsdarstellung in [90, 73] nachgewiesen.

3.2.4 Radial Basis Function (RBF) Netz

Viele in der Regelungstechnik eingesetzten neuronalen Netze gehören zur Familie der Radial Basis Function (RBF) Netze. Im Gegensatz zu anderen neuronalen Ansätzen, wie Multi Layer Perceptron Netzen oder Kohonen-Netzen, weisen sie eine feste und lokale Zuordnung der einzelnen Neuronen zu Bereichen des Eingangsraums auf. Dies ermöglicht insbesondere eine physikalische Interpretierbarkeit der adaptierten Gewichte, die z.B. der Diagnose eines Prozesses dienen kann.

Im Folgenden werden die Gewichte der Neuronen, ihre Aktivierungsfunktionen und die zugehörigen Zentren auch unter dem Begriff *Stützwerte* zusammengefasst. Dabei bezeichnet der *Wert* eines Stützwerts das zugeordnete Gewicht Θ_i und die *Lage* (oder auch *Koordinate*) eines Stützwerts das Zentrum χ_i der zuständigen Aktivierungsfunktion. Der Ausgang $\hat{y}$ eines RBF-Netzes mit p Neuronen kann als gewichtete Summe der Aktivierungsfunktionen gebildet werden, wenn das Skalarprodukt aus Gleichung (3.6) in Summendarstellung übergeführt wird.

$$\hat{y}(\mathbf{x}) = \widehat{\mathcal{NL}}(\mathbf{x}) = \sum_{i=1}^p \hat{\Theta}_i \mathcal{A}_i(\mathbf{x}) \quad (3.10)$$

Üblicherweise werden als Aktivierungsfunktionen Gauß'sche Glockenkurven verwendet, deren Darstellung an die der Standardverteilung mit der Varianz σ^2 angeglichen ist [90].

$$\mathcal{A}_i = \exp\left(-\frac{\mathcal{C}_i}{2\sigma^2}\right) \quad (3.11)$$

Hier bezeichnet σ einen Glättungsfaktor, der den Grad der Überlappung zwischen benachbarten Aktivierungen bestimmt, und $\mathcal{C}_i$ das Abstandsquadrat des Eingangsvektors vom i -ten Stützwert, d.h. vom Zentrum χ_i der zugehörigen Aktivierungsfunktion.

$$\mathcal{C}_i = \|\mathbf{x} - \chi_i\|^2 = (\mathbf{x} - \chi_i)^T (\mathbf{x} - \chi_i) = \sum_{k=1}^n (x_k - \chi_{ik})^2 \quad (3.12)$$

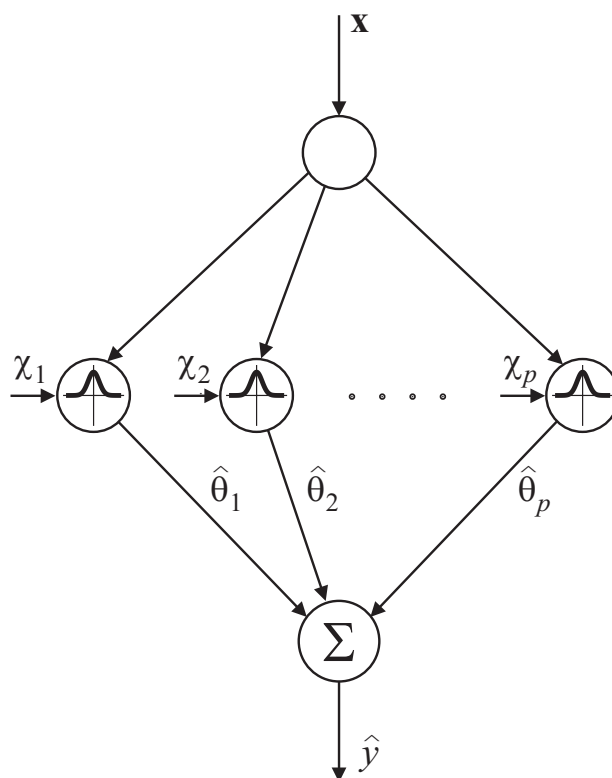


Bild 3.6: Struktur des RBF-Netzes

n gibt die Dimension des Eingangsvektors $\mathbf{x}$ an. Alternativ kann auch die so genannte Manhattan-Distanz für die Abstandsfunktion $\mathcal{C}_i$ verwendet werden:

$$\mathcal{C}_i = \sum_{k=1}^n |x_k - \chi_{ik}| \quad (3.13)$$

Damit kann die Struktur eines RBF-Netzes auch grafisch gemäß Bild 3.6 umgesetzt werden. Das erzielte Approximationsverhalten ist in Bild 3.7 an einem Beispiel dargestellt. Dabei fällt allerdings die ungünstige Approximation zwischen den Stützwerten und die ungünstige Extrapolation dieses Netzes aufgrund der fehlenden Monotonie-Erhaltung auf [54]. Dadurch kann der Wert der approximierten Funktion zwischen den Zentren zweier Aktivierungsfunktionen (z.B. χ_1 und χ_2) auch außerhalb des durch ihre Gewichte begrenzten Bereichs liegen, d.h. eine Monotonie der Gewichte bedingt nicht notwendigerweise auch einen monotonen Verlauf der approximierten Funktion. Ein weiterer Punkt ist das ungünstige Extrapolationsverhalten außerhalb des Stützwertebereichs. Hier sinkt der Ausgang des Netzes auf null ab (Bild 3.7). Da aber die Erhaltung von Monotonie und ein sinnvolles Extrapolationsverhalten der gelernten Kennlinie wesentliche Forderungen in regelungstechnischen Anwendungen sind, führt dies zu einer Modifikation des RBF-Netzes.

3.2.5 General Regression Neural Network (GRNN)

Das General Regression Neural Network (GRNN) stellt eine Weiterentwicklung des RBF-Netzes aus den oben genannten Gründen dar. Der wesentliche Unterschied besteht in einer

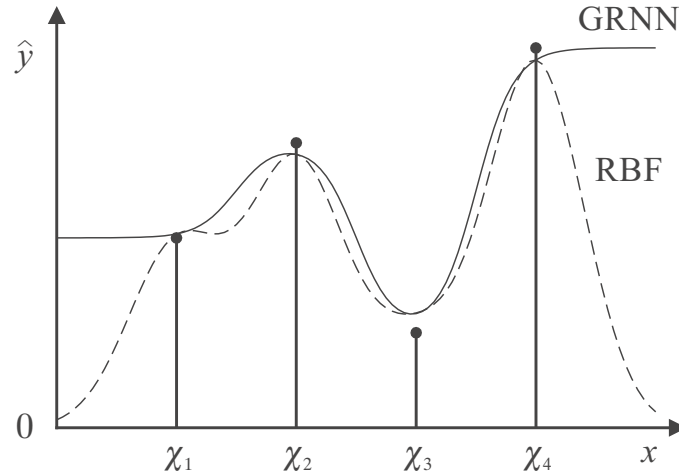


Bild 3.7: Vergleich der Approximation von RBF-Netz und GRNN

Normierung aller Aktivierungsfunktionen auf deren Summe, wie in Bild 3.8 dargestellt. Als Abstandsfunktion $\mathcal{C}_i$ kommt dabei das Abstandsqadrat nach Gleichung (3.12) zum Einsatz. Die Aktivierungsfunktionen ergeben sich damit zu

$$\mathcal{A}_i = \frac{\exp\left(-\frac{\mathcal{C}_i}{2\sigma^2}\right)}{\sum_{k=1}^p \exp\left(-\frac{\mathcal{C}_k}{2\sigma^2}\right)} \quad (3.14)$$

Damit gilt

$$\sum_{i=1}^p \mathcal{A}_i = 1 \quad (3.15)$$

Durch die Normierung wird sichergestellt, dass der Wert der approximierten Funktion stets innerhalb der durch den Wert der angrenzenden Stützpunkte gegebenen Grenzen verläuft und gleichzeitig eine Monotonie der Stützpunkte auch einen monotonen Verlauf der approximierten Funktion bewirkt. Diese Eigenschaft führt insbesondere auch zu einer verbesserten Extrapolation, bei der die approximierte Funktion dem jeweils nächstliegenden – und damit wahrscheinlichsten – Stützpunkt asymptotisch zustrebt. Zur besseren Vergleichbarkeit unterschiedlich parametrierter GRNN wird des weiteren ein normierter Glättungsfaktor σ_{norm} eingeführt, der auf den Abstand $\Delta\chi$ zweier Stützpunkte normiert ist.

$$\sigma_{norm} = \frac{\sigma}{\Delta\chi} \quad (3.16)$$

Die nachfolgenden Betrachtungen zu Lerngesetzen, Stabilität und Parameterkonvergenz gelten für RBF-Netze allgemein und damit auch für das daraus abgeleitete GRNN.

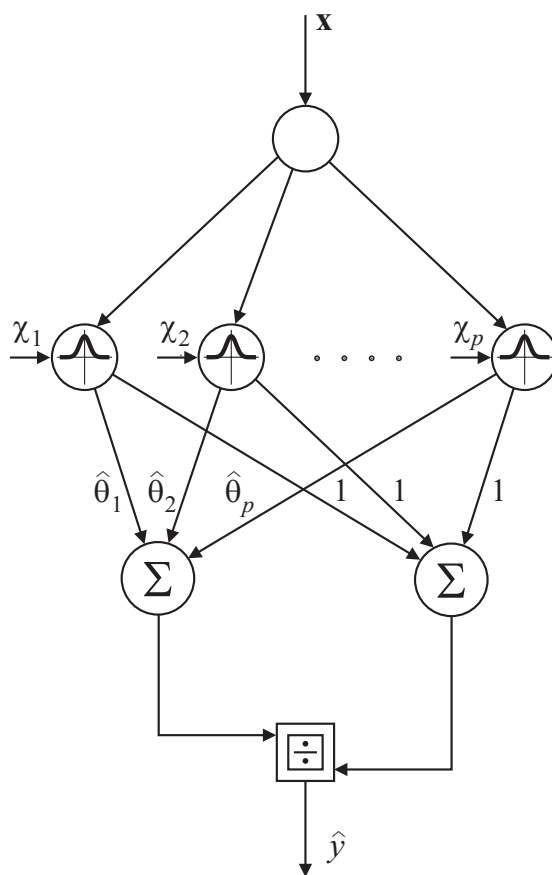


Bild 3.8: Struktur des GRNN

3.2.6 Lerngesetz

Ausgehend von der Gleichung für den Schätzwert $\hat{y}$ am Ausgang eines GRNN

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^p \hat{\Theta}_i \mathcal{A}_i(\mathbf{x}) = \hat{\Theta}^T \mathcal{A} \quad (3.17)$$

wird zur Adaption der Gewichte der bereits oben eingeführte Lernfehler e bzw. der daraus abgeleitete quadratische Fehler E herangezogen.

$$e(\mathbf{x}) = \sum_{i=1}^p \hat{\Theta}_i \mathcal{A}_i(\mathbf{x}) - y(\mathbf{x}) \quad (3.18)$$

$$E(\mathbf{x}) := \frac{1}{2} e^2 = \frac{1}{2} \left(\sum_{i=1}^p \hat{\Theta}_i \mathcal{A}_i(\mathbf{x}) - y(\mathbf{x}) \right)^2 \quad (3.19)$$

Die notwendige Änderung jedes Gewichts $\hat{\Theta}_i$ wird durch ein Gradientenabstiegsverfahren (auch Delta-Lernregel genannt [26]) festgelegt. Dazu wird der quadratische Fehler nach dem jeweiligen Gewicht abgeleitet.

$$\frac{dE(\mathbf{x})}{d\hat{\Theta}_i} = \left(\sum_{k=1}^p \hat{\Theta}_k \mathcal{A}_k(\mathbf{x}) - y(\mathbf{x}) \right) \mathcal{A}_i(\mathbf{x}) = e(\mathbf{x}) \mathcal{A}_i(\mathbf{x}) \quad (3.20)$$

Somit bestimmen sich die notwendigen Änderungen der Gewichte zueinander wie es dem Beitrag jedes Gewichts zum Schätzwert $\hat{y}$ und damit zum Lernfehler e entspricht. Eine zusätzliche Skalierung mit einem Lernfaktor η dient der Einstellung einer gewünschten Lerngeschwindigkeit bzw. Glättungswirkung bei der Adaption. Das negative Vorzeichen stellt eine Anpassung der Gewichte in Richtung kleinerer Fehler sicher. Das vollständige Lerngesetz für jedes Gewicht lautet damit

$$\frac{d}{dt} \hat{\Theta}_i = -\eta e \mathcal{A}_i \quad (3.21)$$

oder vektoriell für alle Gewichte gleichzeitig

$$\frac{d}{dt} \hat{\Theta} = -\eta e \mathcal{A} \quad (3.22)$$

Bei mehrschichtigen neuronalen Netzen ist dieses Verfahren auch als *Backpropagation* bekannt. Für RBF-Netze, die mit dem Gradientenlerngesetz aus Gleichung (3.22) adaptiert werden, erfolgt in Kap. 3.2.6.1 ein Stabilitätsnachweis, sowie die Definition geeigneter Anregungssignale, die Parameterkonvergenz von geschätzter und realer Nichtlinearität garantieren.

3.2.6.1 Stabilität nach Lyapunov

Die Stabilität eines Systems, das durch die nichtlineare Differentialgleichung

$$\frac{d}{dt} \mathbf{x} = \mathbf{f}(\mathbf{x}, t) \quad (3.23)$$

beschrieben wird, ist nach Lyapunov wie folgt definiert [17]:

Definition 3.3: Stabilität nach Lyapunov

Der Gleichgewichtszustand $\mathbf{x}_0$ des Systems in Gleichung (3.23) wird als **stabil** bezeichnet, wenn für jedes $\varepsilon > 0$ und $t_0 \geq 0$ ein δ existiert, so dass aus $\|\mathbf{x}_0\| < \delta$ folgt $\|\mathbf{x}(t, \mathbf{x}_0, t_0)\| < \varepsilon$ für alle $t \geq t_0$.

Anschaulich gesprochen folgt aus einer kleinen Störung stets eine kleine Abweichung vom Gleichgewichtszustand, bzw. die Funktion $\mathbf{f}$ bleibt nahe am Ursprung $\mathbf{0}$, wenn ihr Anfangswert nur mit genügend kleinem Abstand zum Ursprung gewählt wird. Bild 3.9 veranschaulicht die Stabilitätsdefinition nach Lyapunov.

Unter Verwendung des Parameterfehlers $\Phi = \hat{\Theta} - \Theta$ aus Gleichung (3.8) lässt sich die Stabilität des oben hergeleiteten Lerngesetzes nach Lyapunov beweisen. Für die Ableitung des Parameterfehlers gilt

$$\frac{d}{dt} \Phi = \frac{d}{dt} \hat{\Theta} = -\eta \mathcal{A} \left(\hat{\Theta}^T - \Theta^T \right) \mathcal{A} = -\eta \mathcal{A} \Phi^T \mathcal{A} \quad (3.24)$$

Mit der Beziehung aus Gleichung (3.9) lässt sich diese Gleichung umformen zu

$$\frac{d}{dt} \Phi = -\eta \mathcal{A} \Phi^T \mathcal{A} = -\eta \mathcal{A} \mathcal{A}^T \Phi \quad (3.25)$$

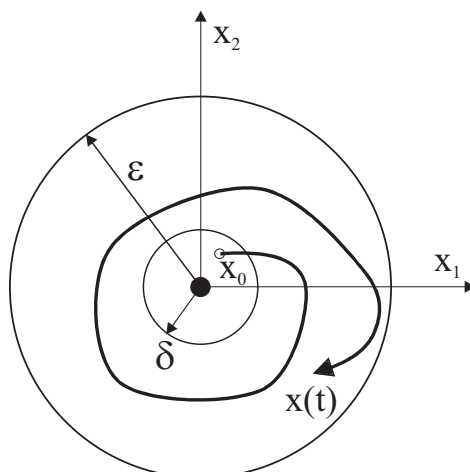


Bild 3.9: Stabilität nach Lyapunov

Als Lyapunov-Funktion V wird die positiv semidefinite Funktion

$$V(\Phi) = \frac{1}{2} \Phi^T \Phi \quad (3.26)$$

gewählt. Ihre zeitliche Ableitung entlang der durch Gleichung (3.24) festgelegten Trajektorien bestimmt sich mit positivem Lernfaktor η zu

$$\begin{aligned} \frac{d}{dt} V(\Phi) &= \frac{1}{2} \cdot 2 \cdot \frac{d}{dt} (\Phi^T) \Phi = (-\eta \mathcal{A} \mathcal{A}^T \Phi)^T \Phi = -\eta (\mathcal{A} \mathcal{A}^T \Phi)^T \Phi \quad (3.27) \\ &= -\eta \underbrace{\mathcal{A}^T \Phi}_e \underbrace{\mathcal{A}^T \Phi}_e = -\eta e^2 \leq 0 \end{aligned}$$

Damit ist dV/dt negativ semidefinit, wodurch die Beschränktheit des Parameterfehlers Φ und die Stabilität des obigen Systems nach Lyapunov gezeigt ist.

3.2.6.2 Asymptotische Stabilität und Parameterkonvergenz

Neben der Stabilität nach Lyapunov ist auch die *asymptotische* Stabilität des betrachteten Lernvorgangs von Interesse, da erst diese die Parameterkonvergenz und damit eine sinnvolle Interpretierbarkeit der adaptierten Gewichte ermöglicht.

Definition 3.4: Asymptotische Stabilität nach Lyapunov

Der Gleichgewichtszustand $\mathbf{x}_0$ des Systems in Gleichung (3.23) wird als **asymptotisch stabil** bezeichnet, wenn er stabil ist und ein $\delta > 0$ existiert, so dass aus $\|\mathbf{x}\| < \delta$ folgt $\lim_{t \rightarrow \infty} \mathbf{x}(t) = \mathbf{0}$.

Für diesen Nachweis ist eine ausreichende Anregung (*Persistent Excitation*) notwendig. Eine solche Anregung liegt vor, wenn für alle Einheitsvektoren $\mathbf{v}_i$ in $\mathbb{R}^p$ und positives ε_0

und t_0 ein endliches Zeitintervall T gefunden werden kann, so dass gilt

$$\frac{1}{T} \int_t^{t+T} |\mathcal{A}^T(\mathbf{x}) \mathbf{v}_i| d\tau \geq \varepsilon_0 \quad \text{für alle} \quad t \geq t_0 \quad (3.28)$$

Anschaulich gesprochen heißt dies, dass die Aktivierung für jedes Neuron nie dauerhaft auf den Einheitsvektoren $\mathbf{v}_i$ senkrecht stehen darf, so dass sich jeder Parameterfehler stets über den Lernfehler e auswirkt und dV/dt somit negativ definit ist, solange keine vollständige Parameterkonvergenz erreicht ist.

Damit strebt die gewählte Lyapunov-Funktion V asymptotisch zu null und damit auch der Parameterfehler.

$$\lim_{t \rightarrow \infty} V(t) = 0 \quad (3.29)$$

$$\lim_{t \rightarrow \infty} \Phi(t) = \mathbf{0} \quad (3.30)$$

Dadurch wird bei ausreichender Anregung die Konvergenz der Parameter erreicht:

$$\lim_{t \rightarrow \infty} \hat{\Theta}(t) = \Theta \quad (3.31)$$

3.2.6.3 Automatische Einstellung der Lernschrittweite

Über die Lernschrittweite η wurde bisher nur ausgesagt, dass sie für stabile Adaption positiv sein muss. Welcher Wert eine möglichst schnelle Konvergenz vor allem bei zeitdiskreter Realisierung in einem Digitalrechner garantiert, bleibt zunächst offen. Es handelt sich bei der Lernschrittweite um einen Designparameter, der durch Simulationsversuche und *trial and error* eingestellt wird. In den folgenden Abschnitten soll diese Vorgehensweise durchbrochen und eine automatische und in gewissem Sinne optimale Einstellung der Lernschrittweite erarbeitet werden. Die Optimalität bezieht sich dabei auf die folgenden beiden Annahmen:

- Es wird angenommen, der Eingangsvektor $\mathbf{x}$ sei konstant oder ändere sich nur langsam.
- Bei zeitdiskreter Realisierung des Lerngesetzes in Gleichung (3.22) soll die Lernschrittweite genau so gewählt werden, dass beim nächsten Integrationsschritt der Ausgangsfehler e bei konstantem Eingangsvektor $\mathbf{x}$ ein Minimum erreicht. η soll so gewählt werden, dass jede andere Wahl zu einem größeren Ausgangsfehler führen würde.

Zur Bestimmung einer optimalen Lernschrittweite wird das Prinzip der Linienoptimierung für das Gradientenverfahren angewendet [24]. Die zu minimierende Funktion lautet:

$$E(\hat{\Theta}, \mathbf{x}) = \frac{1}{2} e^2 = \frac{1}{2} \left(\hat{\Theta}^T \mathcal{A}(\mathbf{x}) - y(\mathbf{x}) \right)^2 \quad (3.32)$$

Die freien, zu optimierenden Variablen sind die Netzwerkgewichte $\hat{\Theta}$. Es ist zu beachten, dass die zu minimierende Zielfunktion (3.32) sowohl von den Netzwerkgewichten als auch

vom Eingangsvektor $\mathbf{x}$ des GRNN abhängt. Da der Eingangsvektor $\mathbf{x}$ nicht beeinflussbar ist, wird er zunächst bei jedem Minimierungsschritt als konstant angenommen.

Da das GRNN und das zugehörige Lerngesetz in einem Digitalrechner zeitdiskret realisiert werden, muss zunächst das Lerngesetz zeitdiskret approximiert werden. Die zeitliche Differentiation von $\hat{\Theta}$ in Gleichung (3.22) wird durch den Differenzenquotienten ersetzt. Der Hochindex k bezeichnet den aktuellen Zeitschritt, der Hochindex $^{k+1}$ den darauffolgenden Zeitschritt. Das Adaptionsgesetz für die Netzwerkgewichte lautet mit der Abtastzeit T_{ab} und Rechteckapproximation:

$$\frac{\hat{\Theta}^{k+1} - \hat{\Theta}^k}{T_{ab}} = -\eta e^k \mathcal{A} \quad (3.33)$$

Dabei ist der aktuelle Fehler definiert zu

$$e^k = \left(\hat{\Theta}^k \right)^T \mathcal{A} - y^k \quad (3.34)$$

und das aktuelle Fehlermaß ergibt sich zu

$$E^k = \frac{1}{2} (e^k)^2 \quad (3.35)$$

Das Fehlermaß E^k aus Gleichung (3.35) ist die mittels Gradientenverfahren und Linienoptimierung zu minimierende Funktion. Als Suchrichtung $\mathbf{s}$ wird der steilste Abstieg (negative Gradient, siehe auch Gleichung (3.20)) gewählt.

$$\mathbf{s} = -\frac{\partial E(\hat{\Theta}^k)}{\partial \hat{\Theta}^k} = -e^k \cdot \frac{\partial e^k}{\partial \hat{\Theta}^k} = -e^k \cdot \mathcal{A} \quad (3.36)$$

Es gilt nun zu klären, wie groß η gewählt werden muss, damit entlang der bereits festgelegten Richtung $\mathbf{s}$ das Fehlermaß E minimal wird. Es handelt sich also um ein eindimensionales Minimierungsproblem entlang der Richtung $\mathbf{s}$; dies wird als Linienoptimierung bezeichnet [24]. Die optimale Adaptionsschrittweite η zum aktuellen Zeitschritt ergibt sich aus der Lösung der folgenden Optimierungsaufgabe.

$$\min_{\eta} E(\eta) = \min_{\eta} E \left(\hat{\Theta}^k + \eta \mathbf{s} \right) = \min_{\eta} E \left(\underbrace{\hat{\Theta}^k - \eta e^k \mathcal{A}}_{\hat{\Theta}^{k+1}} \right) \quad (3.37)$$

Aus Gleichung (3.37) folgt zugleich der Wert des Fehlermaßes E^{k+1} bei Verwendung der optimalen Schrittweite η zum Zeitpunkt $k + 1$.

$$E(\eta) = E^{k+1} \quad (3.38)$$

Aus Gleichung (3.37) und (3.38) wird deutlich, dass das Fehlermaß zum jeweils nächsten Abtastschritt minimiert wird. Zunächst muss die notwendige Bedingung $dE(\eta)/d\eta = 0$ am Linienminimum erfüllt sein. Dazu wird $E(\eta)$ ausführlicher angeschrieben.

$$E(\eta) = \frac{1}{2} \left[\left((\hat{\Theta}^k)^T - \eta e \mathcal{A}^T \right) \cdot \mathcal{A} - y(\mathbf{x}) \right]^2 = \frac{1}{2} \left[(\hat{\Theta}^k)^T \cdot \mathcal{A} - \eta e \mathcal{A}^T \cdot \mathcal{A} - y(\mathbf{x}) \right]^2 \quad (3.39)$$

Die partielle Ableitung nach η und Nullsetzen ergibt

$$\frac{dE(\eta)}{d\eta} = \left[(\hat{\Theta}^k)^T \cdot \mathcal{A} - \eta e^k \mathcal{A}^T \cdot \mathcal{A} - y(\mathbf{x}) \right] \cdot (-e^k \cdot \mathcal{A}^T \mathcal{A}) \stackrel{!}{=} 0 \quad (3.40)$$

Für $e^k = 0$ ist auch das Fehlermaß E^k gleich null. Eine Adaption ist dann nicht mehr nötig. Dies äußert sich auch in der Suchrichtung $\mathbf{s} = -e^k \mathcal{A} = 0$. Die Adaptionsschrittweite η kann für $e^k = 0$ beliebig gewählt werden. Für $e^k \neq 0$ ist der letzte Term in Gleichung (3.40) ungleich null. Mit Gleichung (3.34) muss dann gelten

$$e^k \cdot (1 - \eta \mathcal{A}^T \mathcal{A}) = 0 \quad \Rightarrow \quad \eta = \eta_{opt} = \frac{1}{\mathcal{A}^T \mathcal{A}} \quad (3.41)$$

Ob es sich bei dem Extremwert des Fehlermaßes $E(\eta)$ in Gleichung (3.41) tatsächlich um ein Minimum handelt, muss mit der hinreichenden Bedingung zweiter Ordnung geklärt werden. Dazu muss die zweite Ableitung von E am potentiellen Minimum größer null sein.

$$\frac{d^2 E(\eta)}{(d\eta)^2} = \frac{d}{d\eta} [(e^k - \eta e^k \mathcal{A}^T \mathcal{A}) \cdot (-e^k \mathcal{A}^T \mathcal{A})] = (e^k)^2 \cdot (\mathcal{A}^T \mathcal{A})^2 \geq 0 \quad (3.42)$$

Damit ist gezeigt, dass die Wahl von η nach Gleichung (3.41) tatsächlich das einzige Minimum darstellt. Gleichung (3.41) stellt demnach eine Vorschrift zur automatischen und optimalen Wahl der Lernschrittweite dar.

Die bisherigen Ausführungen bezogen sich auf die zeitdiskrete Realisierung des Adaptionsgesetzes für die Netzwerkgewichte. Da in vielen Regelungssystemen die Algorithmen zeitkontinuierlich programmiert werden und die Diskretisierung durch eine Auswahl eines Integrationsalgorithmus erfolgt, ist ein Zusammenhang zwischen der Lernschrittweite der zeitdiskreten Realisierung und der Lernschrittweite des zeitkontinuierlichen Lerngesetzes nötig. Durch Vergleich von Gleichung (3.33) der zeitdiskreten Approximation und Gleichung (3.37) der zeitdiskreten Realisierung des Lerngesetzes, ist folgender Zusammenhang abzulesen.

$$\eta_{kont} = \frac{1}{T_{ab}} \eta = \frac{1}{T_{ab} \cdot \mathcal{A}^T \mathcal{A}} \quad (3.43)$$

Die Minimierung des Fehlermaßes erfolgte unter der Annahme eines konstanten Eingangsvektors $\mathbf{x}$ und damit auch eines konstanten Aktivierungsvektors $\mathcal{A}$. Bei variablem Eingangsvektor $\mathbf{x}$ müsste in Gleichung (3.39) zwischen dem aktuellen und zukünftigen Eingangsvektor unterschieden werden ($\mathbf{x}^k$ und $\mathbf{x}^{k+1}$). Da zukünftige Eingangssignale nicht verfügbar sind, bleiben nur zwei Möglichkeiten: Man führt die Einstellung der Lernschrittweite wie oben beschrieben durch und ist sich der Tatsache bewusst, dass man wegen des variablen Eingangssignals nur eine suboptimale Lerngeschwindigkeit erzielt. Als zweite Möglichkeit könnte man das zukünftige Eingangssignal aus vergangenen Signalen extrapolieren (z.B. linear, quadratisch oder Splines). Da es sich dabei aber ebenfalls um eine Schätzung handelt, scheint der zusätzliche Aufwand nicht gerechtfertigt, zumal Simulationen gezeigt haben, dass auch ohne Extrapolation sehr gute Ergebnisse erzielt werden (siehe Kap. 3.4). Wenn man von einer ausreichend kleinen Abtastzeit T_{ab} bei der Realisierung im Digitalrechner ausgeht, ändert sich der Eingangsvektor $\mathbf{x}$ von einem zum nächsten Zeitschritt nur geringfügig. Es liegt daher nahe, die Einstellungen für die Lernschrittweite gemäß der Gl. (3.41) und (3.43) auch im Falle zeitvarianter Eingangsgrößen $\mathbf{x}$ zu verwenden.

Bei der Wahl anderer Integrationsalgorithmen im Digitalrechner (z.B. Runge-Kutta-Verfahren) anstelle von Gleichung (3.33), müsste dieses Integrationsverfahren bei der Linienoptimierung berücksichtigt werden. Auch dies ist in der Praxis nicht nötig. Vielmehr hat sich folgendes Vorgehen bewährt: Die Lernschrittweite η wird zunächst nach Gleichung (3.41) bzw. (3.43) eingestellt. Aufgrund von Messfehlern und Rauscheinflüssen ergibt sich normalerweise ein relativ unruhiges Lernergebnis. Dies lässt sich durch Gewichtung des theoretisch optimalen η mit einem Faktor zwischen 0 und 1 vermeiden. Gleichung (3.41) stellt somit eine erste Schätzung der Lernschrittweite dar, die in der Praxis dann durch einen zusätzlichen Gewichtungsfaktor an die benötigten Anforderungen angepasst wird.

3.2.6.4 Lernstruktur und Fehlermodelle

Zur Adaption der betrachteten RBF-Netze und des GRNN sind je nach Anwendung verschiedene Lernstrukturen möglich. Da es im vorliegenden Fall um die möglichst identische Nachbildung einer Nichtlinearität geht, wird im Weiteren mit einer Vorwärtslernstruktur nach Bild 3.10 gearbeitet. Bei dieser werden die Nichtlinearität und das neuronale Netz parallel mit der identischen Eingangsgröße x betrieben; der Lernfehler e zwischen dem tatsächlichen Ausgang y und dem geschätzten Ausgang $\hat{y}$ steuert die Adaption. Alternativen zu dieser Lernstruktur, wie die inverse Lernstruktur, die ein inverses Modell der Strecke abbildet, werden daher nicht betrachtet. Ausgehend von dem oben hergeleiteten Lernge-

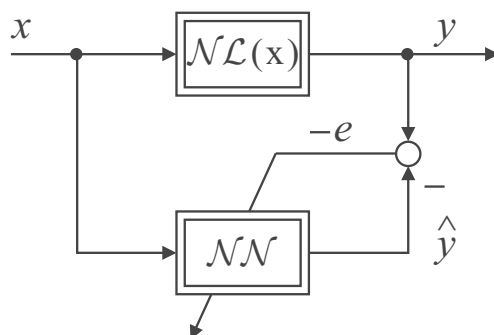


Bild 3.10: Vorwärtslernstruktur

setz werden in der Literatur verschiedene Fehlermodelle unterschieden, deren Einsatz im Wesentlichen davon bestimmt ist, ob der messbare Lernfehler direkt oder nur indirekt bzw. verzögert vorliegt. Als Grundlage für die nachfolgenden Ausführungen werden nun die verwendeten Lernstrukturen und Fehlermodelle vorgestellt [17].

Fehlermodell 1

Bei direktem Vorliegen des Lernfehlers e kann die Struktur des so genannten Fehlermodells 1 eingesetzt werden, wie in Bild 3.11 gezeigt. Die Nichtlinearität der Strecke wird dabei aus Gründen der Anschaulichkeit als Skalarprodukt nach Gleichung (3.5) dargestellt. Als Lerngesetz kann direkt Gleichung (3.22) verwendet werden.

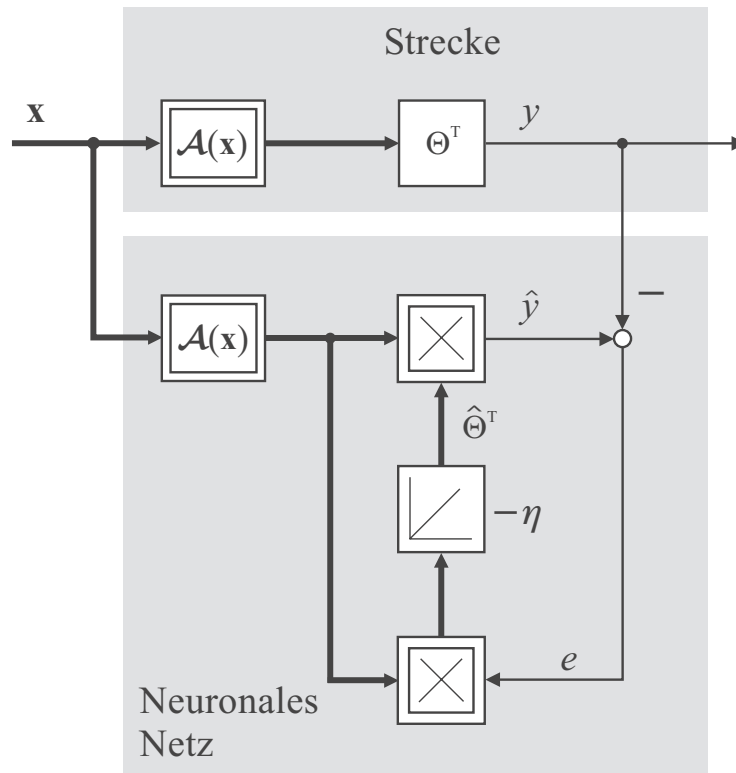


Bild 3.11: Identifikationsstruktur nach Fehlermodell 1

Fehlermodell 2

Fehlermodell 2 ist ein Spezialfall von Fehlermodell 4. Alle Zustandsgrößen der Fehlerübertragungsfunktion $H(s)$ müssen messbar sein. Da dieser Sonderfall nur sehr selten zutrifft und Fehlermodell 2 im allgemeiner gefassten Fehlermodell 4 eingeschlossen ist, soll auf die Darstellung von Fehlermodell 2 verzichtet werden [17].

Fehlermodell 3

Das Lerngesetz nach Fehlermodell 1 ist auch dann noch zulässig, wenn der Ausgangswert y , bedingt durch das Messverfahren oder einen Beobachter eine lineare und asymptotisch stabile SPR-Übertragungsfunktion¹⁾ $H(s)$ durchläuft. Die Lernstruktur einschließlich einer SPR-Übertragungsfunktion im Ausgangspfad wird als Fehlermodell 3 [17] bezeichnet und ist in Bild 3.12 dargestellt. Eine Übertragungsfunktion ist *Strictly Positive Real*, wenn sie die folgende Definition erfüllt:

Definition 3.5: SPR-Übertragungsfunktion

Eine Übertragungsfunktion $H(s)$ ist eine streng positiv reelle (SPR-) Übertragungsfunktion, wenn gilt:

1. $H(s)$ ist asymptotisch stabil, d.h. alle Pole besitzen negativen Realteil.

¹⁾ Strictly Positive Real

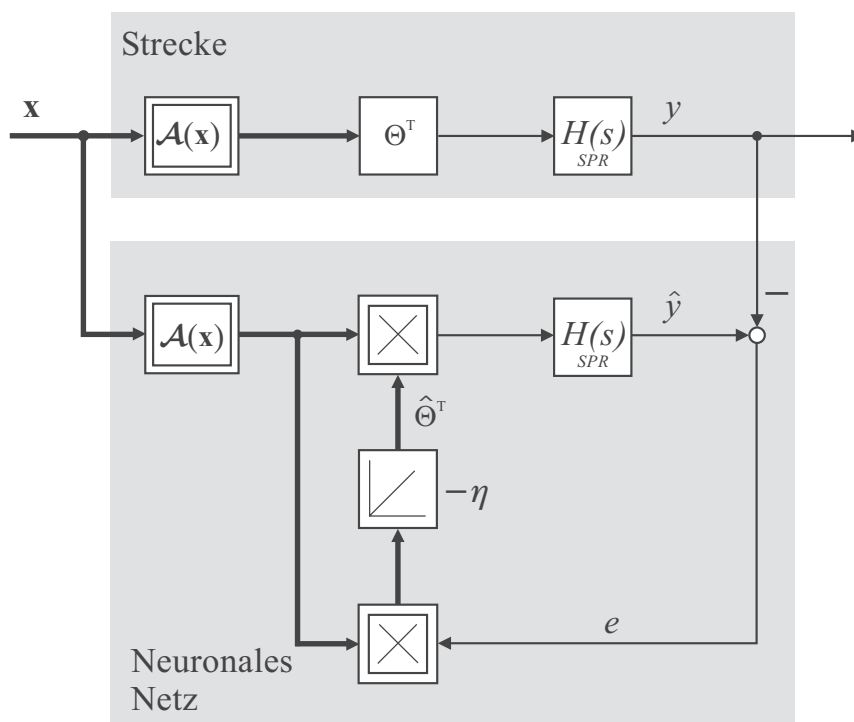


Bild 3.12: Identifikationsstruktur nach Fehlermodell 3

2. Der Realteil von $H(s)$ längs der $j\omega$ -Achse ist stets positiv, d.h. $\Re\{H(j\omega)\} > 0$ für alle $\omega \geq 0$.

Diese Definition bedeutet anschaulich, dass eine SPR-Übertragungsfunktion im Bode-Diagramm keine Phasendrehung größer 90° erzeugen darf.

Die Stabilität von Fehlermodell 3 wird nun durch eine geeignete Lyapunov-Funktion nachgewiesen. Die inneren Zustände der Übertragungsfunktion $H(s)$ der realen Strecke und des neuronalen Netzes seien mit x_H und $\hat{x}_H$ bezeichnet. Der Zustandsfehler e_x zwischen Strecke und neuronalem Netz ergibt sich zu

$$e_x = \hat{x}_H - x_H \quad (3.44)$$

Wenn die Fehlerübertragungsfunktion mit passenden Systemmatrizen $\mathbf{A}$, $\mathbf{b}$ und $\mathbf{c}$ in Zustandsdarstellung beschrieben wird, ergeben sich für neuronales Netz und Strecke die folgenden Differentialgleichungen:

$$\dot{\mathbf{x}}_H = \mathbf{A}\mathbf{x}_H + \mathbf{b}\Theta^T\mathcal{A}(x) \quad y = \mathbf{c}^T\mathbf{x}_H \quad (3.45)$$

$$\dot{\hat{\mathbf{x}}}_H = \mathbf{A}\hat{\mathbf{x}}_H + \mathbf{b}\hat{\Theta}^T\mathcal{A}(x) \quad \hat{y} = \mathbf{c}^T\hat{\mathbf{x}}_H \quad (3.46)$$

Die Differentialgleichung des Zustandsfehlers e_x lautet mit dem Parameterfehler Φ :

$$\dot{e}_x = \mathbf{A}e_x + \mathbf{b}\Phi^T\mathcal{A}(x) \quad (3.47)$$

Als Adaptionsgesetz für die Stützwerte wird die bereits aus Kap. 3.2.6 bekannte Gleichung verwendet (für den Beweis ohne Lernschrittweite η).

$$\dot{\Phi} = \dot{\Theta} = -e \mathcal{A} \quad (3.48)$$

Als positiv definite Lyapunov Funktion wird folgender Ausdruck verwendet:

$$V(\mathbf{e}_x, \Phi) = \mathbf{e}_x^T \mathbf{P} \mathbf{e}_x + \Phi^T \Phi \quad (3.49)$$

Die zeitliche Ableitung von V entlang der durch die Gleichungen (3.47) und (3.48) beschriebenen Trajektorien ergibt sich zu

$$\dot{V} = \dot{\mathbf{e}}_x^T \mathbf{P} \mathbf{e}_x + \mathbf{e}_x^T \mathbf{P} \dot{\mathbf{e}}_x - 2 \Phi^T \mathcal{A} e \quad (3.50)$$

Mit der Zwischenrechnung

$$\dot{\mathbf{e}}_x^T = \mathbf{e}_x^T \mathbf{A}^T + \mathcal{A}^T \Phi \mathbf{b}^T \quad (3.51)$$

lässt sich $\dot{V}$ weiter berechnen zu

$$\begin{aligned} \dot{V} &= (\mathbf{e}_x^T \mathbf{A}^T + \mathcal{A}^T \Phi \mathbf{b}^T) \mathbf{P} \mathbf{e}_x + \mathbf{e}_x^T \mathbf{P} (\mathbf{A} \mathbf{e}_x + \mathbf{b} \Phi^T \mathcal{A}) - 2 \Phi^T e \mathcal{A} \\ &= \mathbf{e}_x^T (\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A}) \mathbf{e}_x + \mathcal{A}^T \Phi \mathbf{b}^T \mathbf{P} \mathbf{e}_x + \mathbf{e}_x^T \mathbf{P} \mathbf{b} \Phi^T \mathcal{A} - 2 \Phi^T e \mathcal{A} \\ &= \mathbf{e}_x^T (\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A}) \mathbf{e}_x + 2 \mathbf{e}_x^T \mathbf{P} \mathbf{b} \Phi^T \mathcal{A} - 2 \Phi^T e \mathcal{A} \end{aligned} \quad (3.52)$$

Das Kalman-Yakubovich-Lemma [17] besagt, dass für eine streng positiv reelle Übertragungsfunktion $H(s) = \mathbf{c}^T (s\mathbf{E} - \mathbf{A})^{-1} \mathbf{b}$ positiv definite Matrizen $\mathbf{P}$ und $\mathbf{Q}$ existieren, für die gilt

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} = -\mathbf{Q} \quad (3.53)$$

$$\mathbf{P} \mathbf{b} = \mathbf{c} \quad (3.54)$$

Wählt man die Matrizen $\mathbf{P}$ und $\mathbf{Q}$ gemäß den Gleichungen (3.53) und (3.54), so folgt daraus schließlich

$$\dot{V} = -\mathbf{e}_x^T \mathbf{Q} \mathbf{e}_x \leq 0 \quad (3.55)$$

Damit ist die globale Stabilität gemäß Lyapunov von Fehlermodell 3 bewiesen. Asymptotische Stabilität (d.h. $V \rightarrow 0$ für $t \rightarrow \infty$) und damit auch Parameterkonvergenz liegt nur bei ausreichender Anregung des Eingangs des neuronalen Netzes vor (siehe Kap. 3.2.6.2).

Identifikation verallgemeinert (Fehlermodell4)

Die Verallgemeinerung des obigen Fehlermodells 3 entsteht, wenn die SPR-Bedingung an die Übertragungsfunktion $H(s)$ fallengelassen wird. Für eine stabile Identifikation muss die Lernstruktur dann zum so genannten Fehlermodell 4 nach [17] erweitert werden, wie in Bild 3.13 gezeigt. Für ein phasenrichtiges Lernen des neuronalen Netzes muss zum einen der Schätzwert $\hat{y}$ in gleicher Weise verzögert werden, wie der gemessene Ausgangswert y . Andererseits darf diese Verzögerung aus Gründen der Stabilität nicht mehr im Ausgangszweig des neuronalen Netzes eingebracht werden. Um diesen Konflikt zu lösen, wird die

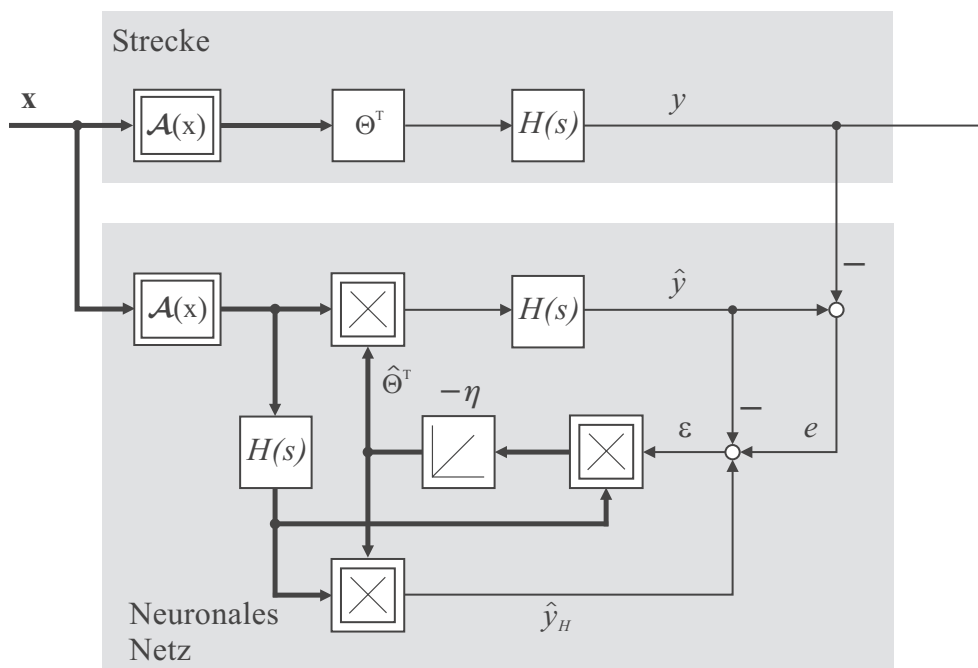


Bild 3.13: Erweiterte Identifikationsstruktur (Fehlermodell 4)

Linearität der Schätzwertbildung $\hat{\Theta}^T \mathcal{A}$ genutzt und die Verzögerung mit der linearen und asymptotisch stabilen Übertragungsfunktion $H(s)$ in die Aktivierung vorverlagert, so dass für den verzögerten Hilfsschätzwert $\hat{y}_H$ gilt

$$\hat{y}_H(\mathbf{x}) = \sum_{i=1}^p \hat{\Theta}_i H(s) \mathcal{A}_i(\mathbf{x}) = \hat{\Theta}^T H(s) \mathbf{E} \mathcal{A}(\mathbf{x}) \quad (3.56)$$

Dabei muss für jedes Element der Aktivierung $\mathcal{A}$ der Ausgang der Übertragungsfunktion $H(s)$ separat berechnet werden. Die dabei eingeführte hybride Notierung mit der gleichzeitigen Verwendung von Größen im Zeit- und Frequenzbereich ist an [17] angelehnt und dient der Vereinfachung der Schreibweise; $\mathbf{E}$ bezeichnet die Einheitsmatrix.

Der Lernfehler ϵ lässt sich nun wie folgt vereinfachen.

$$\begin{aligned} \epsilon(\mathbf{x}) &= \hat{\Theta}^T H(s) \mathbf{E} \mathcal{A}(\mathbf{x}) - H(s) \Theta^T \mathcal{A}(\mathbf{x}) \\ &= \left(\hat{\Theta}^T - \Theta^T \right) \underbrace{H(s) \mathbf{E} \mathcal{A}(\mathbf{x})}_{\text{verzögerte Akt.}} = \Phi^T H(s) \mathbf{E} \mathcal{A}(\mathbf{x}) \end{aligned} \quad (3.57)$$

Die Umformung für den Ausgangswert y ist zulässig, da der Parametervektor Θ der Strecke konstant ist. Durch Einführung einer so genannten *verzögerten Aktivierung* kann der Lernfehler analog zu Gleichung (3.7) vereinfacht und das Lerngesetz angepasst werden:

$$\frac{d}{dt} \Phi = \frac{d}{dt} \hat{\Theta} = -\eta \epsilon(\mathbf{x}) H(s) \mathbf{E} \mathcal{A}(\mathbf{x}) = -\eta \epsilon(\mathbf{x}) \zeta \quad (3.58)$$

Mit der Definition der verzögerten Aktivierung $\zeta = H(s) \mathbf{E} \mathcal{A}(\mathbf{x})$ wird auch für Fehlermodell

4 die Stabilität nach Lyapunov gezeigt. Als positiv definite Lyapunov-Funktion wird

$$V(\Phi) = \frac{1}{2} \Phi^T \Phi \quad (3.59)$$

vorgeschlagen. Die zeitliche Ableitung von V entlang der durch Gleichung (3.58) definierten Trajektorien ergibt:

$$\dot{V} = \Phi^T \dot{\Phi} = -\eta \epsilon(\mathbf{x}) \Phi^T \zeta = -\eta \epsilon(\mathbf{x})^2 \leq 0 \quad (3.60)$$

Damit ist $\dot{V}$ negativ semidefinit, wodurch die Beschränktheit des Parameterfehlers Φ und damit die Stabilität des Fehlermodells 4 nach Lyapunov gezeigt ist. Die Lyapunov-Funktion (3.59) nimmt so lange ab, bis ihre Ableitung null geworden ist. Dies ist der Fall, wenn in Gleichung (3.60) der Fehler ϵ zu null geworden ist. Es liegt also sicher Fehlerkonvergenz vor. Wenn die verzögerte Aktivierung ζ ausreichend anregend ist, so folgt nach Kap. 3.2.6.2 neben Fehlerkonvergenz auch Parameterkonvergenz und damit auch asymptotische Stabilität.

$$\lim_{t \rightarrow \infty} \Phi = \mathbf{0} \quad \Rightarrow \quad \lim_{t \rightarrow \infty} \hat{\Theta} = \Theta \quad (3.61)$$

3.2.6.5 Approximation einer eindimensionalen Nichtlinearität

Es wird nun die Approximationsgüte des GRNN für eine eindimensionale Nichtlinearität untersucht. Insbesondere ist das Interpolationsverhalten bei unterschiedlichen Stützwertezahlen und Glättungsfaktoren σ von Interesse.

Als Nichtlinearität wird folgende Kennlinie verwendet:

$$\mathcal{NL}(x) = 3 \cdot \arctan(2 \cdot x) \quad (3.62)$$

Als Lernverfahren wird jeweils das Gradientenabstiegsverfahren ohne Fehlerübertragungsfunktion verwendet (Fehlermodell 1). Das Training erfolgt so lange, bis keine Verbesserung des Ergebnisses mehr eintritt.

Zunächst wird die Nichtlinearität mit einem GRNN mit 11 Stützwerten und einem normierten Glättungsfaktor $\sigma_{norm} = 0.5$ approximiert. In Bild 3.14 links erkennt man, dass zwischen Funktionsvorgabe und Identifikationsergebnis nahezu kein Unterschied erkennbar ist. Die Dimensionierung des Netzes ist demnach gut gewählt. Der Glättungsfaktor σ bestimmt die Breite der Aktivierungsfunktionen und damit den Bereich der lokalen Wirkung eines Stützwerts. Eine Erhöhung von σ bedeutet eine Ausweitung der lokalen Stützwertewirkung, so dass für $\sigma \rightarrow \infty$ schließlich nur noch der Mittelwert der zu approximierenden Funktion nachgebildet werden kann. Bild 3.15 zeigt das Lernergebnis und die Aktivierungsfunktionen für 11 Stützwerte und $\sigma_{norm} = 1.5$. Die stärkere Überlappung der einzelnen Aktivierungsfunktionen bewirkt eine stärkere Glättung bzw. Mittelwertbildung der gelernten Kennlinie. Im steilen Bereich der arctan-Funktion wird dadurch der Approximationsfehler größer (Bild 3.15 links).

Eine Verringerung des Glättungsfaktors wirkt sich in einer geringeren Überlappung der lokalen Wirkungsbereiche der jeweiligen Stützstellen aus. In Bild 3.16 rechts ist fast keine Überlappung mehr vorhanden, so dass sich quasi ein *harte* Umschaltung zwischen den Werten der Stützstellen ergibt.

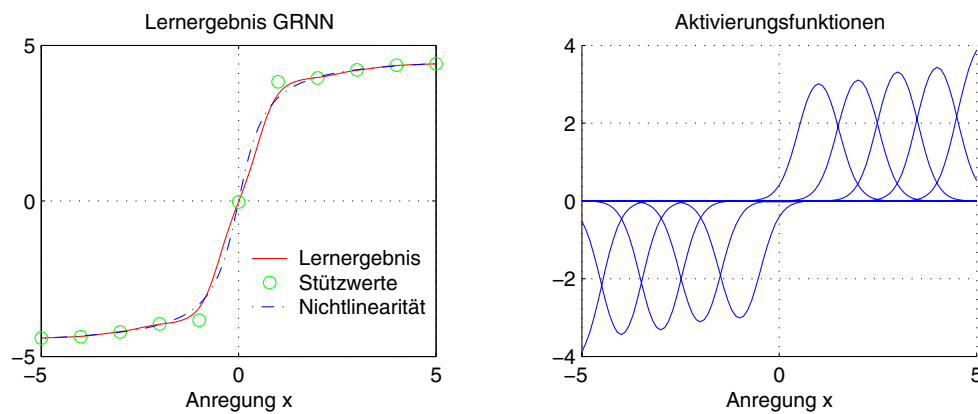


Bild 3.14: Gelernte Kennlinie (-) und wahre Kennlinie (.-) (links) und Aktivierungsfunktionen (rechts) für $\sigma_{norm} = 0.5$

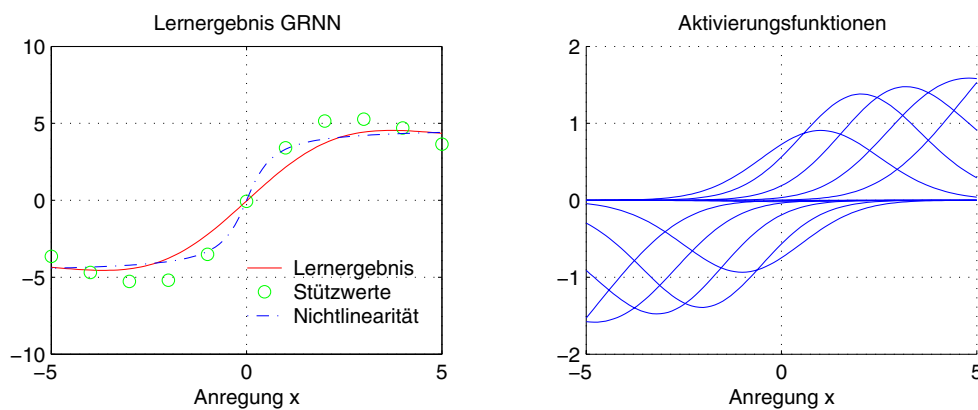


Bild 3.15: Gelernte Kennlinie (-) und wahre Kennlinie (.-) (links) und Aktivierungsfunktionen (rechts) für $\sigma_{norm} = 1.5$

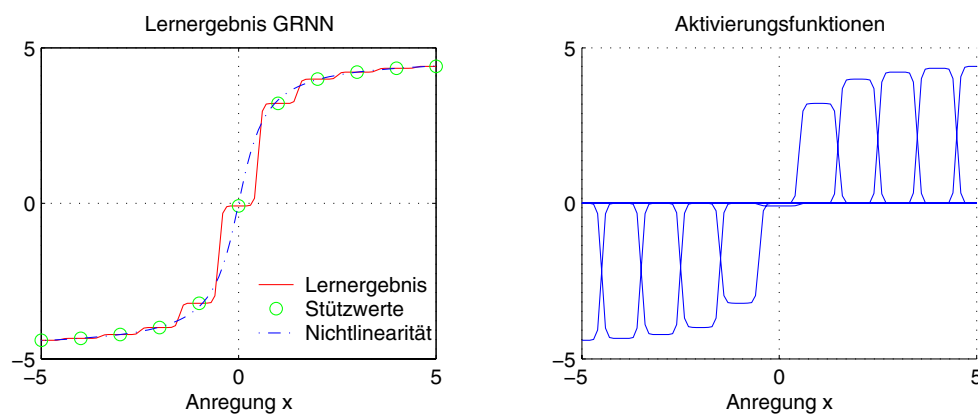


Bild 3.16: Gelernte Kennlinie (-) und wahre Kennlinie (.-) (links) und Aktivierungsfunktionen (rechts) für $\sigma_{norm} = 0.2$

Im Folgenden soll nun noch die Auswirkung einer zu geringen Anzahl von Stützstellen untersucht werden. In Bereichen, in denen sich die zu lernende Funktion nur wenig ändert, reichen auch wenige Stützwerte zur Approximation aus. In Bereichen steiler Übergänge (im vorliegenden Beispiel um den Wert $x = 0$) kann die Funktion nur mit verminderter Genauigkeit nachgebildet werden (Bild 3.17). Eine Steigerung der Approximationsgenauigkeit kann nur durch Erhöhung der Stützwertezahl erreicht werden; durch eine Verlängerung der Lerndauer kann keine Verbesserung erreicht werden. Bild 3.18 links zeigt die Konvergenz

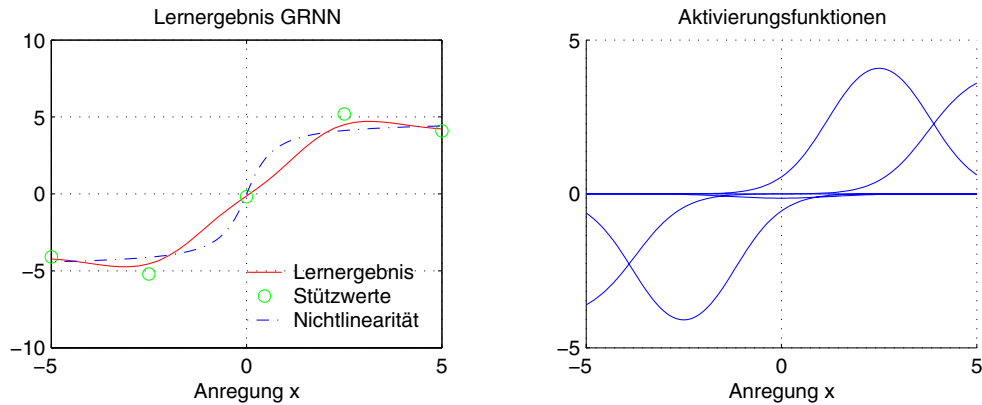


Bild 3.17: Gelernte Kennlinie (—) und wahre Kennlinie (---) (links) und Aktivierungsfunktionen (rechts) für $p = 5$ Stützwerte und $\sigma_{norm} = 0.5$

der fünf Stützwerte im Zeitbereich. Nach etwa zehn Sekunden ergibt sich keine nennenswerte Veränderung mehr. Die Interpolation zwischen zwei Stützstellen erfolgt hier durch

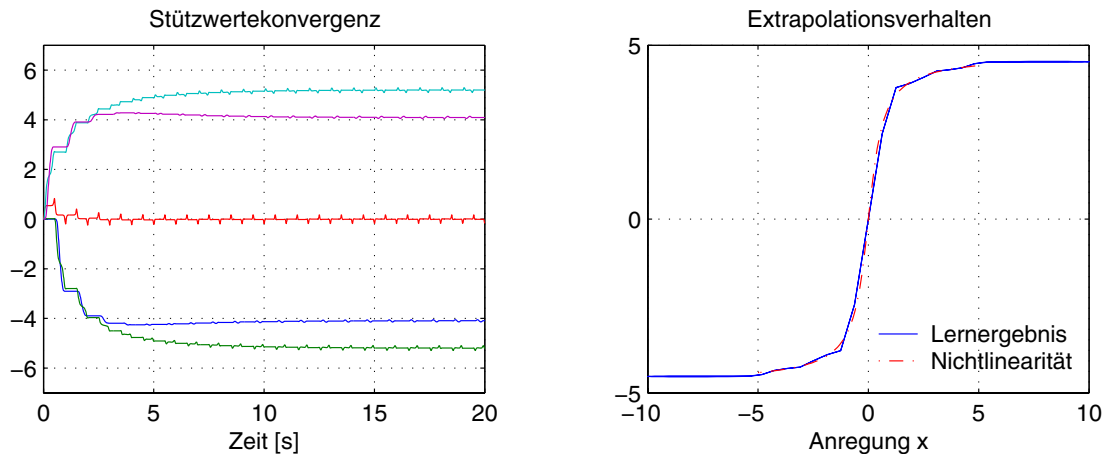


Bild 3.18: Stützwertekonvergenz und Extrapolationsverhalten

sanfte Überlagerung der beiden nächstgelegenen Stützwerte. Eine harte Umschaltung findet wegen ausreichend großem σ nicht statt.

Wenn das trainierte Netzwerk zur Reproduktion der Kennlinie betrieben wird, kann es vorkommen, dass Eingangswerte abgefragt werden, die außerhalb des vorgesehenen Eingangsbereichs liegen. Diese Extrapolationseigenschaften des GRNN werden an einem vollständig

trainierten Netzwerk bei abgeschaltetem Lernen veranschaulicht. Innerhalb des Eingangsbereichs (ist identisch mit dem Trainingsbereich) für $-5 \leq x \leq 5$ wird die Kennlinie exakt wiedergegeben. Außerhalb des Eingangsbereichs erfolgt eine konstante Extrapolation, d.h. der Wert des jeweils äußersten Stützwerts wird konstant gehalten. Für regelungstechnische Anwendungen ist besonders wichtig, dass kein Abfall der Kennlinie auf null erfolgt, da dies normalerweise nicht den physikalischen Tatsachen entspricht. Das Extrapolationsverhalten des GRNN zeigt Bild 3.18 rechts. Außerhalb des Eingangsbereichs (für $x < -5$ und $x > 5$) wird der Ausgangswert auf dem Wert des nächstliegenden Stützwerts gehalten.

3.3 Lernfähiger Beobachter

Im vorangegangenen Kap. 3.1.3 wurde eine Zustandsdarstellung für dynamische Systeme mit isolierten Nichtlinearitäten eingeführt. Diese Darstellung geht von bekanntem (oder vorab identifiziertem) linearem Streckenanteil und unbekannter Nichtlinearität aus. In Kap. 3.2 wurden Verfahren zur Approximation statischer nichtlinearer Funktionen mit Hilfe von neuronalen Netzen eingeführt. Nun wird die Identifikation statischer Nichtlinearitäten mit einer Systemidentifikation für die eingeführte Klasse von dynamischen Systemen kombiniert. Zu diesem Zweck wird ein lernfähiger Beobachter entworfen, der die folgenden zwei Aufgaben gleichzeitig erfüllt:

- Schätzung aller Systemzustände
- Identifikation der auftretenden statischen Nichtlinearität

Beide Ergebnisse haben für die spätere Einbeziehung in eine Regelungsstruktur eine wichtige Bedeutung. Sowohl die Schätzwerte der Systemzustände als auch die identifizierte Nichtlinearität können für das prädiktive Regelungsverfahren aus Kap. 6 genutzt werden. Durch das verwendete GRNN zur Nichtlinearitätsschätzung lässt sich Vorwissen über die Form der Nichtlinearität durch Vorbelegen einzelner Stützwerte leicht einbringen.

Zunächst werden alle Ausführungen für den SISO-Fall betrachtet, eine Erweiterung auf den MIMO-Fall erfolgt schließlich in Kap. 3.3.4.

3.3.1 Strecken mit isolierter Nichtlinearität

Die Eigenschaften und Voraussetzungen für die betrachtete Systemklasse müssen vorab klar definiert werden. Dazu sind zunächst einige Definitionen und Vereinbarungen notwendig.

Definition 3.6: Strecke mit isolierter Nichtlinearität

Als Strecke mit isolierter Nichtlinearität wird eine Strecke bezeichnet, die in Zustandsdarstellung dargestellt werden kann als

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}, u) \quad \text{und} \quad y = \mathbf{c}^T \mathbf{x} + du \quad (3.63)$$

Die isolierte Nichtlinearität ist also durch ein Produkt aus skalarer Nichtlinearität $\mathcal{NL}(\mathbf{x}, u)$ und Einkoppelvektor $\mathbf{k}$ darstellbar.

Alle hier betrachteten Nichtlinearitäten sind statische Nichtlinearitäten, d.h. sie hängen nur von Zustandsgrößen und Eingangssignalen, nicht aber explizit von der Zeit ab.

Der Signalflussplan eines dynamischen SISO-Systems mit isolierter Nichtlinearität ist in Bild 3.19 dargestellt. Darin sind alle blockorientierten Anordnungen wie Rückkopplungen oder Serienschaltungen enthalten.

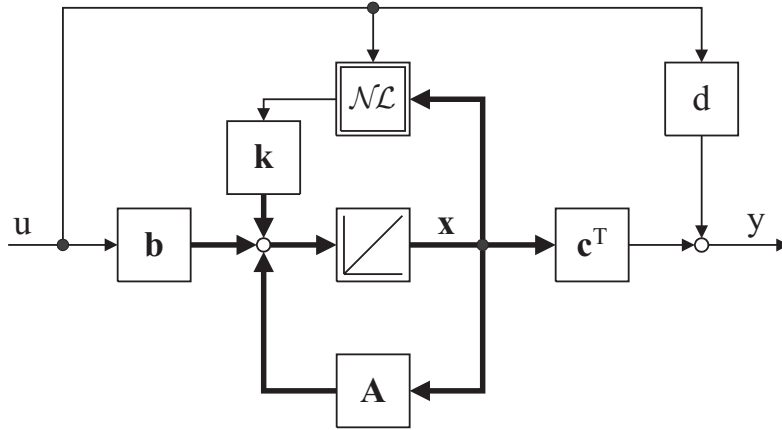


Bild 3.19: Signalflussplan eines dynamischen Systems mit isolierter Nichtlinearität

Für den Entwurf des lernfähigen Beobachters stellt sich nun die Aufgabe, die isolierte Nichtlinearität durch ein neuronales Netz zu approximieren. Dazu findet das in Kap. 3.2.5 eingeführte *General Regression Neural Network* Verwendung.

3.3.2 Beobachterentwurf bei messbarem Eingangsraum

Das neuronale Netz wird in einen Zustandsbeobachter eingebettet und hat die Aufgabe, die isolierte Nichtlinearität $\mathcal{N}(\mathbf{x}, u)$ nachzubilden. Diese Approximation erfolgt mittels eines Lerngesetzes, in das auch der Fehler zwischen Beobachterausgang und Streckenausgang, also $\hat{y} - y$, eingeht.

Die Nichtlinearität kann nur mit dem Streckenausgang identifiziert werden, wenn ihre Auswirkungen am Ausgang auch sichtbar werden.

Definition 3.7: Sichtbarkeit der Nichtlinearität

Die Sichtbarkeit einer Nichtlinearität am Ausgang einer SISO-Strecke ist genau dann gegeben, wenn die Übertragungsfunktion $H_{NL}(s)$

$$H_{NL}(s) = \mathbf{c}^T (s\mathbf{E} - \mathbf{A})^{-1} \mathbf{k} \quad (3.64)$$

vom Angriffspunkt der Nichtlinearität zum Streckenausgang ungleich null ist.

Der lineare Teil der Strecke muss bekannt sein. Es muss Kenntnis über die Systemmatrizen, den Angriffspunkt sowie die Eingangsgrößen der Nichtlinearität vorliegen. Zunächst wird

der Beobachterentwurf für eine Strecke mit isolierter Nichtlinearität bei messbaren oder rückrechenbaren (z.B. durch Differentiation) Eingangsgrößen betrachtet. Das bedeutet, alle Eingänge von $\mathcal{NL}(x_1 \dots x_k, u)$ sind messbar bzw. bestimmbar. Da die Nichtlinearität nicht zwangsläufig von allen Zuständen abhängt, werden die tatsächlich benötigten Signale im Eingangsraum $\mathbf{x}_E$ der Nichtlinearität zusammengefasst.

Beobachteransatz

Die Prozessgleichung für ein SISO-System mit isolierter Nichtlinearität lautet

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}_E, u) \quad \text{und} \quad y = \mathbf{c}^T \mathbf{x} + du \quad (3.65)$$

mit bekannten und konstanten Systemmatrizen $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}^T$, d und $\mathbf{k}$. Der Beobachter wird in Analogie zum Luenbergerbeobachter [15, 74, 75] angesetzt.

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k} \cdot \widehat{\mathcal{NL}}(\mathbf{x}_E, u) - \mathbf{l} \cdot e \\ &= (\mathbf{A} - \mathbf{l}\mathbf{c}^T)\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k} \cdot \widehat{\mathcal{NL}}(\mathbf{x}_E, u) + \mathbf{l}\mathbf{c}^T \mathbf{x} \\ \hat{y} &= \mathbf{c}^T \hat{\mathbf{x}} + du \end{aligned} \quad (3.66)$$

wobei der Beobachterfehler gemäß

$$e = \hat{y} - y \quad (3.67)$$

definiert wird. Bei den folgenden Betrachtungen wird nun die isolierte Nichtlinearität als Eingang einer ansonsten linearen Strecke betrachtet. Nur so können alle linearen Methoden wie Superpositionsprinzip und Darstellung als Übertragungsfunktion angewendet werden. Der Schätzwert $\widehat{\mathcal{NL}}$ der Nichtlinearität wird als neuronales Netz (GRNN) implementiert. In Bild 3.20 ist Strecke und zugehöriger Beobachter dargestellt. Für die Dimensionierung des Rückführvektors $\mathbf{l}$ gelten alle bekannten Einstellvorschriften für den linearen Beobachterentwurf (Polvorgabe, LQ-Optimierung, Kalman-Filter [72]) zur Erzielung eines asymptotisch stabilen und ausreichend schnellen Beobachter-Eigenverhaltens, so dass hierauf nicht weiter eingegangen wird.

Zur Bestimmung eines Lerngesetzes für die geschätzte Nichtlinearität $\widehat{\mathcal{NL}}$ ist der dynamische Zusammenhang zwischen Parameterfehler der Nichtlinearität und Beobachterfehler e notwendig. Dazu wird die Fehlerdifferentialgleichung des Zustandsfehlers $\mathbf{e}_Z = \hat{\mathbf{x}} - \mathbf{x}$ hergeleitet. Durch Differenzbildung von Gleichung (3.66) und (3.65) erhält man:

$$\dot{\mathbf{e}}_Z = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) \mathbf{e}_Z + \mathbf{k} \cdot (\widehat{\mathcal{NL}}(\mathbf{x}_E, u) - \mathcal{NL}(\mathbf{x}_E, u)) \quad (3.68)$$

Zur einfacheren Darstellung wird nun auf eine hybride Notation mit Größen im Laplace- und Zeitbereich übergegangen. Diese Schreibweise ist in der Literatur verbreitet [17] und dient der kompakten Schreibweise. Die Anfangswerte der Laplace-Transformation werden vernachlässigt.

$$s \cdot \mathbf{e}_Z = (\mathbf{A} - \mathbf{l}\mathbf{c}^T) \mathbf{e}_Z + \mathbf{k} \cdot (\widehat{\mathcal{NL}} - \mathcal{NL}) \quad (3.69)$$

$$\mathbf{e}_Z = (s\mathbf{E} - \mathbf{A} + \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{k} \cdot (\widehat{\mathcal{NL}} - \mathcal{NL}) \quad (3.70)$$

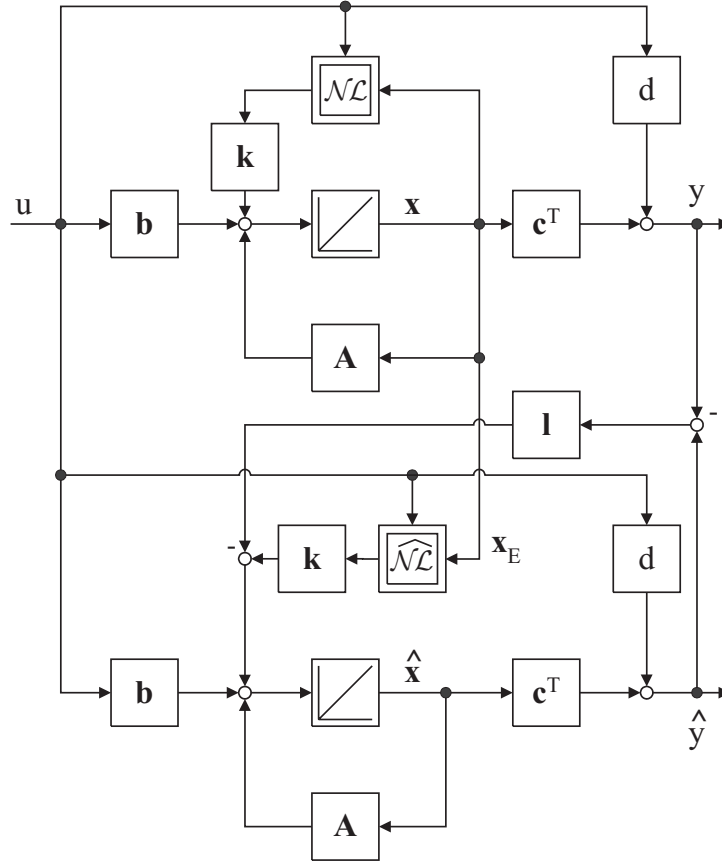


Bild 3.20: Lernfähiger Beobachter für eine SISO-Strecke mit isolierter Nichtlinearität

Der Ausgangsfehler ergibt sich aus dem Zusammenhang $e = \mathbf{c}^T \mathbf{e}_Z$ zu²⁾

$$e = \underbrace{\mathbf{c}^T (s\mathbf{E} - \mathbf{A} + \mathbf{l} \mathbf{c}^T)^{-1} \mathbf{k}}_{H(s)} \cdot (\widehat{\mathcal{NL}} - \mathcal{NL}) = H(s) \cdot (\widehat{\mathcal{NL}} - \mathcal{NL}) \quad (3.71)$$

Die lineare Fehlerübertragungsfunktion $H(s)$ beschreibt das dynamische Verhalten zwischen Schätzfehler der Nichtlinearität $(\widehat{\mathcal{NL}} - \mathcal{NL})$ und Ausgangsfehler $e = \hat{y} - y$. Diese Dynamik ist bei gegebener Beobachtbarkeit des linearen Streckenteils (siehe Definition 3.8) durch Wahl des Rückführvektors $\mathbf{l}$ weitgehend frei wählbar [15]. Insbesondere kann $H(s)$ unter dieser Voraussetzung stets asymptotisch stabil gewählt werden.

Nun wird für die zu schätzende Nichtlinearität $\widehat{\mathcal{NL}}$ ein GRNN nach Gleichung (3.17) eingesetzt. Für die tatsächliche Nichtlinearität wird ein optimal trainiertes GRNN mit konstanten Stützwerten angesetzt.

$$\widehat{\mathcal{NL}}(\mathbf{x}_E, u) = \hat{\Theta}^T \mathcal{A}(\mathbf{x}_E, u) \quad \mathcal{NL}(\mathbf{x}_E, u) = \Theta^T \mathcal{A}(\mathbf{x}_E, u) \quad (3.72)$$

Mit der Definition des Parameterfehlervektors $\Phi = \hat{\Theta} - \Theta$ ergibt sich aus Gleichung (3.71)

²⁾ Die exakte Schreibweise ist: $e(s) = H(s) \cdot \mathcal{L} \{ \widehat{\mathcal{NL}} - \mathcal{NL} \}$, wobei $\mathcal{L}$ die Laplace-Transformation bezeichnet. Das $\mathcal{L}$ -Zeichen wird der Einfachheit halber jedoch in den weiteren Ausführungen weggelassen.

schließlich:

$$e = H(s)\Phi^T \mathcal{A}(\mathbf{x}_E, u) \quad (3.73)$$

Gleichung (3.73) stellt genau Fehlermodell 3 (siehe Bild 3.12) bzw. Fehlermodell 4 (siehe Bild 3.13) dar. Erfüllt $H(s)$ die SPR-Bedingung aus Definition 3.5, so kann das vereinfachte Lerngesetz aus Gleichung (3.22) angewendet werden.

$$\frac{d}{dt} \hat{\Theta} = \dot{\hat{\Theta}} = -\eta e \mathcal{A} \quad (3.74)$$

Bei allgemeiner Fehlerübertragungsfunktion $H(s)$ ohne besondere Eigenschaften, wird die Adaptionsgleichung aus Fehlermodell 4 (Gleichung (3.58)) verwendet.

$$\frac{d}{dt} \hat{\Theta} = \dot{\hat{\Theta}} = -\eta \epsilon H(s) \mathbf{E} \mathcal{A} \quad (3.75)$$

Dafür wird der in Bild 3.13 dargestellte Fehler ϵ zur Parameteradaption herangezogen.

$$\epsilon = \hat{\Theta}^T H(s) \mathbf{E} \mathcal{A} - H(s) \Theta^T \mathcal{A} = \Phi^T H(s) \mathbf{E} \mathcal{A} \quad (3.76)$$

Alle Aussagen zu Stabilität und Parameterkonvergenz aus Kap. 3.2.6.4 bleiben uneingeschränkt gültig, so dass eine garantiert stabile Identifikation der unbekannten Nichtlinearität gewährleistet ist.

Wenn nun der Parameterfehler des GRNN nach null strebt, so folgt aus Gleichung (3.71), dass auch der Ausgangsfehler des Beobachters nach null streben muss.

$$\widehat{\mathcal{NL}} \rightarrow \mathcal{NL} \quad \Rightarrow \quad \Phi \rightarrow 0 \quad \Rightarrow \quad e \rightarrow 0 \quad (3.77)$$

Wenn die geschätzte Nichtlinearität gegen die wahre strebt, so folgt aus Gleichung (3.68) jedoch auch, dass der Zustandsfehler $\mathbf{e}_Z$ nach null strebt, wenn die Eigenwerte der Matrix $\mathbf{A} - \mathbf{l}c^T$ negativen Realteil besitzen. Das bedeutet, die geschätzten Zustandsgrößen $\hat{\mathbf{x}}$ streben gegen die wahren Zustände $\mathbf{x}$.

$$\widehat{\mathcal{NL}} \rightarrow \mathcal{NL} \quad \Rightarrow \quad \Phi \rightarrow 0 \quad \Rightarrow \quad \hat{\mathbf{x}} \rightarrow \mathbf{x} \quad (3.78)$$

Damit sind beide Funktionen des lernfähigen Beobachters gezeigt. Es erfolgt sowohl eine Schätzung der Nichtlinearität, als auch eine Schätzung aller Systemzustände.

3.3.3 Beobachterentwurf bei nicht messbarem Eingangsraum

Im Gegensatz zu Kap. 3.3.2 ist nun die Messbarkeit des gesamten Eingangsraumes der Nichtlinearität $\mathcal{NL}(\mathbf{x}_E, u)$ nicht mehr gefordert, vielmehr genügt das Ein- und Ausgangssignal der Strecke.

Zusätzliche Voraussetzung

Da jetzt eine Messung des Eingangsraumes der Nichtlinearität nicht mehr vorliegt, muss für den linearen Teil der Strecke zusätzlich zu den Voraussetzungen in Kap. 3.3.1 noch die vollständige Zustandsbeobachtbarkeit gefordert werden.

Definition 3.8: Beobachtbarkeit im linearen Teil

Eine SISO Strecke der Ordnung N mit isolierter Nichtlinearität

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}, u) \quad \text{und} \quad y = \mathbf{c}^T \mathbf{x} + du \quad (3.79)$$

ist im linearen Teil genau dann vollständig zustandsbeobachtbar, wenn die Beobachtbarkeitsmatrix $\mathbf{Q}_{obs}$.

$$\mathbf{Q}_{obs} = [\mathbf{c} \quad \mathbf{A}^T \mathbf{c} \quad \dots \quad (\mathbf{A}^T)^{(N-1)} \mathbf{c}] \quad (3.80)$$

regulär ist, wenn also gilt

$$\det(\mathbf{Q}_{obs}) \neq 0 \quad (3.81)$$

Die Forderung nach Beobachtbarkeit im linearen Teil ist nötig, da die Eingangssignale $\mathbf{x}_E$ der Nichtlinearität aus dem Streckenausgangssignal y rekonstruiert werden müssen. Dies kann nur bei Beobachtbarkeit des linearen Teils der Strecke gelingen. Es ist zu beachten, dass es sich hierbei um eine notwendige, aber nicht hinreichende Voraussetzung handelt, da zur Zustandsrekonstruktion auch die Nichtlinearität eine Rolle spielt.

Beobachteransatz

Der Beobachter wird in Analogie zu einem Luenberger-Beobachter angesetzt [15]. Die zu identifizierende Nichtlinearität besitzt als Eingangssignale den realen Eingangsraum $\mathbf{x}_E$. Dieser kann jedoch wegen der fehlenden Messung nicht bestimmt werden; deshalb wird im Beobachter die Nichtlinearität mit beobachtetem $\hat{\mathbf{x}}_E$ als Eingangssignal angesetzt. Die Streckenbeschreibung ist durch Gleichung (3.65) gegeben. Die zugehörige Beobachtergleichung lautet:

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k} \cdot \widehat{\mathcal{NL}}(\hat{\mathbf{x}}_E, u) - \mathbf{l} \cdot e \\ &= (\mathbf{A} - \mathbf{l}\mathbf{c}^T)\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k} \cdot \widehat{\mathcal{NL}}(\hat{\mathbf{x}}_E, u) + \mathbf{l}\mathbf{c}^T \mathbf{x} \\ \hat{y} &= \mathbf{c}^T \hat{\mathbf{x}} + du \end{aligned} \quad (3.82)$$

Der Rückführvektor $\mathbf{l}$ wird dazu benutzt, ein geeignetes Eigenverhalten des Beobachters einzustellen. In Bild 3.21 ist der Signalflussplan von Strecke und Beobachter dargestellt. Im Unterschied zu Bild 3.20 ist das Eingangssignal $\hat{\mathbf{x}}_E$ von $\widehat{\mathcal{NL}}$ nun eine Untermenge des geschätzten Zustandsvektors $\hat{\mathbf{x}}$ (siehe Markierung in Bild 3.21). Die geschätzte und reale Nichtlinearität werden jetzt durch die Approximationsgleichungen des GRNN dargestellt und die Fehlerdynamik zwischen Parameterfehler Φ des GRNN und Ausgangsfehler e des Beobachters berechnet. Mit der Fehlerübertragungsfunktion $H(s) = \mathbf{c}^T (s\mathbf{E} - \mathbf{A} + \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{k}$ ergibt sich:

$$e = H(s) \cdot \left(\widehat{\mathcal{NL}}(\hat{\mathbf{x}}_E, u) - \mathcal{NL}(\mathbf{x}_E, u) \right) = H(s) \left(\hat{\Theta}^T \mathcal{A}(\hat{\mathbf{x}}_E, u) - \Theta^T \mathcal{A}(\mathbf{x}_E, u) \right) \quad (3.83)$$

Um die Fehlergleichung mit einem Parameterfehlervektor Φ darstellen zu können und somit die aus Kap. 3.2.6.4 bekannten Lerngesetze anwenden zu können, werden virtuelle Stützwerte eingeführt. In Gleichung (3.83) ergibt sich das Problem, dass $\mathcal{A} = \mathcal{A}(\mathbf{x}_E)$

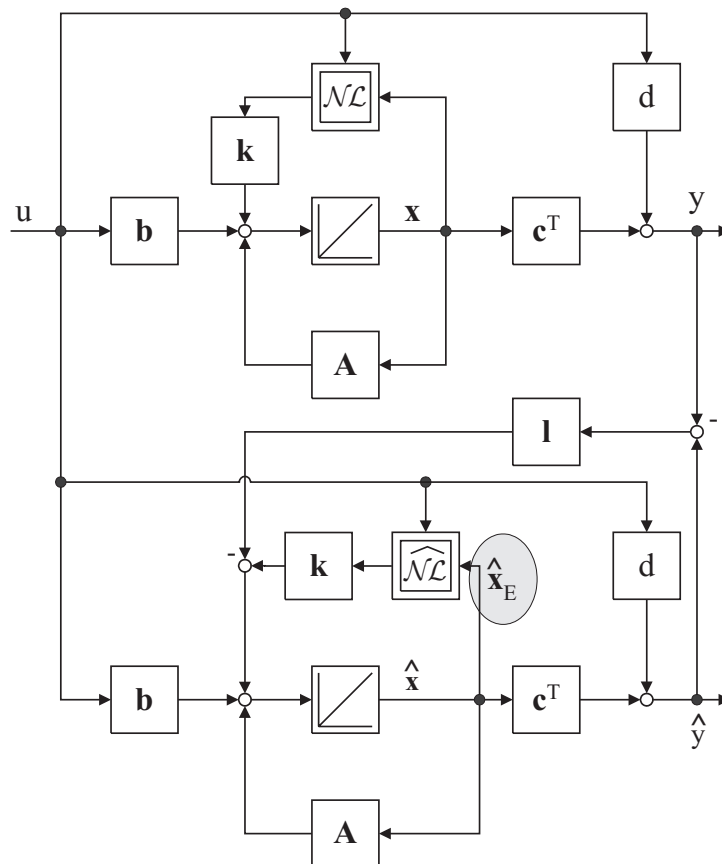


Bild 3.21: Lernfähiger Beobachter bei nicht messbarem Eingangsraum

nicht zur Verfügung steht, sondern nur der Aktivierungsvektor im Beobachterzustandsraum $\hat{\mathcal{A}} = \mathcal{A}(\hat{\mathbf{x}}_E)$. Während des Einschwingvorgangs des Beobachters und des Lernvorgangs des GRNN gilt im Allgemeinen:

$$\mathcal{A}(\mathbf{x}_E, u) \neq \mathcal{A}(\hat{\mathbf{x}}_E, u) \quad \Rightarrow \quad \mathcal{A} \neq \hat{\mathcal{A}} \quad (3.84)$$

Man kann jedoch umgekehrt die reale Nichtlinearität durch den Aktivierungsvektor $\hat{\mathcal{A}}$ darstellen. Dazu setzt man an:

$$\mathcal{NL} = \Theta^T \mathcal{A}(\mathbf{x}_E, u) = \Theta^T \mathcal{A} = \check{\Theta}^T \mathcal{A}(\hat{\mathbf{x}}_E, u) = \check{\Theta}^T \hat{\mathcal{A}} \quad (3.85)$$

In Gleichung (3.85) stellen $\check{\Theta}$ zeitvariante virtuelle Stützwerte dar, die die Bewegung der Nichtlinearität im Beobachterzustandsraum $\hat{\mathbf{x}}$ beschreiben. Sie variieren, wenn sich $\mathbf{x}$ und $\hat{\mathbf{x}}$ und somit auch die Aktivierungen $\mathcal{A}$ und $\hat{\mathcal{A}}$ ändern. Wegen der Lokalität der Stützwertewirkung und der eindeutigen Abbildung der Nichtlinearität durch das GRNN bei ausreichender Anregung, gilt nach [46] jedoch, dass sich die virtuellen Stützwerte genau den realen angleichen, wenn sich die Beobachterzustände $\hat{\mathbf{x}}$ den Streckenzuständen $\mathbf{x}$ angleichen.

$$\hat{\mathbf{x}} \rightarrow \mathbf{x} \quad \Rightarrow \quad \hat{\mathcal{A}} \rightarrow \mathcal{A} \quad \Rightarrow \quad \check{\Theta} \rightarrow \Theta \quad (3.86)$$

Es gilt nun zu zeigen, wann Gleichung (3.86) tatsächlich erfüllt ist. Mit Gleichung (3.83) kann man mit dem Parameterfehlervektor $\Phi = \hat{\Theta} - \check{\Theta}$ ansetzen:

$$e = H(s) \left(\hat{\Theta}^T \hat{\mathcal{A}} - \Theta^T \mathcal{A} \right) = H(s) \left(\hat{\Theta}^T - \check{\Theta}^T \right) \hat{\mathcal{A}} = H(s) \Phi^T \hat{\mathcal{A}} \quad (3.87)$$

Für diese Fehlergleichung kann wieder die aus Kap. 3.2.6.4 bekannte stabile Adaptionsgleichung für Fehlermodell 4 verwendet werden.

$$\dot{\Phi} = -\eta \epsilon H(s) \mathbf{E} \hat{\mathcal{A}} \quad (3.88)$$

Als Adaptionfehler ergibt sich in Analogie zu Gleichung (3.57):

$$\epsilon = \Phi^T H(s) \mathbf{E} \hat{\mathcal{A}} \quad (3.89)$$

Infolge der Zeitvarianz der virtuellen Stützwerte kann man nun aber **nicht** folgern, dass $\dot{\Phi} = \dot{\check{\Theta}}$ ist. Die interessierende Größe ist aber nach wie vor $\hat{\Theta}$. Im Folgenden wird daher das gleiche Adaptionsgesetz wie in Gleichung (3.75) verwendet.

$$\dot{\check{\Theta}} = -\eta_{virt} \epsilon H(s) \mathbf{E} \hat{\mathcal{A}} \quad (3.90)$$

Dies bedeutet, für die Änderung des Parameterfehlervektors:

$$\dot{\Phi} = \dot{\hat{\Theta}} - \dot{\check{\Theta}} = -\eta_{virt} \epsilon H(s) \hat{\mathcal{A}} - \dot{\check{\Theta}} = - \left(\eta_{virt} \epsilon H(s) \hat{\mathcal{A}} + \dot{\check{\Theta}} \right) \quad (3.91)$$

Aus Gleichung (3.91) und (3.88) ist ersichtlich, dass ein stabiler Lernvorgang genau dann gewährleistet ist, wenn sich das Vorzeichen von $\dot{\Phi}$ trotz $\dot{\check{\Theta}}$ nicht ändert, d.h. wenn der Term $\dot{\check{\Theta}}$ gegenüber $\dot{\check{\Theta}}$ überwiegt. Da die Lernschrittweite η_{virt} positiv sein muss, ergibt sich eine Bedingung der Art:

$$0 < \eta_{min} < \eta_{virt} \quad (3.92)$$

Die Untergrenze η_{min} wird durch $\dot{\check{\Theta}}$ bestimmt. Sie ist von den Beobachterpolen und somit vom Rückführvektor $\mathbf{l}$ abhängig. Je weiter die Beobachterpole in der linken komplexen s-Ebene liegen, desto näher liegen $\mathbf{x}$ und $\hat{\mathbf{x}}$ trotz ungenügend trainierter Nichtlinearität beisammen und desto kleiner ist $\dot{\check{\Theta}}$. Es kann nur auf die Existenz einer solchen Untergrenze geschlossen werden, eine genaue Angabe über deren Wert kann nicht gemacht werden. Sicherlich spielt bei der Untergrenze auch die Form der Nichtlinearität eine Rolle.

Mit diesem Verfahren kann auch bei nicht messbaren Zustandsgrößen die isolierte Nichtlinearität identifiziert und alle Zustandsgrößen geschätzt werden. Es steht damit sowohl die Funktionalität der Schätzung der Nichtlinearität als auch der Zustandsbeobachtung zur Verfügung.

3.3.4 Identifikation mehrerer Nichtlinearitäten

Da komplexe Systeme häufig mehr als nur eine Nichtlinearität enthalten, ist eine Erweiterung auf mehrere isolierte Nichtlinearitäten erforderlich. Daher werden jetzt nichtlineare

Systeme mit mehreren Ein- und Ausgängen (MIMO) mit folgender Systembeschreibung betrachtet:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \underline{\mathcal{N}}\mathcal{L}(\mathbf{x}_E) \quad (3.93)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \quad (3.94)$$

$$\underline{\mathcal{N}}\mathcal{L}(\mathbf{x}_E) = \sum_{i=1}^k \mathbf{k}_i \cdot \mathcal{N}\mathcal{L}_i \quad (3.95)$$

Die Unterstreichung von $\underline{\mathcal{N}}\mathcal{L}$ kennzeichnet eine vektorwertige Nichtlinearität mit k unbekannten Komponenten. Deren Anzahl darf die Ausgangsdimension von $\mathbf{y}$ nicht übersteigen, d.h.

$$k \leq \text{Rang}(\mathbf{C}) \quad (3.96)$$

Der Beobachter wird, analog zu den Überlegungen im SISO Fall, angesetzt zu

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \sum_{i=1}^k \mathbf{k}_i \widehat{\mathcal{N}}\mathcal{L}_i(\mathbf{x}_E) - \mathbf{L}(\hat{\mathbf{y}} - \mathbf{y}) \quad (3.97)$$

$$\hat{\mathbf{y}} = \mathbf{C}\hat{\mathbf{x}}$$

Stabilität für den linearen Beobachteranteil wird wiederum vorausgesetzt. Durch Differenzbildung von Gleichung (3.97) und (3.93) können die einzelnen Komponenten des Beobachterfehlers $\mathbf{e} = \hat{\mathbf{y}} - \mathbf{y}$ berechnet werden:

$$e_i = \hat{y}_i - y_i = \mathbf{c}_i (s\mathbf{E} - \mathbf{A} + \mathbf{L}\mathbf{C})^{-1} \sum_{i=1}^k \mathbf{k}_i \left(\widehat{\mathcal{N}}\mathcal{L}_i - \mathcal{N}\mathcal{L}_i \right) \quad (3.98)$$

$\mathbf{c}_i$ steht für den i -ten Zeilenvektor der Auskoppelmatrix $\mathbf{C}$. Die unbekannten Komponenten von $\underline{\mathcal{N}}\mathcal{L}(\mathbf{x}_E)$ werden zum Vektor $\underline{\mathcal{N}}\mathcal{L}(\mathbf{x}_E)^k$ der Länge k zusammengefasst. Durch die Wahl ebensovieler Beobachterfehler $\mathbf{e}^k$ erhält man letztlich ein System von k Fehlerdifferentialgleichungen der Form

$$\mathbf{e}^k = \mathbf{H}_F \left(\widehat{\underline{\mathcal{N}}\mathcal{L}}(\mathbf{x}_E)^k - \underline{\mathcal{N}}\mathcal{L}(\mathbf{x}_E)^k \right) \quad (3.99)$$

mit der Fehlerübertragungsmatrix $\mathbf{H}_F$, deren Elemente sich aus Gleichung (3.98) ergeben. Die Fehlerübertragungsmatrix beschreibt den dynamischen Einfluss der Schätzfehler der einzelnen Nichtlinearitäten auf den jeweiligen Beobachterfehler. Besitzt sie reine Diagonalgestalt, so liegt ein fehlerentkoppeltes System vor, was letztlich k Einzelsystemen mit je einer Nichtlinearität entspricht. In den überwiegenden Fällen wird jedoch kein entkoppeltes System vorliegen. Der bisherige Ansatz zur Identifikation der Nichtlinearitäten führt daher nicht zum Erfolg.

Es wird der folgende Weg gewählt: Das verkoppelte System wird durch eine Transformation in k entkoppelte Einzelsysteme überführt. Die hierfür entworfene *Fehlertransformation* zeichnet sich durch ihren Aufbau aus Integratoren aus, was sich besonders in der praktischen Anwendung als großer Vorteil gegenüber Transformationen mit differenzierenden Anteilen erweist.

Durch eine Linkstransformation des Fehlervektors mit der inversen Fehlermatrix $\mathbf{H}_F^{-1}$ erhält man den transformierten Fehlervektor $\mathbf{e}_T$.

$$\mathbf{e}_T = \mathbf{H}_F^{-1} \mathbf{e}^k = \mathbf{H}_F^{-1} \mathbf{H}_F \left(\widehat{\mathcal{N}}(\mathbf{x}_E)^k - \mathcal{N}(\mathbf{x}_E)^k \right) \quad (3.100)$$

Die Matrix $\mathbf{E} = \mathbf{H}_F^{-1} \mathbf{H}_F$ besitzt selbstredend reine Diagonalgestalt. Damit ist gewährleistet, dass im Fehlervektor $\mathbf{e}_T$ die Fehler entkoppelt vorliegen. Die einzelnen Komponenten der inversen Fehlerübertragungsmatrix $\mathbf{H}_F^{-1}$ werden in vielen Fällen nur durch differenzierende Elemente darstellbar sein. Dies stellt jedoch erhebliche Probleme in Bezug auf die praktische Realisierung an einer realen Strecke dar (Störeinflüsse durch Messrauschen). Die Matrix $\mathbf{H}_F^{-1}$ stellt eine Filterung der k Fehlersignale dar. Durch Nachschalten eines weiteren Filternetzwerkes $\mathbf{H}_W$ in Diagonalgestalt

$$\mathbf{H}_W = \begin{bmatrix} H_{11} & 0 & \dots & 0 \\ 0 & H_{22} & \dots & 0 \\ 0 & \vdots & \ddots & 0 \\ 0 & \dots & 0 & H_{kk} \end{bmatrix} \quad (3.101)$$

ergibt sich für das gesamte Entkopplungsfilter der Zusammenhang

$$\mathbf{H}_H = \mathbf{H}_W \mathbf{H}_F^{-1} \quad (3.102)$$

Die k Elemente der Matrix $\mathbf{H}_W$ wählt man geschickterweise so, dass sich für die einzelnen Elemente des gesamten Entkopplungsnetzwerkes $\mathbf{H}_H$ Polynome von relativem Grad größer gleich null ergeben. Dadurch ist eine Realisierung des Entkopplungsnetzwerkes mit Integratoren gewährleistet. Wird nun das Entkopplungsfilter in Gleichung (3.99) eingesetzt, so ergibt sich diese zu

$$\mathbf{e}_H = \mathbf{H}_H \mathbf{e}^k = \mathbf{H}_W \mathbf{H}_F^{-1} \mathbf{H}_F \left(\widehat{\mathcal{N}}^k - \mathcal{N}^k \right) = \mathbf{H}_W \left(\widehat{\mathcal{N}}^k - \mathcal{N}^k \right) \quad (3.103)$$

Die i -te Komponente des transformierten (gefilterten) Fehlervektors $\mathbf{e}_H$ korreliert nun über die gewählten Übertragungsfunktionen H_{ii} mit dem i -ten Approximationsfehler von $\widehat{\mathcal{N}}^k - \mathcal{N}^k$. Dadurch wurde das verkoppelte System in k entkoppelte Systeme überführt, die sich durch Anwendung der aus Kap. 3.2.6.4 bekannten Fehlermodelle darstellen lassen. Die Adaptionsgesetze aus Kap. 3.3.2 und 3.3.3 sind demnach wieder anwendbar.

Das Entkopplungsnetzwerk $\mathbf{H}_H$ kann durch relativ freie Wahl der Matrix $\mathbf{H}_W$ entscheidend in seiner Dynamik beeinflusst werden. Es lassen sich die Frequenzgänge der einzelnen Übertragungsglieder der Matrix $\mathbf{H}_H = \mathbf{H}_W \mathbf{H}_F^{-1}$ in weiten Bereichen einstellen, was sich auf die Lerndynamik auswirkt. Da die Übertragungsglieder H_{ii} im Lerngesetz in der verzögerten Aktivierung auftauchen, bestimmen sie maßgeblich die Lerngeschwindigkeit des eingesetzten GRNN.

Die entkoppelten Fehlersignale $e_{H,i}$ ergeben sich zu

$$e_{H,i} = H_{ii} \left(\widehat{\mathcal{N}}_i^k - \mathcal{N}_i^k \right) \quad (3.104)$$

Die i -ten Komponenten der Vektoren $\underline{\mathcal{N}}^k$ und $\widehat{\underline{\mathcal{N}}}^k$ werden durch die Approximationsgleichungen des GRNN ersetzt.

$$\mathcal{N}_i^k = \Theta_i^T \cdot \mathcal{A}_i(\mathbf{x}_E) \quad \widehat{\mathcal{N}}_i^k = \hat{\Theta}_i^T \cdot \mathcal{A}_i(\mathbf{x}_E) \quad (3.105)$$

Durch die bereits bekannte Bildung des erweiterten Fehlers

$$\epsilon_i = \left(\hat{\Theta}_i^T - \Theta_i^T \right) H_{ii} \mathbf{E} \mathcal{A}_i \quad (3.106)$$

gelingt es, ein nach Lyapunov stabiles Lerngesetz zu formulieren (siehe Kap. 3.3.2).

$$\dot{\hat{\Theta}}_i = -\eta_i \epsilon_i H_{ii} \mathbf{E} \mathcal{A}_i \quad (3.107)$$

Durch Fehlerentkopplung ist es nun gelungen k isolierte Nichtlinearitäten mit k Messsignalen separierbar zu identifizieren. Für diesen Identifikationsansatz ist zwingend notwendig, dass die Anzahl der Nichtlinearitäten die Anzahl der unabhängigen Messgrößen nicht übersteigt. Wenn dies nicht der Fall ist, ergibt sich für die Matrix $\mathbf{H}_F$ in Gleichung (3.99) keine quadratische Matrix und damit ist auch keine Diagonalisierung nach Gleichung (3.100) möglich.

3.4 Experimentelle Validierung

Die theoretischen Ableitungen der Kap. 3.2 und 3.3 zur Identifikation isolierter Nichtlinearitäten werden nun an einem Prüfstand praktisch validiert. Dafür wird der in Anhang A ausführlich beschriebene Versuchsaufbau verwendet.

Viele Subsysteme mechatronischer Antriebsanordnungen bestehen aus elastisch gekoppelten rotierenden Massen. Die einfachste Anordnung stellt das Zweimassensystem (ZMS) dar. In industriellen Anlagen taucht es etwa bei gekoppelten Roboter-Gelenken, bei der Ankopplung von Motor und Walze einer Druckmaschine, bei einer elektrisch angetriebenen Drosselklappe im Luftpfad eines Verbrennungsmotors oder bei großen Radioteleskopen zwischen Antrieb und Reflektor (siehe Kap. 9.2) auf. Der verwendete Versuchsaufbau besitzt also auch praktische Relevanz.

Als Nichtlinearität wird ein drehzahlabhängiges Gegenmoment gewählt. Dies stellt einen sehr starken Reibungseinfluss dar. Das selbe System wird auch für die prädiktive Regelung in Kap. 8 verwendet. Die Gleichung der Nichtlinearität lautet:

$$M_2 = 6.4[Nm] \cdot \arctan(10[s/rad] \cdot \Omega_2) \quad (3.108)$$

Die Einheiten werden im Folgenden der Einfachheit halber weggelassen. Die Dynamik des Momentenregelkreises ist im Vergleich zu den Zeitkonstanten des mechanischen Teils sehr schnell, so dass sie nicht berücksichtigt werden muss. Die Zustandsdarstellung des Prozesses ist gegeben durch:

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} -\frac{d}{J_1} & -\frac{c}{J_1} & \frac{d}{J_1} \\ 1 & 0 & -1 \\ \frac{d}{J_2} & \frac{c}{J_2} & -\frac{d}{J_2} \end{bmatrix}}_{\mathbf{A}} \mathbf{x} + \underbrace{\begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{b}} u + \underbrace{\begin{bmatrix} 0 \\ 0 \\ -\frac{1}{J_2} \end{bmatrix}}_{\mathbf{k}} \cdot \underbrace{6.4 \arctan(10 \cdot \Omega_2)}_{\mathcal{N}(x_3)}$$

$$y = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{\mathbf{c}^T} \cdot \mathbf{x} = \Omega_2 \quad (3.109)$$

Der Zustandsvektor besteht aus der Motordrehzahl Ω_1 , der Lastdrehzahl Ω_2 und der Winkelverdrehung $\Delta\phi$.

$$\mathbf{x} = [\Omega_1 \quad \Delta\phi \quad \Omega_2]^T \quad (3.110)$$

Die Stellgröße $u = M_1$ ist das Antriebsmoment von Motor 1, die Nichtlinearität wird über Motor 2 in das System eingeprägt ($M_2 = \mathcal{NL}$). Am Signalfussplan des Gesamtsystems in Bild A.5 wird der Eingriffspunkt der Nichtlinearität besonders deutlich. Die Parameterwerte wurden in Anhang A durch lineare Identifikation vorab ermittelt. Es soll noch hervorgehoben werden, dass durch die Inkrementalgeber an beiden Maschinen vollständige Zustandsmessbarkeit vorliegt. Für den Beobachterentwurf und die Identifikation der Nichtlinearität wird jedoch nur die Drehzahl Ω_2 verwendet.

Wegen der Messbarkeit des Eingangssignals der Nichtlinearität kann der Beobachterentwurf gemäß Kap. 3.3.2 erfolgen. Die Fehlerübertragungsfunktion ergibt sich zu

$$H(s) = \mathbf{c}^T (s\mathbf{E} - \mathbf{A} + \mathbf{l}\mathbf{c}^T)^{-1} \mathbf{k} \quad (3.111)$$

Gleichung (3.111) erfüllt nicht die SPR-Bedingung, so dass zum Lernen Fehlermodell 4 implementiert werden muss. Der Beobachter-Rückführvektor $\mathbf{l}$ wurde durch Polvorgabe festgelegt. Dazu wurde das Dämpfungsoptimum mit einer Ersatzzeit von $T_{ers} = 50 \text{ ms}$ gewählt [29]. Das System wird zur Identifikation geregelt betrieben. Dazu wurde ein Zustandsregler mit Integralanteil ebenfalls nach dem Dämpfungsoptimum mit einer Ersatzzeit von 50 ms entworfen. Dieser berücksichtigt die Nichtlinearität nicht und hat daher auch kein besonders gutes dynamisches Verhalten. Dies ist auch nicht gefordert, da der Regler lediglich dafür sorgt, dass der Eingangsraum der Nichtlinearität vollständig durchfahren wird.

Der Eingangsraum wird zwischen $-10[\text{rad/s}] \leq \Omega_2 \leq 10[\text{rad/s}]$ sinusförmig mit einer Frequenz von 0.2 Hz durchfahren. Der Dauer der einzelnen Messreihen beträgt 60 s. Die Implementierung erfolgte auf dem Echtzeitsystem *xPC-Target* mit einer Abtastzeit von 2 ms. Das GRNN besitzt 41 Stützwerte und wird zuerst mit der mit einem Faktor 0.5 gewichteten optimalen Lernschrittweite (siehe Kap. 3.2.6.3) trainiert. Der Glättungsfaktor wurde auf $\sigma_{norm} = 0.8$ eingestellt. Bild 3.22 zeigt den Soll- und Istdrehzahlverlauf. Der Regler kann das Zweimassensystem sicher im Bereich $\pm 10 \text{ rad/s}$ halten, auch wenn die Dynamik an den Drehzahlulldurchgängen (höchste Steigung der Nichtlinearität) nicht besonders gut ist. Die zugehörige Stellgröße ist in Bild 3.22 rechts dargestellt. Das System operiert nahe seiner Stellbegrenzung von $\pm 22 \text{ Nm}$. Der Lernvorgang in den ersten 20 s und das Lernergebnis des GRNN nach 60 s sind in Bild 3.23 dargestellt. Man erkennt sehr gut das Einschwingen der Beobachterdrehzahl auf die gemessene Drehzahl (links); dies kann durch Variation von η beschleunigt und verlangsamt werden. Das Lernergebnis (rechts) bildet die vorgegebene Nichtlinearität sehr gut nach. Zur Darstellung sei angemerkt, dass das in Anhang A experimentell ermittelte Reibmoment jeweils subtrahiert wurde. Es ist damit nur die zusätzliche Nichtlinearität ohne Reibeinfluss identifiziert worden.

Wenn die Lernschrittweite η um 50% größer gewählt wird, so bleibt der Lernvorgang stabil, Messrauschen wird jedoch zusätzlich verstärkt. Dies macht sich am unruhigeren Verlauf des

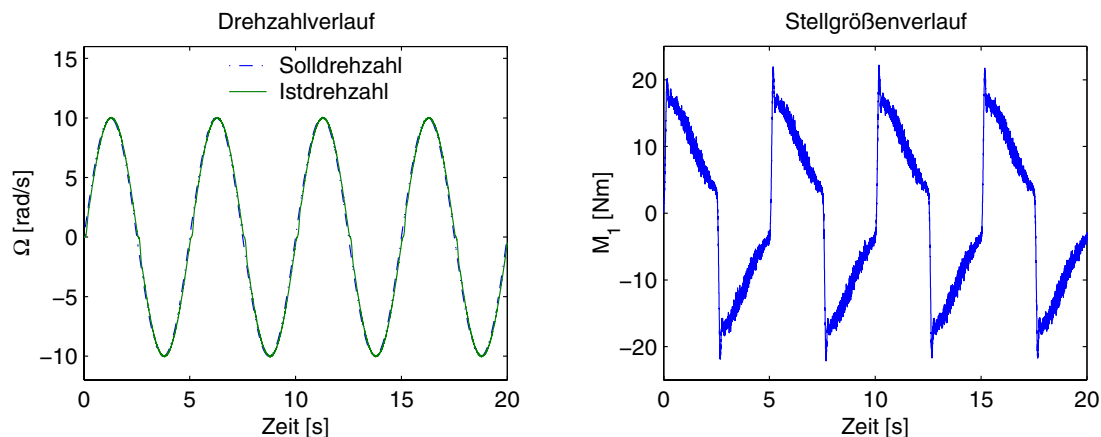


Bild 3.22: Soll- und Ist-Drehzahlverlauf eines Identifikationsvorgangs (links); Stellgrößenverlauf (rechts)

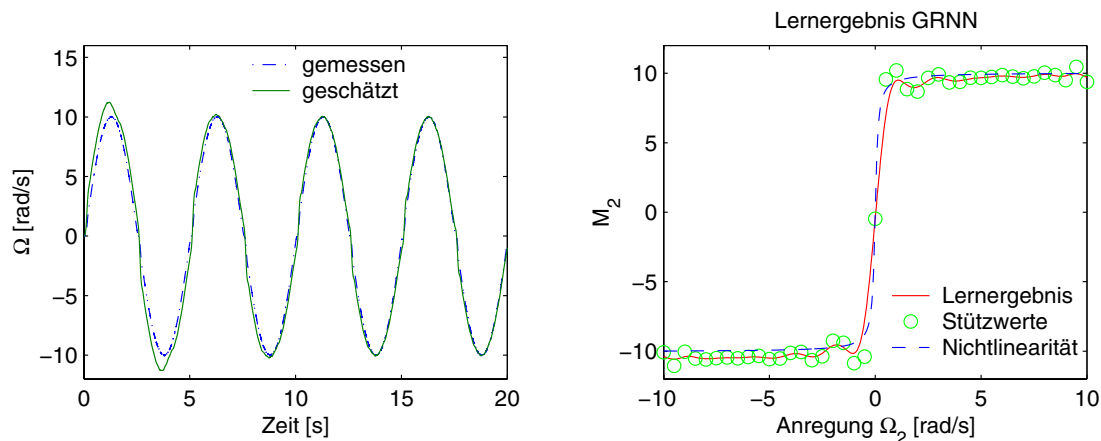


Bild 3.23: Einschwingvorgang der Beobachterdrehzahl (links); Lernergebnis (rechts)

Lernergebnisses in Bild 3.24 in der Nähe des steilen Übergangs der Nichtlinearität bemerkbar. Allerdings ist bereits beim zweiten Durchfahren des Eingangsbereichs die Abweichung von gemessener und beobachteter Drehzahl Ω_2 nahezu verschwunden.

Wird die Lernschrittweite η um 50% kleiner als in Bild 3.22 gewählt, so ist das GRNN nicht mehr in der Lage, die Nichtlinearität nach einer Lerndauer von 20 s ausreichend genau nachzubilden. Bild 3.25 zeigt das (unbefriedigende) Lernergebnis bei kleiner Lernschrittweite (rechts) und die relativ langsame Konvergenz der beobachteten auf die gemessene Drehzahl (links).

Wenn die Stützwertzahl zu gering gewählt wird, leidet darunter die Approximationsgüte. Bild 3.26 rechts zeigt das Lernergebnis bei einer Stützwertzahl von $n_{stütz} = 21$. Im Bereich des steilen Übergangs kann die Nichtlinearität nur ungenügend approximiert werden, während in den Außenbereichen das Lernergebnis mit der vorgegebenen Kennlinie gut übereinstimmt. Dies wirkt sich vor allem bei Drehzahlnulldurchgängen der beobachteten Drehzahl aus. Diese weicht auch nach beendetem Lernvorgang noch deutlich von der

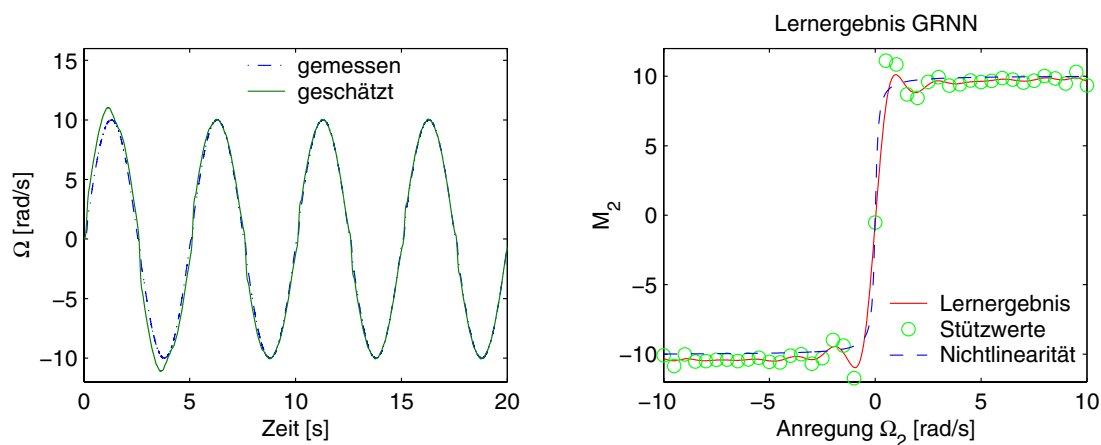


Bild 3.24: Drehzahlverlauf (links) und Lernergebnis (rechts) bei großem η

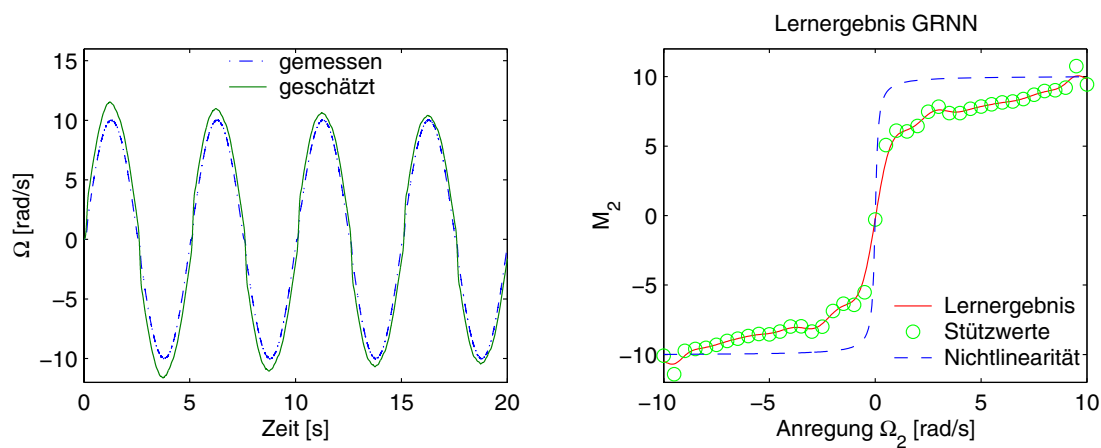


Bild 3.25: Drehzahlverlauf (links) und Lernergebnis (rechts) bei kleinem η

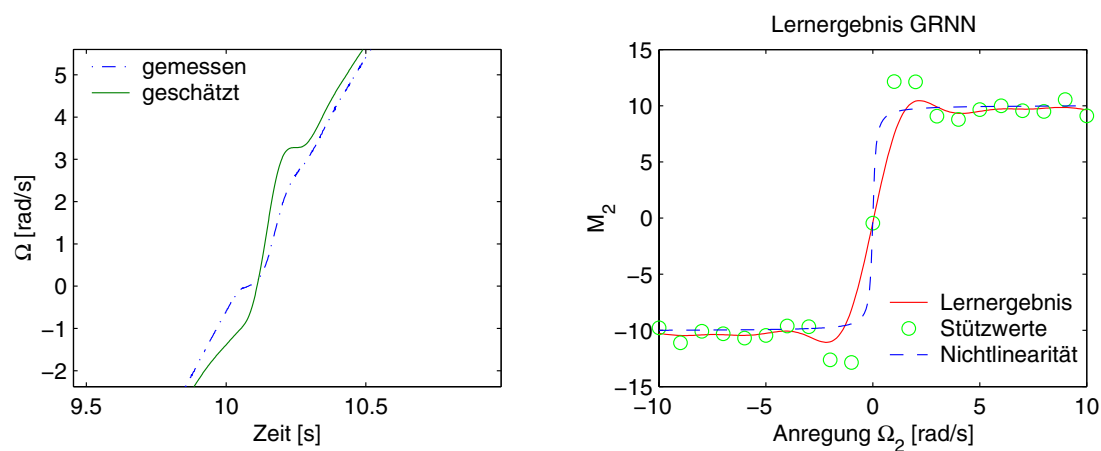


Bild 3.26: Drehzahlverlauf (links) und Lernergebnis (rechts) bei $n_{\text{stütz}} = 21$

gemessenen Drehzahl ab (Bild 3.26 links). Dieses Problem könnte dadurch umgangen werden, dass ein GRNN mit nicht äquidistanter Stützstellenverteilung zum Einsatz kommt. Im Bereich des steilen Übergangs können mehr Stützstellen spendiert werden, als in den Randbereichen. Dadurch spart man trotz guter Approximationsgüte Rechenleistung bei der Implementierung ein.

3.5 Zusammenfassung

In diesem Kapitel wurde ein systematisches Verfahren zum Entwurf lernfähiger Beobachter für eine definierte Klasse nichtlinearer Systeme entwickelt. Die Identifikation von Nichtlinearitäten erfolgt durch neuronale Netze (GRNN). Die Beobachtung und Identifikation laufen garantiert stabil und online ab, was eine Grundvoraussetzung für den Einsatz in Echtzeit-Regelsystemen ist. Zur Identifikation wurde der Beobachter zur Reibungs-, Lose- und Unwuchtidentifikation erfolgreich in [40, 57, 56] praktisch eingesetzt.

Der lernfähige Beobachter liefert für Regelungszwecke folgende verwertbare Informationen:

- Schätzwerte aller Systemzustände
- Approximationen der vorhandenen Nichtlinearitäten

Beides wird in Kap. 6 für eine modellbasierte prädiktive Regelung als notwendige Grundlage genutzt. Das Prozessmodell einer prädiktiven Regelung stellt einen ebenso wichtigen Baustein wie der Reglerentwurf selbst dar. Durch Verwendung des lernfähigen Beobachters muss keine vollständige Zustandsmessbarkeit vorliegen und nichtlineare Effekte müssen nicht vorab bekannt sein.

In Kap. 4 wird die bisherige Modellierungstheorie auf ein spezielles Regelungsproblem angewandt. Das GRNN agiert als unterstützender Regler für Systeme mit periodischen Nichtlinearitäten. Dabei ist besonders die mathematische Beweisbarkeit der stabilen Adaption wichtig, da das GRNN im geschlossenen Regelkreis betrieben wird.

4 Selbstlernender Regler für periodische Nichtlinearitäten

4.1 Einführung

Die Theorie des lernfähigen Beobachters lässt sich für eine spezielle Klasse von Nichtlinearitäten zur direkten Regelung einsetzen. Der hier betrachtete Fall von rotierenden Antriebsanordnungen stellt einen Spezialfall der in Kap. 3.3 untersuchten Systemklasse dar. Es wird angenommen, dass die vorhandenen Nichtlinearitäten periodisch auf das System einwirken. Dies tritt immer bei Unwuchten, Unrundheiten oder sonstigen, mit der Winkelstellung der rotierenden Komponente periodischen, Effekten auf. Die Eigenschaft der Periodizität der Nichtlinearität kann für einen Reglerentwurf gewinnbringend ausgenutzt werden. Wenn der Verlauf eines geeigneten Kompensationssignals in einer Periode bekannt ist, kann sein Verlauf für die folgende Periode vorausgesagt werden und so wieder rechtzeitig aufgeschaltet werden. Der in diesem Abschnitt vorgestellte Regelungsentwurf benötigt zudem keine Information über die Form der angreifenden Nichtlinearität, so dass eine vorherige Identifikation nicht notwendig ist. Als adaptives Reglerelement kommt wieder das in Kap. 3.2.5 eingeführte GRNN zum Einsatz. Dieses generiert direkt ein Kompensationssignal, so dass der Effekt der Nichtlinearität am Systemausgang nicht mehr sichtbar ist. Hier erfolgt statt des Erlernens der Nichtlinearität ein Erlernen eines optimalen Kompensationssignals.

Es wird von einer Strecke mit bekanntem linearen Anteil und einer unbekannten isoliert angreifenden Nichtlinearität ausgegangen. Der Angriffspunkt der Nichtlinearität sei ebenfalls bekannt, ihre exakte Form hingegen wird als unbekannt und evtl. auch als zeitvariant angenommen. Die allgemeine Zustandsgleichung derartiger Systeme mit einem Ein- und Ausgang (SISO) lautet:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A} \cdot \mathbf{x} + \mathbf{b} \cdot u + \mathbf{k} \cdot \mathcal{N}(\mathbf{p}) \\ y &= \mathbf{c}^T \cdot \mathbf{x} + d \cdot u\end{aligned}\tag{4.1}$$

Die Systemmatrix $\mathbf{A}$, der Einkoppelvektor $\mathbf{b}$, der Einkoppelvektor $\mathbf{k}$ der Nichtlinearität, der Ausgangsvektor $\mathbf{c}$ und der Durchgriff d stellen den linearen Systemteil dar und werden als bekannt angenommen. $\mathbf{k}$ beschreibt den Angriffspunkt der Nichtlinearität. Der Vektor $\mathbf{p}$ stellt die Eingangssignale der Nichtlinearität $\mathcal{N}$ dar. In ihm können sowohl interne Zustandsgrößen, als auch externe Einflussgrößen enthalten sein. In Antriebsanordnungen sind typische Elemente von $\mathbf{p}$ die Lage oder Drehzahl von Motoren, externe Einflussgrößen können z.B. Temperatur oder Feuchte sein. Wichtig ist die Mess- oder Beobachtbarkeit (z.B. auch durch den Beobachter aus Kap. 3.3) aller in $\mathbf{p}$ enthaltenen Elemente. Der Angriffspunkt und die Eingangssignale der Nichtlinearität sind also bekannt, ihre Form bzw. ihre Parameter werden jedoch als unbekannt angenommen.

Für derartige Systeme existieren keine allgemeingültigen Einstellregeln für lineare Regler. Regelungen, die durch “ausprobieren” eingestellt wurden, können die beschriebenen Systeme zwar oft stabilisieren, eine definierte Regelgüte kann aber nicht erzielt werden. Eine

auf dem lernfähigen Beobachter aufbauende Zustandsregelung wird in [41, 83] entwickelt. Aufgrund des erhöhten Entwurfsaufwands und weil die periodische Eigenschaft der Nichtlinearität bei dem Verfahren in [41, 83] nicht berücksichtigt wird, soll ein anderer Weg gegangen werden. Für den linearen Streckenteil wird ein linearer Regler nach bekannten Methoden entworfen (Polvorgabe, Riccati-Optimierung, symmetrisches oder Betragsoptimum). Dieser Entwurfsschritt soll nicht näher erläutert werden, sondern wird als bereits durchgeführt betrachtet. Dies hat den Vorteil, dass bestehende Regelungen an in Betrieb befindlichen Anlagen nicht verändert werden müssen. Die Auswirkung der unbekannten Nichtlinearität $\mathcal{NL}(\mathbf{p})$ auf die Ausgangsgröße y wird durch ein zusätzliches Stellsignal $q(\mathbf{p})$ kompensiert. Da die Nichtlinearität nicht bekannt ist, wird zur Erzeugung des Kompensationssignals q ein lernfähiges neuronales Netz verwendet. Aufgrund der einfachen Struktur und der nachweisbar stabilen Lerngesetze wird auch hier das *General Regression Neural Network* (GRNN) [90] aus Kap. 3.2.5 verwendet.

4.2 Entwurf des selbstlernenden Reglers

4.2.1 Streckenbeschreibung

Zunächst werden die Systemgleichungen (4.1) in Übertragungsfunktionen umgewandelt, da sich diese für die weitere Behandlung besser eignen und die grundsätzliche Idee besser deutlich wird. Die Übertragungsfunktion des linearen Systemteils ist unter Verwendung der Einheitsmatrix $\mathbf{E}$ definiert als

$$G(s) = \mathbf{c}^T \cdot [s \cdot \mathbf{E} - \mathbf{A}]^{-1} \cdot \mathbf{b} + d \quad (4.2)$$

$G(s)$ beschreibt somit das Ein– Ausgangsverhalten der Strecke, wenn keine Nichtlinearität vorhanden wäre. Die Übertragungsfunktion, die den Einfluss der Nichtlinearität auf das Ausgangssignal y beschreibt, lautet

$$G_2(s) = \mathbf{c}^T \cdot [s \cdot \mathbf{E} - \mathbf{A}]^{-1} \cdot \mathbf{k} \quad (4.3)$$

Die Übertragungsfunktion $G_1(s)$ ist nun so definiert, dass gilt

$$G(s) = G_1(s) \cdot G_2(s) \quad (4.4)$$

Als Regelung der beschriebenen Strecke wird eine konventionelle Ausgangsgrößenregelung angenommen; es könnte z.B. ein P–, PI–, oder PID–Regler verwendet werden. Die Reglerübertragungsfunktion wird mit $R(s)$ bezeichnet.

Ist diese Partitionierung der Regelstrecke möglich, so ergibt sich die Struktur in Bild 4.1. Der Sollwert des Regelkreises ist w , die Regelgröße ist y und die zur Verfügung stehende Stellgröße ist u . Ausgehend von der Struktur in Bild 4.1 wird nun der selbstlernende Regler hergeleitet.

Die Einstellvorschriften des linearen Reglers $R(s)$ sollen hier nicht weiter behandelt werden. Alle linearen Reglersyntheseverfahren, wie etwa Polvorgabe [60], Riccati-Optimierung

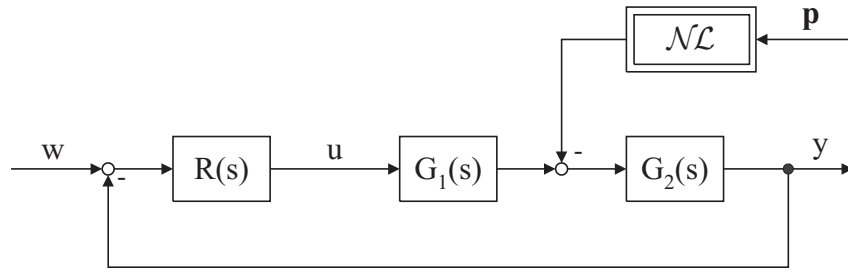


Bild 4.1: Betrachtete Streckenstruktur mit Regler

[24], Dämpfungsoptimum oder symmetrisches Optimum [29] sind anwendbar. Die Nichtlinearität wird zwar zu einer Abweichung im Regelergebnis führen, diese wird jedoch anschließend vom selbstlernenden Regler kompensiert. Für das vorgestellte Verfahren ist es nicht notwendig, dass der lineare Streckenteil $G(s)$ asymptotisch stabil ist; es muss lediglich sichergestellt sein, dass das Gesamtsystem einschließlich Regler mit und ohne angreifende Nichtlinearität asymptotisch stabil ist. Diese Voraussetzung wird für den Start des Lernvorgangs benötigt, da zu diesem Zeitpunkt eine Kompensation der Nichtlinearität noch nicht erfolgt.

Der Ausgangswert y des Systems in Bild 4.1 berechnet sich nach Gleichung (4.5).

$$y = \frac{R(s) \cdot G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot w(s) - \frac{G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot \mathcal{NL}(\mathbf{p})(s) \quad (4.5)$$

Dabei ist $\mathcal{NL}(\mathbf{p})$ das Ausgangssignal des Blocks $\mathcal{NL}$ in Bild 4.1. Es wird als Zeitsignal betrachtet, das in den Laplace-Raum transformiert werden kann. Das in den Laplace-Raum transformierte Signal wird mit $\mathcal{NL}(\mathbf{p})(s)$ bezeichnet. Diese Darstellung darf nicht darüber hinweg täuschen, dass der Messvektor $\mathbf{p}$ nicht nur aus unabhängigen Größen, sondern auch aus internen Zuständen der Übertragungsfunktionen $G_1(s)$ und $G_2(s)$ bestehen kann.

4.2.2 Selbstlernende Kompensation

Ziel der selbstlernenden Kompensation ist die vollständige Unterdrückung der Auswirkungen der Nichtlinearität $\mathcal{NL}$ am Streckenausgang y . Dies wird durch eine Modifikation der Stellgröße u erreicht. Prinzipiell sind für dieses Vorgehen drei Verfahren zu unterscheiden:

1. Offline-Identifikation der Nichtlinearität und anschließende analytische Berechnung eines entsprechenden Kompensationsgesetzes.
2. Online-Identifikation der Nichtlinearität in einem lernfähigen Beobachter nach Kap. 3.3. Das jeweils aktuelle Identifikationsergebnis wird dann wie in [40, 56] einem statischen Kompensationsregelgesetz zugeführt.
3. Online-Identifikation des benötigten Kompensationssignals durch Minimierung eines Fehlersignals, das die Regelgüte bewertet. Dazu ist eine vorherige Identifikation der Nichtlinearität **nicht** nötig.

Hier wird die Online-Identifikation des Kompensationssignals weiter verfolgt. Gegenüber der Online-Identifikation der Nichtlinearität durch einen lernfähigen Beobachter entfällt hier der zusätzliche Beobachterentwurf und der dadurch bedingte Rechenaufwand. Außerdem ist ein fest eingestelltes Kompensationsfilter empfindlich auf Unsicherheiten der Streckenparameter, und die hier angenommene Periodizität der Nichtlinearität wird nicht ausgenutzt. Bei einem Kompensationsfilter nach [40] besteht außerdem die Problematik, die Dynamik zwischen Eingriffspunkt der Stellgröße und der Nichtlinearität zu invertieren, was häufig zu differenzierenden Gliedern führt. Bei periodischen Nichtlinearitäten kann diese Periodizität dazu ausgenutzt werden, das Kompensationssignal ebenfalls durch eine statische Funktion darzustellen. Eine Invertierung der Streckendynamik ist dann nicht mehr nötig.

Die Modifikation der Stellgröße wird nach Gleichung (4.6) vorgenommen.

$$u = u' + q(\mathbf{p}) \quad (4.6)$$

u' ist das Ausgangssignal des linearen Reglers $R(s)$. Das Kompensationssignal $q(\mathbf{p})$ wird, wie die Nichtlinearität, als von $\mathbf{p}$ abhängig angesetzt. Eine Begründung für diese Wahl wird aus folgender Überlegung deutlich: Würde das Kompensationssignal am selben Punkt wie die Nichtlinearität angreifen (Punkt **K** in Bild 4.2), so wäre das ideale Kompensationssignal $q(\mathbf{p}) = \mathcal{NL}(\mathbf{p})$. Es wäre demnach eine Funktion von $\mathbf{p}$. Da zwischen den Angriffspunkten des Kompensationssignals und der Nichtlinearität jedoch die lineare Übertragungsfunktion $G_1(s)$ wirksam ist, erfolgt dadurch eine dynamische Beeinflussung, die sich durch eine Dämpfung und Phasenverschiebung des periodischen Kompensationssignals $q(\mathbf{p})$ bemerkbar macht. Die Periode von $q(\mathbf{p})$ und damit auch die Abhängigkeit von $\mathbf{p}$ wird durch die lineare Übertragungsfunktion $G_1(s)$ aber nicht geändert.

Das Kompensationssignal $q(\mathbf{p})$ wird adaptiv während des Betriebs generiert. Zum Erlernen des optimalen Kompensationssignals $q(\mathbf{p})$ wird ein *General Regression Neural Network* (GRNN) verwendet. Dieses hat die Eigenschaft eines universellen Funktionsapproximators und kann durch garantiert stabile Lerngesetze online trainiert werden. Details zu Lerngesetzen und Stabilitätsuntersuchungen finden sich in Kap. 3.2.6.4.

Die verwendete Kompensationsstruktur ist in Bild 4.2 dargestellt. Das GRNN generiert das Kompensationssignal $q(\mathbf{p})$ derart, dass die resultierende Stellgröße u den Einfluss der Nichtlinearität $\mathcal{NL}(\mathbf{p})$ auf die Regelgröße y über die Übertragungsfunktion $G_1(s)$ hinweg vollständig unterdrücken kann. Das Ausgangssignal y unter Berücksichtigung des Kompensationssignals $q(\mathbf{p})$ kann nun gemäß Bild 4.2 nach Gleichung (4.7) berechnet werden.

$$\begin{aligned} y(s) &= \frac{R(s) \cdot G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot w(s) - \frac{G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot \mathcal{NL}(\mathbf{p})(s) \\ &+ \frac{G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot q(\mathbf{p})(s) \end{aligned} \quad (4.7)$$

Wenn die Nichtlinearität durch das Kompensationssignal $q(\mathbf{p})$ vollständig unterdrückt ist, dann gilt für $q(\mathbf{p})$ und $\mathcal{NL}(\mathbf{p})$ der folgende Zusammenhang:

$$\frac{G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot \mathcal{NL}(\mathbf{p})(s) = \frac{G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot q(\mathbf{p})(s) \quad (4.8)$$

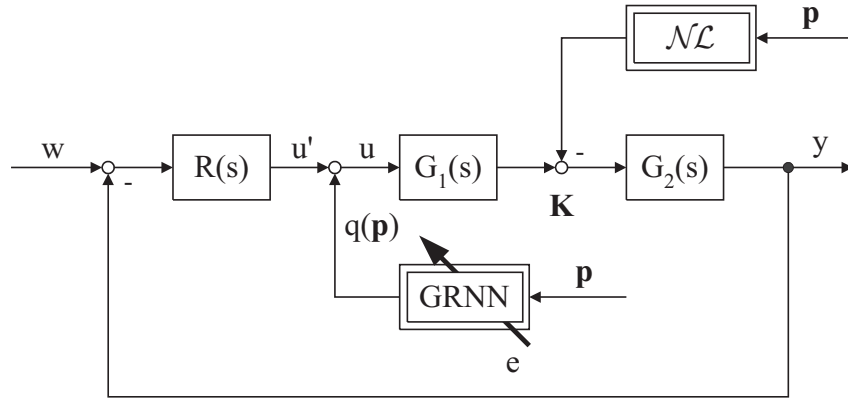


Bild 4.2: Struktur zur Kompensation von $\mathcal{NL}(\mathbf{p})$

Dieser Zusammenhang lässt sich vereinfachen zu

$$G_1(s) \cdot q(\mathbf{p})(s) = \mathcal{NL}(\mathbf{p})(s) \quad (4.9)$$

Für die Beziehung zwischen Sollwert und Regelgröße ergibt sich demgemäß **bei optimaler Kompensation** der Nichtlinearität die Beziehung in Gleichung (4.10).

$$y_{opt}(s) = \frac{R(s) \cdot G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot w(s) \quad (4.10)$$

Die rechte Seite von Gleichung (4.10) stellt somit die Wunsch-Systemantwort auf eine Sollwertänderung w dar. Als Fehlersignal, welches eine Aussage über die Güte der erreichten Kompensation der Nichtlinearität liefert, wird die Differenz zwischen optimalem und tatsächlichem Ausgangssignal verwendet.

$$e(s) = y(s) - y_{opt}(s) = y(s) - \frac{R(s) \cdot G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \cdot w(s) \quad (4.11)$$

Dieses Fehlersignal wird zum Training des neuronalen Netzes verwendet, welches das Kompensationssignal $q(\mathbf{p})$ online erzeugt. Aus Gleichung (4.11) ist sofort ersichtlich, dass bei optimaler Kompensation ($y = y_{opt}$) das Fehlersignal e gleich null sein muss. Unter Verwendung von Gleichung (4.5) und (4.11) lässt sich das Fehlersignal e gemäß Gleichung (4.12) darstellen. Im Folgenden wird zur besseren Übersicht teilweise auf den Zusatz “(s)” bei Laplace-Größen verzichtet. Diese, in der Literatur übliche, hybride Notation bedeutet jedoch stets, dass bei Verwendung von Übertragungsfunktionen der Laplace-Bereich gemeint ist.

$$e = y - \frac{R \cdot G_1 \cdot G_2}{1 + R \cdot G_1 \cdot G_2} \cdot w = \frac{G_1 \cdot G_2}{1 + R \cdot G_1 \cdot G_2} \cdot q(\mathbf{p}) - \frac{G_2}{1 + R \cdot G_1 \cdot G_2} \cdot \mathcal{NL}(\mathbf{p}) \quad (4.12)$$

Durch Einführung der Fehlerübertragungsfunktion $H(s)$ nach Gleichung (4.13) vereinfacht sich das Fehlersignal e gemäß Gleichung (4.14).

$$H(s) = \frac{G_1(s) \cdot G_2(s)}{1 + R(s) \cdot G_1(s) \cdot G_2(s)} \quad (4.13)$$

$$e = H(s) \cdot \left[q(\mathbf{p}) - \underbrace{G_1^{-1}(s) \cdot \mathcal{NL}(\mathbf{p})}_{\mathcal{NL}^*(\mathbf{p})} \right] \quad (4.14)$$

Das Kompensationssignal $q(\mathbf{p})$ wird durch das neuronale Netz derart erzeugt, dass es gleich der modifizierten Nichtlinearität $\mathcal{NL}^*(\mathbf{p})$ ist und somit Gleichung (4.9) erfüllt ist.

4.2.2.1 Approximation des Kompensationssignals durch ein neuronales Netz

Neuronale Netze haben grundsätzlich die Eigenschaft nichtlineare statische Funktionen nachbilden zu können. Je nach Netzwerktyp und Auslegung des Netzes variiert die Approximationsgüte. Für die vorliegende Aufgabe können prinzipiell alle Arten von Netzwerken verwendet werden, die online trainiert werden können. Als Beispiele seien *Multilayer Perceptron Netze* (MLP), Netzwerke mit *radial symmetrischen Basisfunktionen* (RBF) oder das *General Regression Neural Network* (GRNN) genannt. Für Online-Regelungszwecke muss besonderes Augenmerk auf garantierte Stabilität des Lernvorgangs und auch auf Interpretierbarkeit der Netzwerkgewichte zur Online-Überwachung des Lernergebnisses gelegt werden. Für mehrdimensionale Eingangsräume der nichtlinearen Funktion $\mathcal{NL}(\mathbf{p})$ hat das MLP Netzwerk wegen des relativ geringen Rechenaufwands einen großen Vorteil; die Interpretierbarkeit der Gewichte ist allerdings nicht gegeben. Für geringe Eingangsdimensionalität m ($1 \leq m \leq 3$) weist das GRNN sehr gute Eigenschaften durch die Interpretierbarkeit jedes einzelnen Netzwerkgewichtes auf. Außerdem ist es möglich, für die Netzwerkgewichte ein nach Lyapunov stabiles Lerngesetz abzuleiten. Da der Netzwerkausgang gemäß Bild 4.2 direkt in den Regelkreis eingreift, ist dies neben der Interpretierbarkeit der Gewichte das Hauptargument für die Verwendung des GRNN als Funktionsapproximator für $q(\mathbf{p})$. Auf genauere Ausführungen soll hier verzichtet werden und auf Kap. 3.2.5 und [36, 46, 90] verwiesen werden.

Da das GRNN nach Kap. 3.2.5 in der Lage ist jede stetige Funktion nachzubilden, kann die tatsächlich vorhandene Nichtlinearität $\mathcal{NL}(\mathbf{p})$ als ein bereits optimal trainiertes GRNN aufgefasst werden. Der in Gleichung (4.15) eingeführte Approximationsfehler a_i wird im Folgenden stets vernachlässigt, da er durch erhöhte Stützwerteanzahl beliebig verringert werden kann.

$$\mathcal{NL}(\mathbf{p}) = \Theta^T \cdot \mathcal{A}(\mathbf{p}) + a_i \quad (4.15)$$

Θ ist der zeitlich konstante Stützwertevektor eines fiktiven GRNN, das die tatsächlich vorhandene Nichtlinearität optimal approximiert.

Das Kompensationssignal $q(\mathbf{p})$ wird durch das zu adaptierende GRNN mit den Stützwerten $\hat{\Theta}$ dargestellt, d.h. es gilt:

$$q(\mathbf{p}) = \hat{\Theta}^T \cdot \mathcal{A}(\mathbf{p}) \quad (4.16)$$

Um mit Gleichung (4.14) ein nach Lyapunov stabiles Lerngesetz abzuleiten, muss der Term $G_1^{-1}(s) \cdot \mathcal{NL}(\mathbf{p})$ durch eine statische Funktion nachgebildet werden können. Dies ist im Allgemeinen nicht möglich, da eine dynamische Beeinflussung des Kompensationssignals q durch die lineare Übertragungsfunktion $G_1^{-1}(s)$ erfolgt. Bei beliebiger Form von q ist $G_1^{-1}(s) \cdot \mathcal{NL}(\mathbf{p})$ nicht mehr nur eine Funktion von $\mathbf{p}$, sondern hängt auch explizit von der Zeit t ab. Dies ergibt sich aus der Lösung der linearen Differentialgleichung, die die Übertragungsfunktion G_1^{-1} beschreibt. Insbesondere spielen auch die Anfangswerte von

G_1^{-1} eine Rolle. Für periodische Signale $\mathcal{NL}(\mathbf{p})$ vereinfacht sich diese Situation. In diesem Fall setzt sich die Lösung der zugehörigen linearen Differentialgleichung aus einem transienten und einem periodischen Anteil zusammen. Der transiente Anteil klingt für eine asymptotisch stabile Übertragungsfunktion $G_1^{-1}(s)$ ab und es verbleibt nur noch die periodische Lösung, wobei sich die Periodendauer gegenüber dem Eingangssignal $\mathcal{NL}(\mathbf{p})$ nicht ändert. Das durch $G_1^{-1}(s)$ beeinflusste Ausgangssignal der Nichtlinearität $\mathcal{NL}(\mathbf{p})$ wird mit $\mathcal{NL}^*$ bezeichnet.

$$\mathcal{NL}^*(\mathbf{p}) = G_1^{-1}(s) \cdot \mathcal{NL}(\mathbf{p}) \quad (4.17)$$

Für die weiteren Betrachtungen werden nun folgende Voraussetzungen getroffen:

- Der Eingangsvektor $\mathbf{p}$ ist periodisch mit der Periodendauer T , d.h. $\mathbf{p}(t+T) = \mathbf{p}(t)$.
- Da $\mathcal{NL}(\mathbf{p})$ eine statische Funktion ist, gilt $\mathcal{NL}(\mathbf{p}) = \mathcal{NL}(\mathbf{p} + \mathbf{p}_p)$. Hier sind die Elemente von $\mathbf{p}_p$ die Perioden von $\mathcal{NL}$ in $\mathbf{p}$ -Koordinaten. Für $\mathbf{p}_p$ ergibt sich somit $\mathbf{p}_p = \mathbf{p}(T)$.
- Die tatsächlichen Werte von T und $\mathbf{p}_p$ müssen nicht bekannt sein, es muss nur sichergestellt sein, dass Periodizität vorliegt.

Die genannten Voraussetzungen sind immer bei rotierenden Maschinen mit konstanter Drehzahl gegeben, wenn die Nichtlinearität von Winkelgrößen der rotierenden Achsen abhängt. Die Periodendauer ist in diesen Fällen $T = 1/N$ (N : Drehzahl der Maschine) und $p_p = 2\pi$. Dieser Fall tritt bei Kolbenkompressoren, unrunder Abwicklern in kontinuierlichen Fertigungsanlagen oder Kurbelwellen-Starter-Generator Anordnungen in der Automobiltechnik auf.

Die Kompensation der Nichtlinearität $\mathcal{NL}$ durch das Signal q über die Übertragungsfunktion G_1 hinweg muss ebenfalls durch ein mit der selben Periode T periodisches Signal erfolgen, da die Periodendauer des Kompensationssignals q durch die lineare Übertragungsfunktion G_1 nicht verändert wird. Bild 4.3 ist bzgl. des Ein-Ausgangsverhaltens identisch zu Bild 4.2. Der Signalfussplan wurde lediglich so umgeformt, dass Nichtlinearität und Kom-

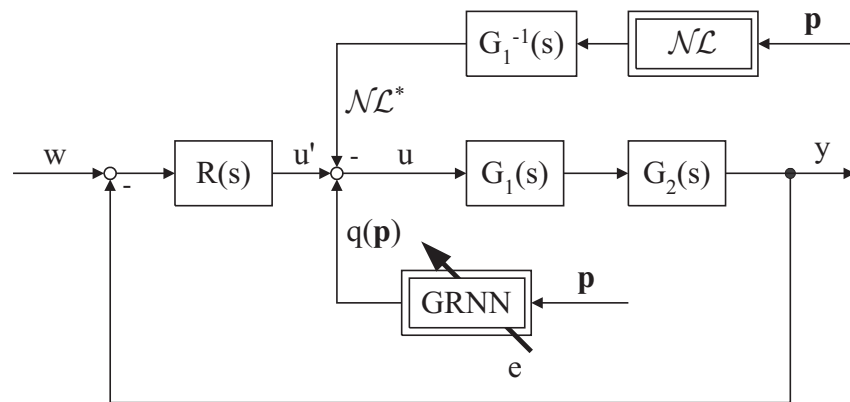


Bild 4.3: Struktur zur Kompensation mit modifizierter Nichtlinearität $\mathcal{NL}^*$

pensationssignal am selben Punkt angreifen. Da das Kompensationssignal $q(\mathbf{p})$ als statische

Funktion angenommen wurde, ist für eine ideale Kompensation zu klären, ob dies für die modifizierte Nichtlinearität $\mathcal{N}^*$ ebenfalls gilt.

Aus Gleichung (4.17) geht hervor, dass $\mathcal{N}^*$ aus $\mathcal{N}$ über die lineare Übertragungsfunktion $G_1^{-1}(s)$ hervorgeht. Das Signal $\mathcal{N}(\mathbf{p}(t))$ ist periodisch mit der Periodendauer T . Die lineare Übertragungsfunktion bewirkt eine Amplitudenänderung und Phasenverschiebung ohne Änderung der Periodendauer. Diese Eigenschaft gilt für alle linearen Übertragungsfunktionen nach Abklingen von transienten Vorgängen. $\mathcal{N}^*$ ist daher wieder periodisch mit der gleichen Periodendauer T und kann ebenfalls als statische Funktion mit dem Eingang $\mathbf{p}$ dargestellt werden. Bei der Implementierung eines GRNN ist es aufgrund der Periodizität der Eingangssignale $\mathbf{p}$ ausreichend, den Eingangsbereich des Netzwerkes auf $\mathbf{0} \leq \mathbf{p} \leq \mathbf{p}_p$ zu beschränken. Diese Maßnahme reduziert die Stützwertezahl und somit auch den Rechenaufwand.

4.2.2.2 Stabiles Lerngesetz

Nachdem nun gezeigt ist, dass sich das Signal $\mathcal{N}^*(\mathbf{p})$ im eingeschwungenen Zustand durch eine statische Funktion nachbilden lässt, obwohl eine dynamische Beeinflussung von $\mathcal{N}$ nach $\mathcal{N}^*$ durch $G_1^{-1}(s)$ erfolgt, kann auf dieser Eigenschaft aufbauend ein stabiles Lerngesetz für das GRNN angegeben werden. Aus der Fehlergleichung (4.14) werden sofort zwei Voraussetzungen sichtbar.

1. Die Funktion $\mathcal{N}(\mathbf{p})$ muss ausreichend oft differenzierbar sein, da der relative Grad von $G_1(s)$ häufig größer als null ist und damit bei der Bildung von $G_1^{-1}(s)$ differenzierende Anteile entstehen.
2. $G_1(s)$ darf nur Nullstellen in der linken s-Halbebene besitzen, d.h. $G_1(s)$ muss minimalphasig sein. Ansonsten würden instabile Polstellen in $G_1^{-1}(s)$ entstehen.

Diese Voraussetzungen bedeuten nicht, dass der Term $\mathcal{N}^*(\mathbf{p}) = G_1^{-1}(s) \cdot \mathcal{N}(\mathbf{p})$ durch numerische Differentiationsverfahren bestimmt werden muss, es kommt lediglich auf seine Existenz an. Es wird nun ein fiktives, optimal trainiertes GRNN nach Gleichung (4.18) mit konstanten Stützstellen angenommen, das die modifizierte Nichtlinearität $\mathcal{N}^*$ optimal approximiert. Dadurch gelingt die Herleitung eines global stabilen Lerngesetzes für die Stützwerte $\hat{\Theta}$ des Kompensationssignals q .

$$\mathcal{N}^*(\mathbf{p}) = G_1^{-1}(s) \cdot \mathcal{N}(\mathbf{p}) = \Theta^{*T} \cdot \mathcal{A}(\mathbf{p}) \quad (4.18)$$

In Gleichung (4.14) kann nun $G_1^{-1} \cdot \mathcal{N}$ durch das optimal trainierte GRNN mit zeitlich konstanten Stützwerten gemäß Gleichung (4.18) ersetzt werden. Das Kompensationssignal q wird durch das zu trainierende GRNN nach Gleichung (4.16) ersetzt. Mit der Fehlerübertragungsfunktion $H(s)$ ergibt sich schließlich:

$$e = H(s) \cdot \left[\hat{\Theta}^T \cdot \mathcal{A}(\mathbf{p}) - \Theta^{*T} \cdot \mathcal{A}(\mathbf{p}) \right] = H(s) \cdot \left[\hat{\Theta}^T - \Theta^{*T} \right] \cdot \mathcal{A}(\mathbf{p}) \quad (4.19)$$

Für die Fehlergleichung (4.19) lassen sich je nach den Eigenschaften von $H(s)$ unterschiedliche Lerngesetze für die Netzwerkgewichte herleiten. Wenn die Übertragungsfunktion $H(s)$

strictly positive real gemäß Definition 3.5 ist, lautet das stabile Adaptionsgesetz für die Stützwerte des GRNN (siehe Kap. 3.2.6.4 und [17]):

$$\dot{\hat{\Theta}} = -\eta \cdot e \cdot \mathcal{A}(\mathbf{p}) \quad (4.20)$$

Bei beliebiger Fehlerübertragungsfunktion muss ein zusätzliches Fehlersignal zur Adaption gebildet werden. Das Lerngesetz lautet dann allgemein:

$$\dot{\hat{\Theta}} = -\eta \cdot \epsilon \cdot H(s) \cdot \mathbf{E} \cdot \mathcal{A}(\mathbf{p}) \quad \text{mit} \quad \epsilon = \left(\hat{\Theta}^T - \Theta^{*T} \right) \cdot H(s) \cdot \mathbf{E} \cdot \mathcal{A}(\mathbf{p}) \quad (4.21)$$

Die Lerngesetze der Gleichungen (4.20) und (4.21) garantieren Stabilität gemäß Lyapunov, d.h. die Fehlersignale e und ϵ streben für große t gegen null.

$$\lim_{t \rightarrow \infty} e(t) = 0 \quad \lim_{t \rightarrow \infty} \epsilon(t) = 0 \quad (4.22)$$

Die Konvergenzgeschwindigkeit kann in beiden Fällen durch entsprechende Wahl der Lernschrittweite η eingestellt werden ($\eta > 0$). Je schlechter die Qualität der Messsignale ist, desto kleiner muss η gewählt werden um durch das Lerngesetz eine Filter- bzw. Glättungswirkung zu erzielen. Die Gleichungen (4.19) und (4.20) bzw. (4.21) bilden ein global stabiles Fehlermodell mit dem für den selbstlernenden Regler ein stabiler Adaptionalgorithmus zur Verfügung steht. Die gesamte Kompensationsstruktur einschließlich Fehlerbildung ist in Bild 4.4 dargestellt. Das GRNN unterstützt den konventionellen Regler bei der Ausre-

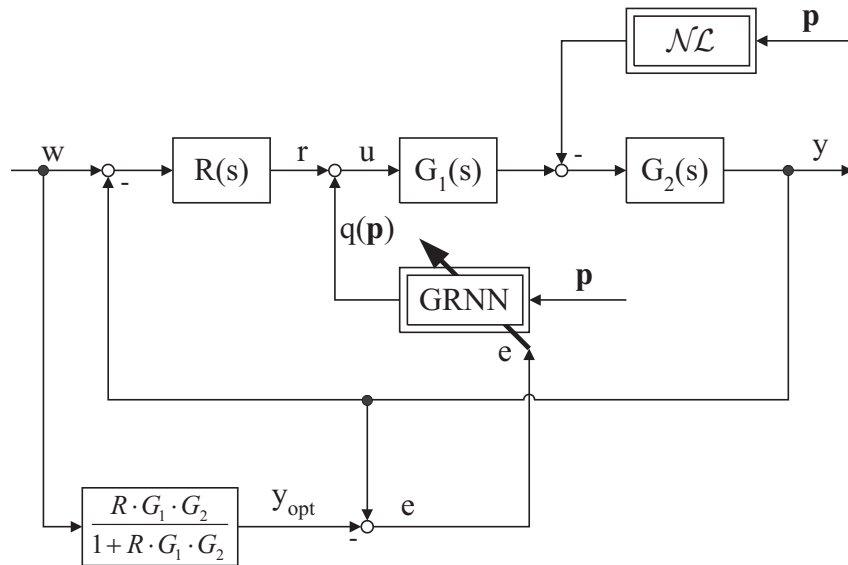


Bild 4.4: Kompensation von $\mathcal{NL}$ durch den selbstlernenden Regler einschließlich Generierung des Fehlersignals

gelung von periodischen Nichtlinearitäten. Alle sonstigen Regelaufgaben verbleiben beim bereits vorhandenen Regler $R(s)$. Typische Anwendungen sind elektrische Antriebsanordnungen mit winkelabhängigen Nichtlinearitäten. Die Störungen sind dann 2π -periodisch. Zu beachten ist, dass für das vorgestellte Verfahren kein spezieller Regler $R(s)$ benötigt wird, vielmehr kann die bestehende Regelung einer Anlage beibehalten werden. Dies ist insbesondere für Zwecke der Nachrüstung interessant.

4.2.3 Zusammenfassung

Der selbstlernende Regler ist in der Lage isolierte Nichtlinearitäten zu kompensieren, die periodisch auf die Regelstrecke einwirken. Dies erfolgt online während des Betriebs. Das erforderliche Kompensationssignal wird durch ein GRNN erzeugt, wobei die Netzwerkgewichte durch ein stabiles Lerngesetz adaptiert werden. Dafür ist es notwendig, dass die Strecke durch einen konventionellen Regler bereits stabilisiert wird, so dass das System in Betrieb genommen werden kann.

Für die Anwendbarkeit des vorgestellten Regelungsverfahrens müssen folgende Voraussetzungen erfüllt sein.

- Darstellbarkeit der Strecke gemäß Bild 4.1.
- Die angreifende Nichtlinearität muss periodisch auf die Strecke einwirken, d.h. es muss gelten: $\mathcal{N}(\mathbf{p}) = \mathcal{N}(\mathbf{p} + \mathbf{p}_p)$ ($\mathbf{p}_p$ ist die Periode in $\mathbf{p}$ -Koordinaten).
- Die Übertragungsfunktion $G_1(s)$ darf nur in der linken komplexen s -Ebene Nullstellen besitzen, d.h. sie muss minimalphasig sein. Dies ist für die Existenz des stabilen Lerngesetzes notwendig.
- Die Kenntnis von $\mathbf{p}_p$ ist für die Auslegung des GRNN von Vorteil. Wegen der Periodizität genügt es, den Eingangsraum zwischen $\mathbf{0} \leq \mathbf{p} \leq \mathbf{p}_p$ vorzusehen. In diesem Bereich werden die Stützwerte des GRNN verteilt.

Da bei diesem Verfahren das Kompensationssignal adaptiv während des Betriebs generiert wird, ist es möglich zeitveränderliche Nichtlinearitäten zu kompensieren. Das GRNN passt sich dann an die jeweils aktuell wirkende Störgröße an. Das Verfahren setzt keinen neuen Regelungsentwurf einer bestehenden Anlage voraus, sondern unterstützt den bestehenden Regler. Aufgabe des selbstlernenden Reglers ist somit nur die Kompensation der Störungen, alle sonstigen Regelungsaufgaben verbleiben beim ursprünglichen Regler $R(s)$.

Bei dem vorgestellten Ansatz kann bei einem Arbeitspunktwechsel die Periodendauer T bzw. die Grundfrequenz $\omega_0 = 2\pi/T$ verändert werden. Dadurch ändert sich Verstärkung und Phasenverschiebung bei der Beeinflussung in Gleichung (4.17). Dies bedeutet, die Stützwerte des Kompensationssignals $q(\mathbf{p})$ müssen nachgelernt werden. Eine Abhilfe besteht darin, dass das GRNN eine zusätzliche Eingangsdimension erhält und das Kompensationssignal q für mehrere Periodendauern T gelernt wird. Zwischenwerte können dann interpoliert werden und der Bedarf des Nachlernens der Stützwerte wird verringert. Diese Möglichkeit wird in Kap. 4.3.2 an einem elektrischen Antrieb mit periodischem Gegenmoment ausgeführt.

Die Leistungsfähigkeit des selbstlernenden Reglers soll nun an zwei Anwendungen untersucht werden. In Kap. 4.3 wird simulativ ein elektrischer Antrieb mit Stellglied untersucht, in Kap. 4.4 wird die Kompensation von Wicklerunrundheiten an einer kontinuierlichen Fertigungsanlage praktisch durchgeführt.

4.3 Elektrischer Antrieb mit Stellglied

Häufig werden Teilsysteme einer elektrischen Antriebsanordnung als leistungselektronisches Stellglied gefolgt von der Mechanik der Antriebsmaschine modelliert. Der nicht explizit untersuchte Stromregelkreis wird als PT_1 -Glieder mit der Zeitkonstanten T_i modelliert. Die Trägheit J der Antriebsmaschine stellt der Integrator mit dem Verstärkungsfaktor $1/J$ dar. Ausgangs- und Regelgröße ist die Motordrehzahl Ω . Die Messung der Winkellage ϕ des Motors erfordert normalerweise keinen zusätzlichen Sensor, da sie bei Inkrementalmessgebern immer als primäre Messgröße zur Verfügung steht. Die Regelung dieses Einmassensystems erfolgt durch einen PI-Regler, der nach dem symmetrischen Optimum (SO) eingestellt ist [29]. Zur Reduktion des Überschwingens des SO-Regelkreises wurde eine Führungsglättung eingeführt. Als Nichtlinearität greift ein winkelabhängiges Gegenmoment an, so dass periodische Drehzahlschwankungen entstehen. Dieses Gegenmoment könnte etwa das lageabhängige Gegenmoment eines Kolbenkompressors sein. Die gesamte Anordnung zeigt Bild 4.5. Als Regler dieses Einmassensystems wird ein PI-Regler mit

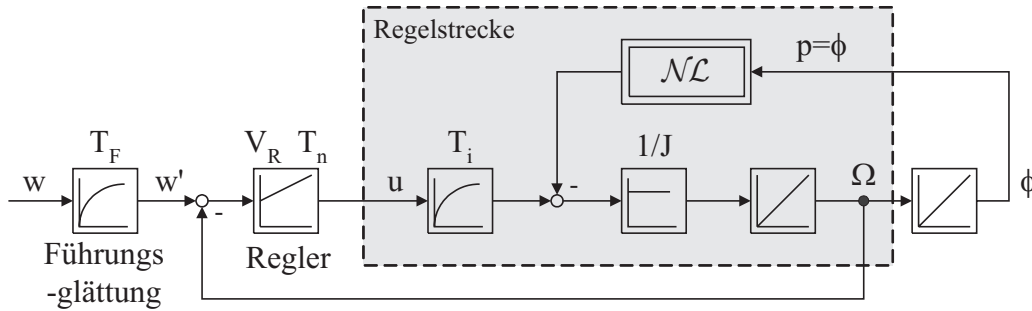


Bild 4.5: Antriebssystem mit winkelabhängigem Gegenmoment

folgender Übertragungsfunktion verwendet:

$$R(s) = V_R \cdot \frac{1 + T_n s}{T_n s} \quad (4.23)$$

Gemäß den vorherigen Abschnitten ergeben sich folgende Ausdrücke für die Funktionen $G_1(s)$ und $G_2(s)$.

$$G_1(s) = \frac{1}{1 + T_i \cdot s} \quad G_2(s) = \frac{1}{J \cdot s} \quad (4.24)$$

Das winkelabhängige Gegenmoment wird nach Gleichung (4.25) vorgegeben und als unbekannt angenommen. Die Kompensation dieser Störgröße erfolgt durch den selbstlernenden Regler. Der Winkel ϕ , der Eingangsgröße sowohl für die nichtlineare Funktion $\mathcal{NL}$ als auch für den selbstlernenden Regler ist, wird in der Praxis durch einen Inkrementalgeber gemessen.

$$\mathcal{NL}(\phi) = M_{\text{gegen}} \cdot \sin^2(\phi) \quad 0 \leq \phi \leq \pi \quad \mathcal{NL}(\phi) = 0 \quad \text{sonst} \quad (4.25)$$

Die Kompensation wird durch die Modifikation der Stellgröße gemäß $u = u' + q$ erreicht. Das Kompensationssignal q wird von einem GRNN online generiert. Es wird ein GRNN mit 40 Stützwerten mit einem Glättungsfaktor $\sigma = \text{“Stützwerteabstand“}$ verwendet. Der Eingangsraum des GRNN liegt zwischen $0 \leq \phi \leq 2\pi$. Die Stützwerte werden in diesem Bereich äquidistant verteilt.

4.3.1 Kompensation bei konstantem Sollwert

Der Sollwert w der Regelung sei zunächst konstant (Regelung auf konstante Drehzahl), so dass die Führungsglättung keinen Einfluss hat. Das Fehlersignal e ergibt sich dann zu:

$$e = \Omega - w = \Omega - w' \quad (4.26)$$

Die für das Lerngesetz benötigte Fehlerübertragungsfunktion $H(s)$ kann aufgrund der bisherigen Angaben berechnet werden.

$$H(s) = \frac{G_1 \cdot G_2}{1 + R \cdot G_1 \cdot G_2} = \frac{T_n \cdot s}{J \cdot T_n \cdot T_i \cdot s^3 + J \cdot T_n \cdot s^2 + V_R \cdot T_n \cdot s + V_R} \quad (4.27)$$

Bild 4.6 zeigt die zur Kompensation mittels selbstlernendem Regler verwendete Struktur. Das Lerngesetz nach Gleichung (4.21) mit der Fehlerübertragungsfunktion $H(s)$ ist im Block "GRNN" integriert. Die Dynamik des Reglers ist in der Fehlerübertragungsfunktion $H(s)$ berücksichtigt, so dass sich keine Probleme durch die Verwendung des Signals $e = \Omega - w'$ als Fehlersignal des GRNN und als Eingangssignal des Reglers ergeben. Die Details zur Realisierung des Lerngesetzes und des GRNN sind in Bild 4.7 dargestellt (siehe dazu auch Gleichung (4.21)).

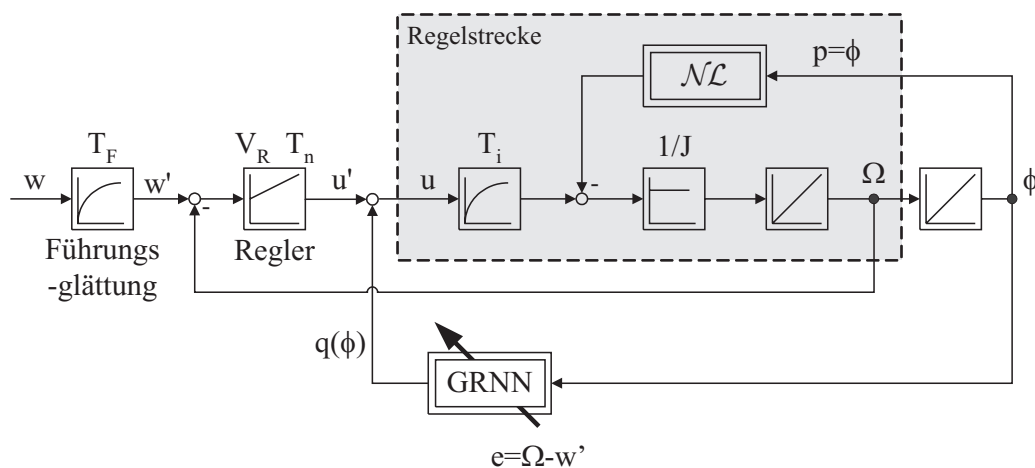


Bild 4.6: Kompensationsstruktur mit selbstlernendem Regler (GRNN)

Die Daten des verwendeten Simulationsmodells sind in Tabelle 4.1 angegeben.

V_R	$= 16,5 \text{ Nms/rad}$	T_n	$= 300 \text{ ms}$
J	$= 0,496 \text{ kgm}^2$	T_i	$= 3 \text{ ms}$
M_{gegen}	$= 10 \text{ Nm}$	w	$= 5 \text{ rad/s}$
η	$= 4000$	σ_{norm}	$= 1$

Tabelle 4.1: Daten des Einmassensystems mit Nichtlinearität und PI-Regler

Bild 4.8 zeigt den Drehzahlverlauf ohne aktivierte Kompensation durch das GRNN. Es ergeben sich Drehzahlschwankungen mit einer Amplitude von 14% des Sollwertes. Man erkennt deutlich, dass der PI-Regler die schnell auftretende Störung nur sehr unbefriedigend

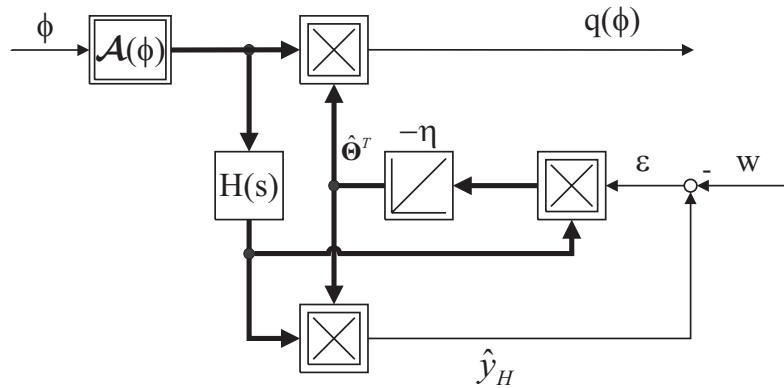


Bild 4.7: Details zur Realisierung des Lerngesetzes und des GRNN

ausregeln kann. Der selbstlernende Regler, kombiniert mit dem vorhandenen PI-Regler, lie-

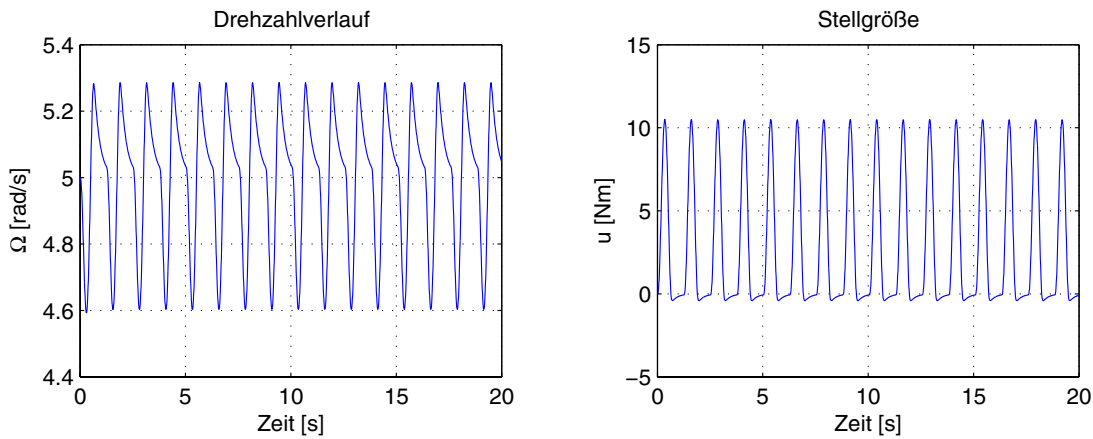


Bild 4.8: Drehzahlverlauf und Stellgröße ohne Kompensation

fert nach einer Lernzeit von 20 s das Ergebnis aus Bild 4.9. Aufgrund der Periodizität der Störung kann das GRNN die Information über die Störung in den Netzwerkgewichten speichern und das entsprechende Kompensationssignal $q(\phi)$ zum richtigen Zeitpunkt erzeugen. Die Störung kann bis auf einen relativen Fehler von etwa 0,2% unterdrückt werden. Der gesamte Lernvorgang ist in Bild 4.10 dargestellt. Die Lerngeschwindigkeit kann durch die Wahl der Lernschrittweite η eingestellt werden. Nach etwa 20 s ist das Kompensationssignal optimal erlernt und eine weitere Verbesserung findet nicht mehr statt.

Nach erfolgtem Lernvorgang liefert das GRNN eine statische Kennlinie, die das erforderliche Kompensationssignal $q(\phi)$ wiedergibt. Bild 4.11 links zeigt das erlernte Kompensationssignal $q(\phi)$ nach einer Lerndauer von 40 s. In Bild 4.11 rechts ist die zugehörige wirkende Nichtlinearität $\mathcal{NL}(\phi)$ dargestellt. Es fällt auf, dass die Amplitude, Form und Phasenlage von $q(\phi)$ und $\mathcal{NL}(\phi)$ gleich sind, d.h. die dynamische Beeinflussung von $q(\phi)$ durch die Fehlerübertragungsfunktion $H(s)$ ist bei der aktuellen Grundfrequenz von $f_0 = 5 \text{ rad/s}/2\pi = 0.8 \text{ Hz}$ nicht merklich. Zudem ist ein Offset von ca. 2 Nm zwischen Nichtlinearität und Kompensationssignal erkennbar. Dies rührt daher, dass sowohl das GRNN als auch der

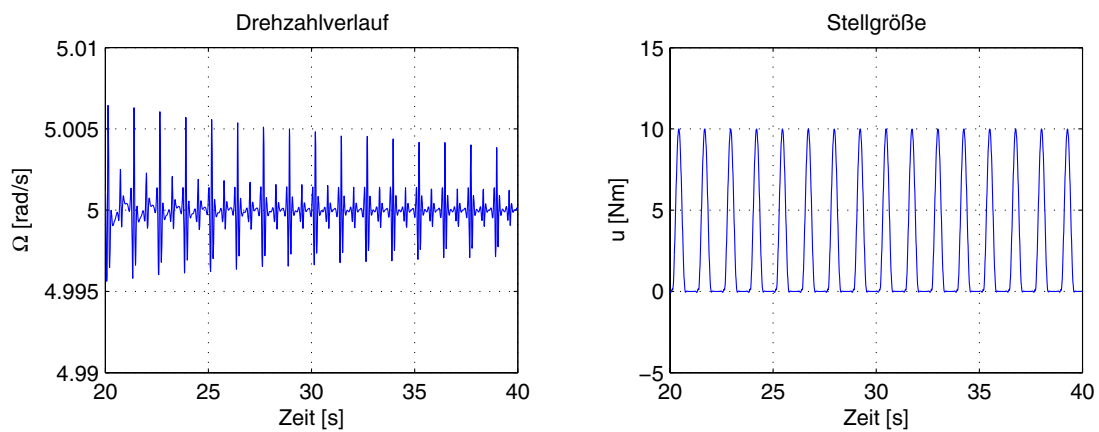


Bild 4.9: Drehzahlverlauf und Stellgröße mit Kompensation nach einer Lernzeit von 20 s

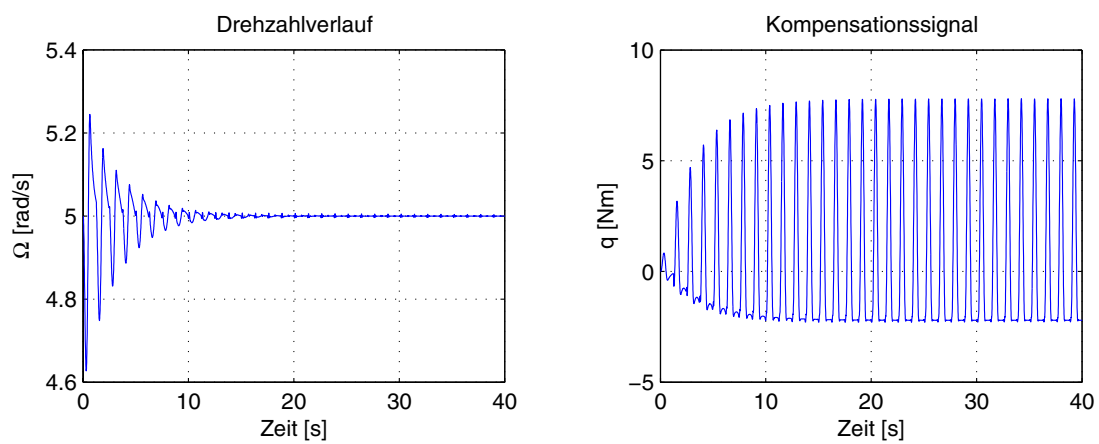


Bild 4.10: Drehzahlverlauf und Kompensationssignal q während des Lernvorgangs

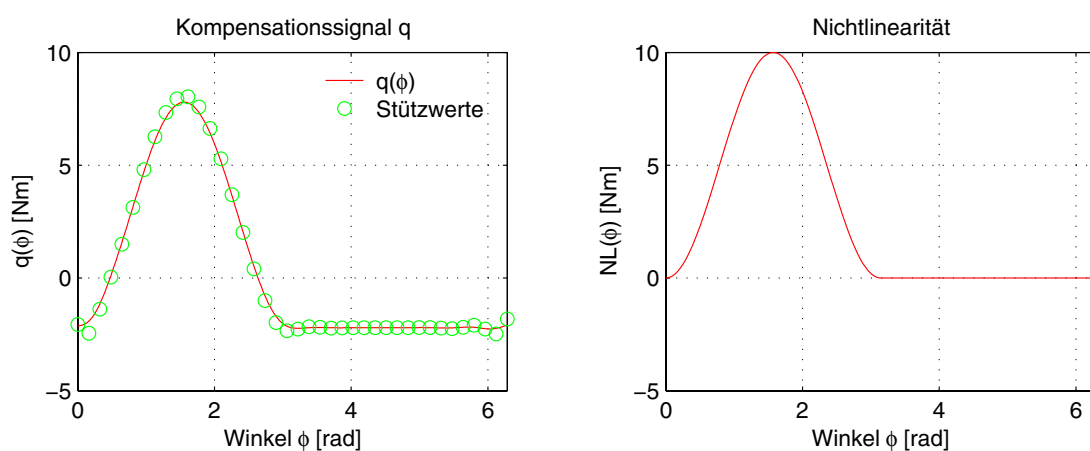


Bild 4.11: Kompensationssignal $q(\varphi)$ und Nichtlinearität $\mathcal{NL}(\phi)$

Integralanteil des PI-Reglers einen konstanten Ausgang erzeugen können. Es ist damit ein Freiheitsgrad zu viel vorhanden, der sich beliebig auf PI-Regler und GRNN aufteilen kann. Zu Veranschaulichung ist das Ausgangssignal u' des PI-Reglers und das Kompensationssignal q in Bild 4.12 dargestellt. Während sich der Mittelwert von q nach unten verschiebt, wächst das Signal u' um den selben Betrag an. Um zu verhindern, dass eine beliebige

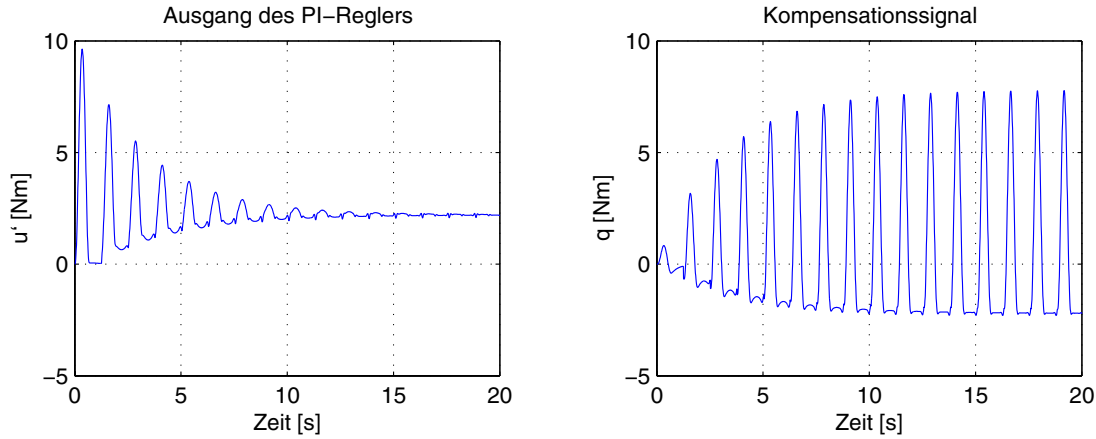


Bild 4.12: Vergleich von u' und q

Aufteilung des Offset des Kompensationssignals stattfindet, muss das GRNN ein mittelwertfreies Signal liefern. Dies kann dadurch erreicht werden, dass zu jedem Abtastschritt der Mittelwert über alle Stützwerte von allen Stützwerten subtrahiert wird.

4.3.2 Kompensation bei Arbeitspunktwechseln

Die Problematik des Arbeitspunktwechsels und der damit verbundenen Veränderung der Periodendauer T der periodischen Nichtlinearität ist in Bild 4.13 dargestellt. Nach einem Drehzahlsprung muss das GRNN nachtrainiert werden, da die Übertragungsfunktion $G_1^{-1}(s)$ in Gleichung 4.17 jetzt bei anderen Frequenzen ausgewertet wird. Die Notwendigkeit des Nachlernens bei einem Arbeitspunktwechsel kann durch eine zusätzliche Eingangsdimension im GRNN vermindert werden. Das Kompensationssignal wird für verschiedene Drehzahlen Ω und damit auch für verschiedene Periodendauern T gelernt. Bei einem erneuten Arbeitspunktwechsel kann dann auf das bereits Gelernte zurückgegriffen werden und der einzige Bedarf des Nachlernens entsteht durch die Übergangsvorgänge zwischen den Arbeitspunkten. Während des Übergangs entsteht ein Regelfehler, der eigentlich nur den PI-Regler zu Stellaktionen veranlassen sollte. Da aber gemäß Gleichung (4.26) das selbe Fehlersignal auch zur Adaption des GRNN verwendet wird, werden dadurch die Stützwerte "verstellt". Um dies zu verhindern, kann die Adaption des GRNN während des Übergangs gestoppt werden. In Bild 4.14 links ist der Drehzahlverlauf bei Verwendung eines zweidimensionalen GRNN dargestellt. Die Eingangssignale des GRNN sind die Lage ϕ und die Drehzahl Ω . Je öfter ein Arbeitspunkt angefahren wurde, desto geringer sind die anfänglichen Drehzahlschwankungen. Bild 4.14 rechts zeigt das zweidimensionale Lernergebnis des GRNN nach einer Dauer von 240 s. Man erkennt deutlich, dass nur entlang

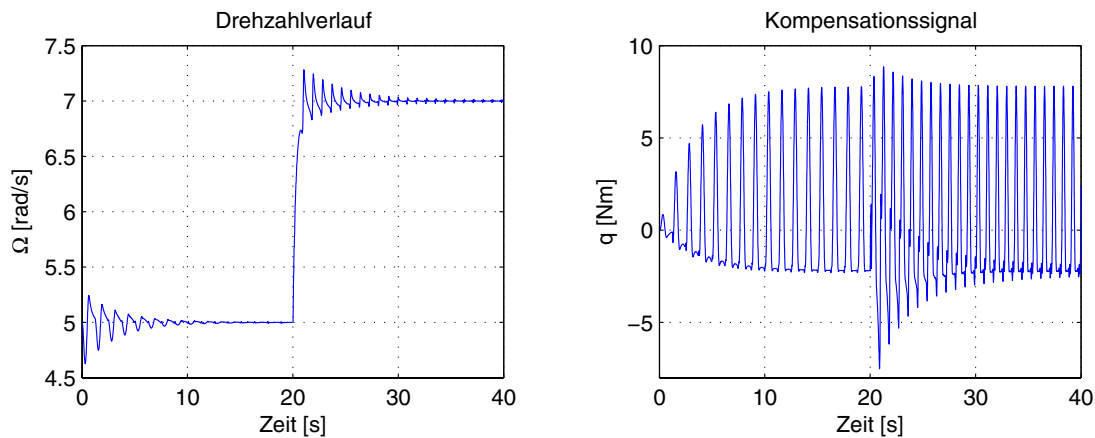


Bild 4.13: Lernvorgang bei einem Drehzahlsprung

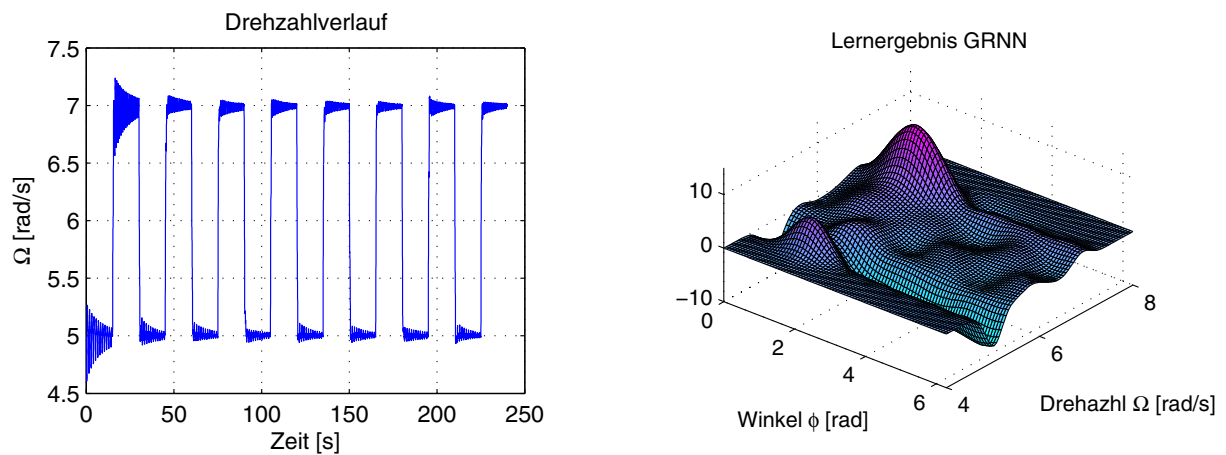


Bild 4.14: Lernvorgang und -ergebnis mit zweidimensionalem GRNN nach mehreren Drehzahlwechseln

der Achsen $\Omega = 5 \text{ rad/s}$ und $\Omega = 7 \text{ rad/s}$ nennenswert gelernt wurde. In den Drehzahlbereichen dazwischen wurde wegen der schnellen Arbeitspunktwechsel kein Lernergebnis erzielt.

Aufgrund der bereits kurzen Lerndauern mit einem eindimensionalen GRNN ist diese Möglichkeit allerdings eher theoretischer Natur, da dem geringeren Nachlernbedarf ein wesentlich höherer Rechenaufwand durch das zweidimensionale neuronale Netz gegenübersteht. Außerdem müsste der gesamte relevante Bereich in einem ausreichend feinen Raster durchfahren werden.

Das dargestellte Beispiel zeigt, dass der selbstlernende Regler sehr gut in der Lage ist periodische Störungen (Nichtlinearitäten) im Ausgangssignal zu kompensieren. Anwendungen sind im Bereich von hochgenauen Drehzahl- und Lageregelungen oder bei der aktiven Schwingungsdämpfung zu sehen.

4.4 Unrundheitskompensation in kontinuierlichen Fertigungsanlagen

Bei der Produktion und Bearbeitung von Papier, Kunststoffolie oder Stahl kommen kontinuierliche Fertigungsanlagen zum Einsatz. In diesem Kapitel wird eine Papierbearbeitungsmaschine bestehend aus einem Abwickler, drei Bearbeitungsstationen und einem Aufwickler gemäß Bild 4.15 betrachtet. Diese ca. 9 m lange Modellarbeitsmaschine

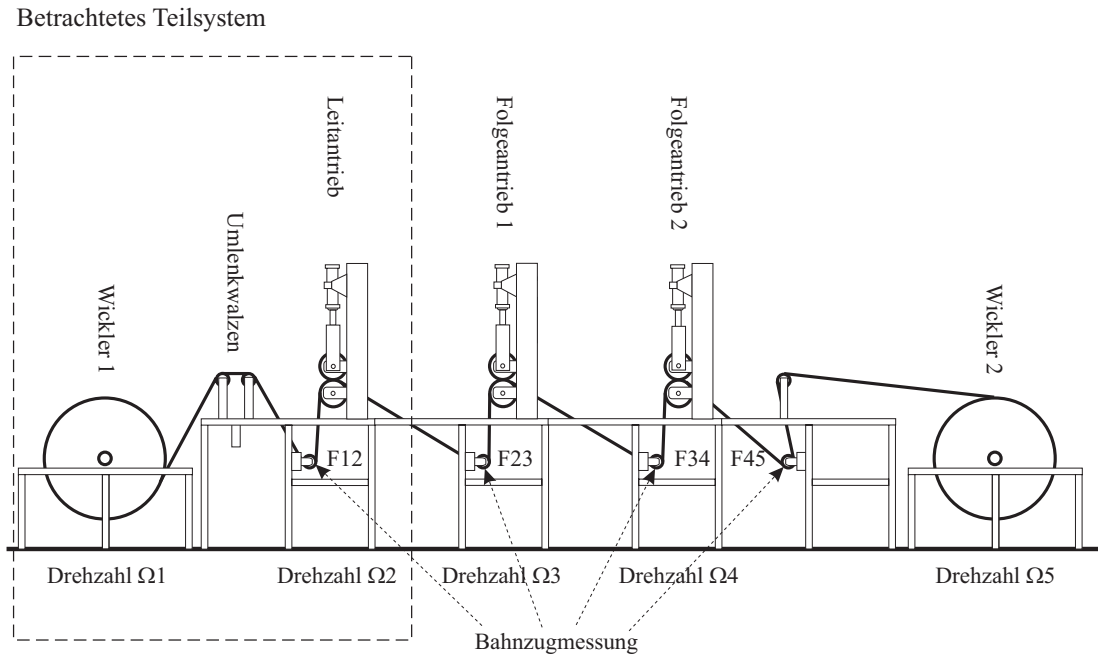


Bild 4.15: Modellarbeitsmaschine MODAM bestehend aus 5 Antrieben

(MODAM) steht am Lehrstuhl für Elektrische Antriebssysteme zu Experimentierzwecken zur Verfügung. Eine genaue Beschreibung der Anlage einschließlich der Modellierung der Teilkomponenten findet sich in [59]. Die Regelung und Bedienung der Anlage läuft unter dem Echtzeitsystem dSpace [53] ab und wird unter Matlab/Simulink entwickelt (siehe Anhang B).

4.4.1 Modellierung

Die Anlage besteht aus zwei Massespeichern (Ab- und Aufwickler), einem Leittrieb und zwei Folgeantrieben. Der Leittrieb prägt dem System die Leitgeschwindigkeit v_2 ein, wobei die Folgeantriebe den Bearbeitungsstationen (Bedrucken etc.) einer industriell eingesetzten Maschine entsprechen. Charakteristisch für derartige Anlagen ist die gegenseitige Kopplung der einzelnen Teilsysteme durch das transportierte Material.

Als Messdaten stehen dem Benutzer die Bahnkräfte F_{ij} und die Motordrehzahlen Ω_j der einzelnen Antriebe zur Verfügung. Ferner sind die jeweiligen Sollmomente M_{soli} der Momentenregelkreise verfügbar. Die Mittelwerte R_m der Radien der beiden Wickler werden im

Online-Betrieb über die Geschwindigkeiten und Drehzahlen ermittelt. Dieser Wert kann zur Berechnung des variablen Trägheitsmoments und demnach auch zur Adaption relevanter Reglerparameter (Drehzahlregelung der Wickler) herangezogen werden. Eine Zusammenfassung der technischen Daten der MODAM findet sich in Anhang B.

Die durchgeführten Untersuchungen zur Unrundheitskompensation beschränkten sich auf den Wickler 1 (siehe Markierung in Bild 4.15). Dieses Teilsystem erstreckt sich vom Wickler selbst, über zwei Umlenkrollen, einer Bahnkraftmesseinrichtung (F_{12}) bis hin zum Leitantrieb. Da angenommen wird, dass der Leitantrieb dem System die Leitgeschwindigkeit v_2 einprägt und diese indirekt aus der (messbaren) Motordrehzahl am Leitantrieb und dem Radius der Antriebswalze ermittelt werden kann, lässt sich das Teilsystem vom Rest der MODAM entkoppeln. Diese Annahme ist nur gültig, wenn für den Antriebsstrang im Leitantrieb ein ausreichend steifes System vorausgesetzt wird. Die Antriebe 1, 3, 4 und 5 sind über Kaskadenregler kraftgeregelt, d.h. deren Aufgabe ist eine konstante Bahnkraft in den einzelnen Abschnitten einzuhalten. Die resultierenden, zu betrachtende Teilsysteme zeigt

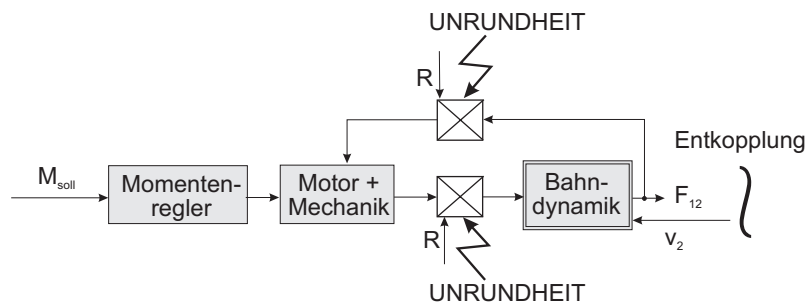


Bild 4.16: Betrachtete Teilsysteme des Wicklersystems: Antriebsstrang und Bahndynamik

Bild 4.16. Eingangsgrößen in das System sind das vom Drehzahlregler berechnete Sollmoment M_{soll} und die durch den Leitantrieb eingeprägte Leitgeschwindigkeit v_2 . Als messbare Zustandsgrößen stehen die Motordrehzahl Ω_1 und die Bahnkraft F_{12} zur Verfügung. Es werden nur Abwickelvorgänge mit $v_2 \geq 0$ betrachtet.

Als periodische Nichtlinearität wird eine Unrundheit am Abwickler (Wickler 1) bei konstanter Bahngeschwindigkeit näher betrachtet. Diese führt zu periodischen Schwankungen der Abwickelgeschwindigkeit der Papierbahn und über die Kontinuitätsgleichung [59] zu erheblichen Kraftschwankungen. Unrundheiten können durch unsachgemäße Lagerung und unsauberes Aufwickeln entstehen. Bei den durchgeführten Messungen wurde eine Schaumstoffmatte in den Abwickler eingewickelt, so dass eine Unrundheit mit einer Amplitude von ca. 8 mm entsteht. Bild 4.17 zeigt den Abwickler mit eingewickeltem Schaumstoff.

Antriebssystem

Der Antrieb des Wickels an der MODAM erfolgt über eine Asynchronmaschine, die über eine Welle mit dem Getriebe verbunden ist. Eine weitere Welle treibt den Aufnehmerdorn für den Papierwickel an (Bild 4.18). Prinzipiell handelt es sich bei diesem System um einen Dreimassenschwinger mit den beiden Federn vor und nach dem Getriebe, der Massenträgheitsmomente des Motors und des Wickels sowie der Getriebemasse. Die Schwierigkeit besteht aber darin, dass die Parameter c_{f12} und c_{f23} nicht genau bekannt oder bestimmbar sind und andererseits nur das gesamte Trägheitsmoment J_1 (ohne Wickelgut) an der

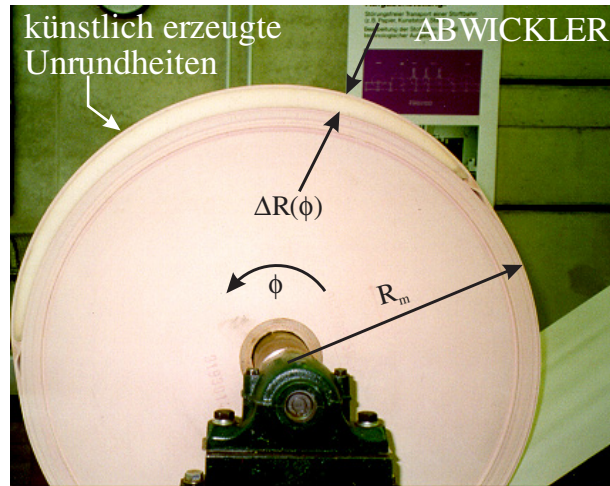


Bild 4.17: Unrundheit am Abwickler durch eingewickelten Schaumstoff

MODAM bekannt ist. Es gilt also für J_1 gemäß den Zuordnungen in Bild 4.18:

$$J_1 = J_1^* + J_{Welle1} + J_{Getr} + \frac{1}{\ddot{u}^2} (J_{welle2} + J_{Dorn}) \quad (4.28)$$

Eine Vereinfachung liefert die Modellierung des Antriebsstranges als **Zweimassensystem**. Dies garantiert nach [56] eine Modellierung der wesentlichen Eigenschaften des Antriebssystems. Die Trägheit des Papierwickels berechnet sich nach [28] zu:

$$J_2^* = \frac{1}{2} \pi B \rho (R^4(t) - R_{min}^4) \quad (4.29)$$

Durch die vierte Potenz von $R(t)$ wird klar, dass dieser Parameter während des Betriebes starken Variationen unterliegt.

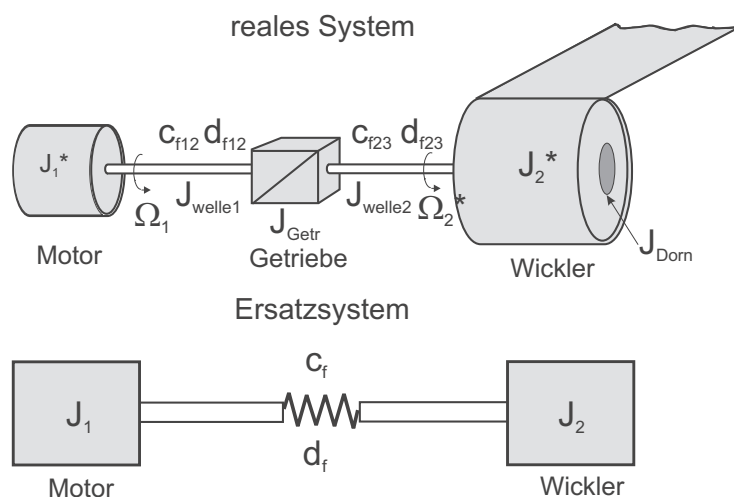


Bild 4.18: Kopplung zwischen Motor und Welle

Neben den Trägheitsmomenten ist die Federsteifigkeit c_f und die Dämpfungskonstante d_f des Ersatzsystems von Bedeutung. Diese Größen wurden im Experiment in [56] ermittelt,

da sich gezeigt hat, dass eine Berechnung aufgrund den vielen unbekannten Verbindungen nur unbefriedigende Ergebnisse ergibt [56]. Alle Größen werden nun auf den Motor bezogen.

$$J_2 = \frac{1}{\ddot{u}^2} \cdot J_2^* \quad (4.30)$$

$$\Omega_2 = \ddot{u} \cdot \Omega_2^* \quad (4.31)$$

Ferner wird der **Momentenregelkreis** als PT₁-Glieder mit einer Ersatzzeitkonstanten von $T_i = 3 \text{ ms}$ modelliert (Herstellerangaben). Das Gegenmoment des Antriebssystems stellt die über den Radius umgerechnete Bahnkraft F_{12} dar. Dies ist gleichzeitig die Kopplung des Bahn- mit dem Antriebssystem (Bild 4.16).

$$M_{\text{gegen}} = \frac{1}{\ddot{u}} \cdot F_{12} \cdot R \quad (4.32)$$

Bahndynamik

Unter dem Bahnsystem wird im Folgenden der Teil der Anlage vom Abwickelpunkt des Wicklers bis zum Leitantrieb verstanden. Die Kopplung vom Antriebssystem in das Bahnsystem erfolgt über die Abwickelgeschwindigkeit v_{ab} . Diese ergibt sich unter Berücksichtigung des Radius und des Getriebes mit der auf den Motor bezogenen Wicklerdrehzahl Ω_2 zu:

$$v_{ab} = \frac{R}{\ddot{u}} \cdot \Omega_2 \quad (4.33)$$

Das dynamische Verhalten der Bahn lässt sich nach [37, 59, 42] mit der **Kontinuitätsgleichung** beschreiben.

$$\frac{d}{dt} \left(\frac{L_n}{1 + \epsilon_{12}} \right) = \frac{v_{ab}}{1 + \epsilon_{01}} - \frac{v_2}{1 + \epsilon_{12}} \quad (4.34)$$

Dabei ist L_n die Bahnlänge zwischen dem Wickler und dem Leitantrieb, ϵ_{01} die im Wickler gespeicherte Dehnung des Materials (Vorgeschichte des Wicklers) und ϵ_{12} die Dehnung des gerade abgewickelten Materials. Die physikalische Aussage der Gleichung (4.34) ist, dass eine Differenz zwischen dem zu- und abgeführten Massenstrom im betrachteten Bahnabschnitt durch eine zeitliche Änderung der Dehnung ϵ_{12} oder der Bahnlänge L_n gespeichert wird. Die Bahnkraft F_{12} ergibt sich aus der Dehnung ϵ_{12} und der Nenndehnung ϵ_n zu:

$$F_{12} = \frac{\epsilon_{12}}{\epsilon_n} \quad \text{mit} \quad \epsilon_n = \frac{1}{E \cdot B \cdot H} \quad (4.35)$$

Dabei entspricht E dem Elastizitätsmodul des verwendeten Materials, B der Breite und H der Höhe des Materials (siehe Anhang B). Durch Linearisierung der Gleichung (4.34) um die Arbeitspunktgeschwindigkeit v_0 erhält man folgenden Zusammenhang:

$$L_n \frac{d}{dt} \Delta \epsilon_{12} = -\Delta v_{ab} + \Delta v_2 + v_0 (\Delta \epsilon_{01} + \Delta \epsilon_{12}) \quad (4.36)$$

Durch Übergang auf die Großsignalgrößen (weglassen von “ Δ ”) und mit der Abkürzung

$$v^* = v_2 - v_{ab} + v_0 \epsilon_{01} \quad (4.37)$$

ergibt sich die Übertragungsfunktion der Bahndynamik zu:

$$\frac{\epsilon_{12}(s)}{v^*(s)} = \frac{1}{v_0} \frac{1}{1 + T_n s} \quad \text{mit} \quad T_n = \frac{L_n}{v_0} \quad (4.38)$$

Das um eine Arbeitsgeschwindigkeit linearisierte Bahnsystem stellt ein PT_1 -Glied mit geschwindigkeitsabhängigem Verstärkungsfaktor und Zeitkonstante dar. Die auf das Bahnsystem wirkende Nichtlinearität stellt die Abweichung des Radius durch die Unrundheit von seinem Mittelwert R_m dar.

$$\mathcal{NL}(\phi) = \Delta R(\phi) = R(\phi) - R_m \quad (4.39)$$

Gesamtmodell

Bild 4.19 zeigt den Signalflussplan des Gesamtmodells bestehend aus Antriebssystem und Bahndynamik. Das Antriebssystem wurde als Zweimassensystem gemäß Anhang A und [29] modelliert. Als Arbeitspunktgeschwindigkeit v_0 für die Linearisierung der Kontinuitätsgleichung wurde die aktuelle Bahngeschwindigkeit v_2 gewählt.

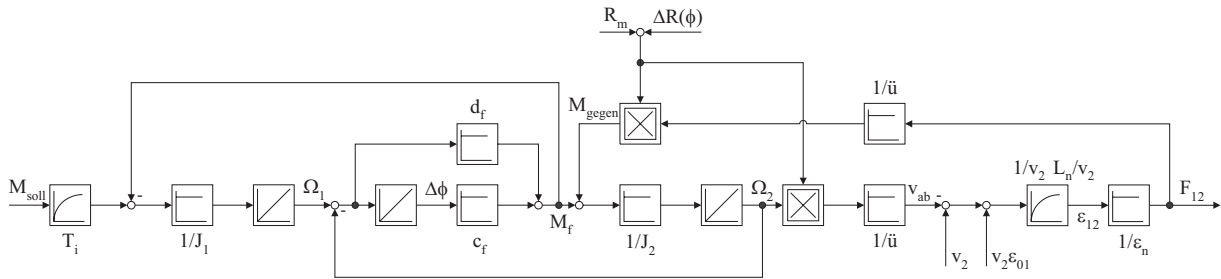


Bild 4.19: Signalflussplan des betrachteten Teilsystems der Modellarbeitsmaschine

Die Regelung des Wicklers erfolgt in Kaskadenstruktur. Der innere Regelkreis umfasst eine Drehzahlregelung der Antriebsmaschine mit der Messgröße Ω_1 . Hierfür wird ein PI-Regler eingesetzt. Die äußere Schleife ist ein PI-Kraftregler, der die gemessene Bahnkraft F_{12} verwendet. Die Übertragungsfunktionen R_N für den Drehzahlregler und R_F für den Kraftregler sind in Gleichung (4.40) angegeben.

$$R_\Omega(s) = V_{R,\Omega} \cdot \frac{1 + T_{n,\Omega} \cdot s}{T_{n,\Omega} \cdot s} \quad R_F(s) = V_{R,F} \cdot \frac{1 + T_{n,F} \cdot s}{T_{n,F} \cdot s} \quad (4.40)$$

Die Reglerparameter wurden bei der Projektierung der Anlage näherungsweise nach dem symmetrischen Optimum [29] eingestellt. Um Überspringen zu vermeiden, wird der Drehzahlsollwert durch eine Führungsglättung mit der Zeitkonstanten T_Ω geglättet. Eine Optimierung der bestehenden Reglereinstellungen wird bewusst nicht durchgeführt, da gezeigt werden soll, dass der selbstlernende Regler sehr gut geeignet ist einen bestehenden Regler ohne Modifikationen bei der Kompensation periodisch wirkender Nichtlinearitäten zu unterstützen. Um bei Beschleunigungs- und Bremsvorgängen Einbrüche in der Bahnkraft zu vermeiden, wird dem Kraftregler eine Vorsteuerdrehzahl und dem Drehzahlregler ein Vorsteuermoment auf der Basis der Sollgeschwindigkeit und der Sollbeschleunigung zur

Verfügung gestellt. Das Vorsteuermoment M_V setzt sich aus dem geforderten Beschleunigungsmoment M_B des Wickels 1 und der Antriebsmaschine, sowie dem aufzubringenden Moment M_F , bedingt durch die Bahnzugkraft zusammen.

$$M_V = M_B + M_F = (J_1 + J_2) \cdot \dot{v}_{2soll} - \frac{R_m \cdot F_{soll}}{\ddot{t}} \quad (4.41)$$

Dabei wird der mittlere Radius R_m aus dem Drehzahlverhältnis des Wickelmotors und des Leitantriebs gebildet. Die Größe $\dot{v}_{2soll}$ stellt die Sollbeschleunigung der gesamten Anlage dar und wird aus dem Geschwindigkeitssollwert gebildet. Da in Kap. 4.4.2 alle Messungen bei $v_2 = const.$ stattfinden, ist der entsprechende Term jeweils null. Die Vorsteuerdrehzahl Ω_V ergibt sich aus dem momentanen Geschwindigkeitssollwert der Papierbahn nach Gleichung (4.42).

$$\Omega_V = \frac{v_{2soll} \cdot \ddot{u}}{R_m} \quad (4.42)$$

Der Kraft- und Drehzahlregler muss jeweils nur die von der Vorsteuerung nicht erfassten Effekte (z.B. Reibung, ungenaue Trägheitsmomente) ausregeln. Einen Überblick über die Regelungsstruktur einschließlich Vorsteuersignale und den Angriffspunkt des Kompensationssignals für den selbstlernenden Regler gibt Bild 4.20. Die Daten der Kaskadenregelung

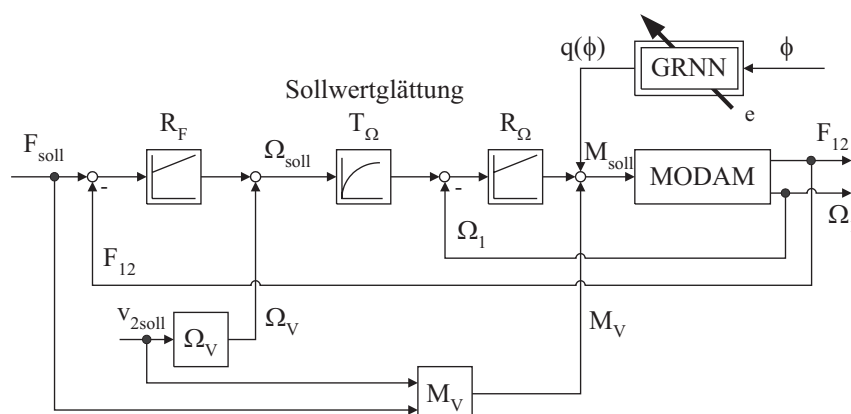


Bild 4.20: Überblick: Regelungs- und Kompensationsstruktur der MODAM

sind in Tabelle 4.2 aufgelistet. Die Parameter des Antriebs- und Bahnsystems finden sich in Anhang B. Die in Kap. 4.4.2 dargestellten Messergebnisse wurden bei einem mittleren Wickelradius von $R_m = 0.32 \text{ m}$ durchgeführt. Der Verstärkungsfaktor $V_{R,\Omega}$ wird abhängig vom Radius R_m bzw. vom Trägheitsmoment J_2 nachgeführt. Der in Tabelle 4.2 aufgeführte Wert für $V_{R,\Omega}$ bezieht sich auf den Radius $R_m = 0.32 \text{ m}$.

Die Unrundheit ΔR hängt von der Winkelstellung ϕ des Wickels ab. Diese ist näherungsweise durch den Inkrementalgeber an der Asynchronmaschine von Wickel 1 messbar (wegen der elastischen Verbindung zwischen Asynchronmaschine und Wickel ist keine exakte Messung möglich). Wenn die Anlage mit konstanter Geschwindigkeit betrieben wird, handelt es sich um eine periodische Störung mit der Grundfrequenz $\omega_0 = \Omega_2/\ddot{u}$, bzw. es gilt $\Delta R(\phi) = \Delta R(\phi + 2\pi)$. In den Messungen aus Kap. 4.4.2 wurde eine Unrundheit gemäß Bild 4.17 verwendet.

$R_m = 0.32 \text{ m}$	mittlerer Wickelradius
$\Delta R \approx 8 \text{ mm}$	Spitzenwert der Unrundheit
$v_2 = 2 \dots 3 \text{ m/s}$	Arbeitsgeschwindigkeit der Papierbahn
$L_n = 3 \text{ m}$	freie Bahnlänge von Wickler 1 bis zum Antrieb 2
$V_{R,\Omega} = 5.36 \text{ Nms/rad}$	P-Verstärkung Drehzahlregler
$T_{n,\Omega} = 0.4 \text{ s}$	Integrationszeit Drehzahlregler
$V_{R,F} = -0.0137 \text{ rad/Ns}$	P-Verstärkung Kraftregler
$T_{n,F} = 1.4 \text{ s}$	Integrationszeit Kraftregler
$T_\Omega = 0.15 \text{ s}$	Sollwertglättung Drehzahlregler

Tabelle 4.2: *Reglereinstellungen an der MODAM*

4.4.2 Unrundheitskompensation durch den selbstlernenden Regler

Zur Durchführung der selbstlernenden Kompensation muss nun noch entschieden werden, welcher Eingriffspunkt für das Kompensationssignal gewählt wird. Es kommt prinzipiell ein zusätzlicher Kraftsollwert, Drehzahlsollwert und Momentensollwert in Betracht. Da es wegen der ausgeprägten Kaskadenstruktur am günstigsten ist, möglichst weit innen in der Kaskade anzugreifen, wird zur Kompensation ein zusätzlicher Momentensollwert verwendet (siehe Bild 4.20).

Zur Vereinfachung kann die Kopplung der Unrundheit ΔR auf das Antriebssystem vernachlässigt werden, da diese isoliert betrachtete Auswirkung auf die Bahnkraft viel geringer ist als die Vorwärtskopplung in das Bahnsystem auf die Abwickelgeschwindigkeit v_{ab} (Bild 4.19). Auf das Antriebssystem wirkt im Modell daher nur der mittlere Radius R_m .

Zur Berechnung des Lerngesetzes für das GRNN des selbstlernenden Reglers benötigt man die Fehlerübertragungsfunktion $H(s)$, die die Dynamik vom Eingriffspunkt des Kompensationssignals bis zum Systemausgang F_{12} beschreibt. Dies geschieht in zwei Schritten. Zuerst wird das Zustandsgleichungssystem des geregelten Abwicklers aufgestellt, woraus dann sehr einfach die Fehlerübertragungsfunktion errechnet werden kann. Zur Aufstellung der Zustandsdarstellung wird der durch ein PT₁-Glieder angenäherte Momentenreglerkreis vernachlässigt. Die Zeitkonstante ist im Vergleich zu den restlichen dominanten Zeitkonstanten vernachlässigbar klein. Das Zustandsgleichungssystem wird aus den Signalflussplänen der Bilder 4.19 und 4.20 aufgestellt.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A} \cdot \mathbf{x} + \mathbf{B} \cdot \mathbf{u} + \mathbf{b}_S \\ y &= \mathbf{c}^T \cdot \mathbf{x}\end{aligned}\tag{4.43}$$

Die zugehörigen Vektoren und Matrizen lauten:

$$\mathbf{x}^T = \begin{bmatrix} \Omega_1 & \Delta\phi & \Omega_2 & \epsilon_{12} & x_\Omega & x_F & x_{T\Omega} \end{bmatrix}\tag{4.44}$$

$$\mathbf{u}^T = \begin{bmatrix} F_{soll} & \Omega_2 \cdot \Delta R(\phi) & q(\phi) \end{bmatrix}\tag{4.45}$$

$$\mathbf{c}^T = \begin{bmatrix} 0 & 0 & 0 & 1/\epsilon_n & 0 & 0 & 0 \end{bmatrix}\tag{4.46}$$

$$\mathbf{A} = \begin{bmatrix} \frac{-d_f - V_{R,\Omega}}{J_1} & -\frac{c_f}{J_1} & \frac{d_f}{J_1} & -\frac{V_{R,\Omega} \cdot V_{R,F}}{J_1 \cdot \epsilon_n} & \frac{1}{J_1} & 0 & \frac{V_{R,\Omega}}{J_1} \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ \frac{d_f}{J_2} & \frac{c_f}{J_2} & -\frac{d_f}{J_2} & \frac{R_m}{\ddot{u} \cdot J_2 \cdot \epsilon_n} & 0 & 0 & 0 \\ 0 & 0 & -\frac{R_m}{L_n \cdot \ddot{u}} & -\frac{v_2}{L_n} & 0 & 0 & 0 \\ -\frac{V_{R,\Omega}}{T_{n,\Omega}} & 0 & 0 & 0 & 0 & 0 & \frac{V_{R,\Omega}}{T_{n,\Omega}} \\ 0 & 0 & 0 & -\frac{V_{R,F}}{T_{n,F} \cdot \epsilon_n} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{V_{R,F}}{\epsilon_n \cdot T_\Omega} & 0 & \frac{1}{T_\Omega} & -\frac{1}{T_\Omega} \end{bmatrix} \quad (4.47)$$

$$\mathbf{B} = \begin{bmatrix} -\frac{R_m}{J_1 \cdot \ddot{u}} & 0 & \frac{1}{J_1} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -\frac{1}{L_n \cdot \ddot{u}} & 0 \\ \frac{V_{R,\Omega} \cdot V_{R,F}}{T_{n,\Omega}} & 0 & 0 \\ \frac{V_{R,F}}{T_{n,F}} & 0 & 0 \\ \frac{V_{R,F}}{T_\Omega} & 0 & 0 \end{bmatrix} \quad (4.48)$$

$$\mathbf{b}_S^T = \begin{bmatrix} \frac{\dot{v}_2 \cdot (J_1 + J_2)}{J_1} & 0 & 0 & \frac{v_2 \cdot (1 + \epsilon_{01})}{L_n} & 0 & 0 & \frac{v_2 \cdot \ddot{u}}{T_\Omega \cdot R_m} \end{bmatrix} \quad (4.49)$$

x_Ω , x_F und x_{T_Ω} sind die Zustände des Drehzahl- und Kraftregelkreises bzw. der Drehzahlsollwertglättung mit der Zeitkonstanten T_Ω . Das Kompensationssignal $q(\phi)$ ist der Übersichtlichkeit halber in den Eingangsvektor $\mathbf{u}$ integriert. Die nichtlineare Störgröße $\Delta R(\phi)$ ist in der zweiten Komponente von $\mathbf{u}$ enthalten. Sie tritt jedoch im Produkt mit der Wicklerdrehzahl Ω_2 auf. Die gesamte angreifende Nichtlinearität $\mathcal{NL}$ ist somit zunächst von zwei Eingangsgrößen abhängig.

$$\mathcal{NL}(\Omega_2, \phi) = \Omega_2 \cdot \Delta R(\phi) \quad (4.50)$$

Man könnte jetzt ein zweidimensionales GRNN mit den Eingangsgrößen Ω_2 und ϕ entwerfen. Da jedoch vorausgesetzt wurde, dass die Anlage mit konstanter Geschwindigkeit betrieben wird, ändert sich die Wicklerdrehzahl nur in dem Maße, wie sich der mittlere Radius R_m durch Zu- oder Abnahme der aufgewickelten Papiermenge ändert. Diese Änderung geht sehr langsam von statten, so dass näherungsweise von einer konstanten Wicklerdrehzahl Ω_2 ausgegangen werden kann. Diese langsamen Änderungen von Ω_2 durch Radiusänderungen können durch das GRNN durch Nachlernen ohne Fehler in der Regelgröße ausgeglichen werden. Dadurch wird die Fähigkeit des Online-Adaptionsverfahrens mittels GRNN, sich an langsam zeitveränderliche Nichtlinearitäten anzupassen, deutlich. Die Nichtlinearität kann demnach als nur von ϕ abhängig betrachtet werden, jedoch ist sie langsam zeitvariant.

$$\mathcal{NL} = \mathcal{NL}(\phi) = \Omega_2 \cdot \Delta R(\phi) \quad \text{mit} \quad \Omega_2 \approx \text{const.} \quad (4.51)$$

Dieser Weg wurde bei den Messreihen gewählt, um einen mehrdimensionalen Eingangsraum des GRNN zu vermeiden. Das GRNN zur Adaption des Kompensationssignals $q(\phi)$ wird gemäß Gleichung (4.52) angesetzt.

$$q(\phi) = \hat{\Theta}^T \cdot \mathcal{A}(\phi) \quad (4.52)$$

Bei einem Arbeitspunktwechsel durch Veränderung der Bahngeschwindigkeit ergibt sich das bereits in Abschnitt 4.2.3 erwähnte Problem, dass ein Nachlernen des GRNN erforderlich ist. Dabei können kurzzeitig Fehler in der Regelgröße auftreten, d.h. die Unrundheit wird nicht vollständig kompensiert.

Die autonome Lösung der Differentialgleichung bedingt durch den konstanten Eingangsvektor $\mathbf{b}_s$ wird zur Berechnung der Fehlerübertragungsfunktion $H(s)$ nicht benötigt und daher nicht weiter betrachtet. Die zweite Komponente des Eingangsvektors enthält den internen Zustand Ω_2 . Dadurch ergeben sich Multiplikationen von Zustandsgrößen und somit ein nichtlineares System, für das eine Übertragungsfunktion nicht existiert. Wegen $\Omega_2 \approx \text{const.}$ wird Ω_2 im Eingangsvektor (und nur dort!) als arbeitspunktabhängiger Parameter aufgefasst und durch die gemessene Motordrehzahl Ω_1 angenähert. Durch diese Näherung erhält man ein lineares System zur Berechnung der Fehlerübertragungsfunktion $H(s)$. Wie sich aus Gleichung (4.19) ergeben hat, ist die Fehlerübertragungsfunktion $H(s)$ die Übertragungsfunktion vom Eingriffspunkt des Kompensationssignals zur Ausgangsgröße y . Im diesem Fall ergibt sich demnach mit $\mathbf{b}_3$ als dritter Spalte der Eingangsmatrix $\mathbf{B}$ folgender Zusammenhang:

$$H(s) = \mathbf{c}^T \cdot [s \cdot \mathbf{E} - \mathbf{A}]^{-1} \cdot \mathbf{b}_3 \quad (4.53)$$

Gleichung (4.53) ergibt eine von den Arbeitspunktgrößen R_m und v_2 abhängige Übertragungsfunktion, deren Koeffizienten bei Arbeitspunktwechseln nachgeführt werden müssen. Die Auswertung von Gleichung (4.53) erfolgt am besten mittels Mathematik-Software.

Zur Generierung des Kompensationssignals $q(\phi)$ wird ein GRNN mit 40 Stützwerten verwendet. Diese werden im Periodizitätsbereich der Wickellage ϕ zwischen $0 \leq \phi \leq 2\pi$ äquidistant verteilt. Glättungsfaktor σ und Lernschrittweite η werden entsprechend Gleichung (4.54) gewählt.

$$\eta = 3 \cdot 10^{-7} \quad \sigma = 1.5 \cdot \frac{2\pi}{39} \quad (1.5\text{-facher Stützwerteabstand}) \quad (4.54)$$

Bei der Wahl der Lernschrittweite ist zu berücksichtigen, dass sich bei einem zu großen Wert externe Störungen stärker im Lernergebnis niederschlagen, da eine zusätzliche Verstärkung des Fehlersignals ϵ vorgenommen wird. Das Lernergebnis wird bei zu hohem η unruhig, was sogar zu Instabilität führen kann. Da von konstanten Sollwerten F_{soll} und v_{2soll} ausgegangen wird, vereinfacht sich die Bildung des Fehlersignals gemäß Gleichung (4.11) zu:

$$e = F_{12} - F_{soll} \quad (4.55)$$

Da $H(s)$ nicht *strictly positive real* ist, muss Fehlermodell 4 gemäß Kap. 3.2.6.4 implementiert werden.

Messergebnisse an der MODAM

In Bild 4.21 sind zunächst die Kraftschwankungen ohne Unrundheit (links) und mit Unrundheit (rechts) ohne Einsatz des selbstlernenden Reglers dargestellt. Ohne künstliche Unrundheit ergeben sich Schwankungen in einem Band von etwa 10 N, mit künstlicher Unrundheit schwankt die Bahnkraft mit einer Amplitude von etwa 55 N bei jeweils konstanter Sollkraft von $F_{soll} = 150 \text{ N}$ und einer Bahngeschwindigkeit von $v_2 = 2 \text{ m/s}$. Bei

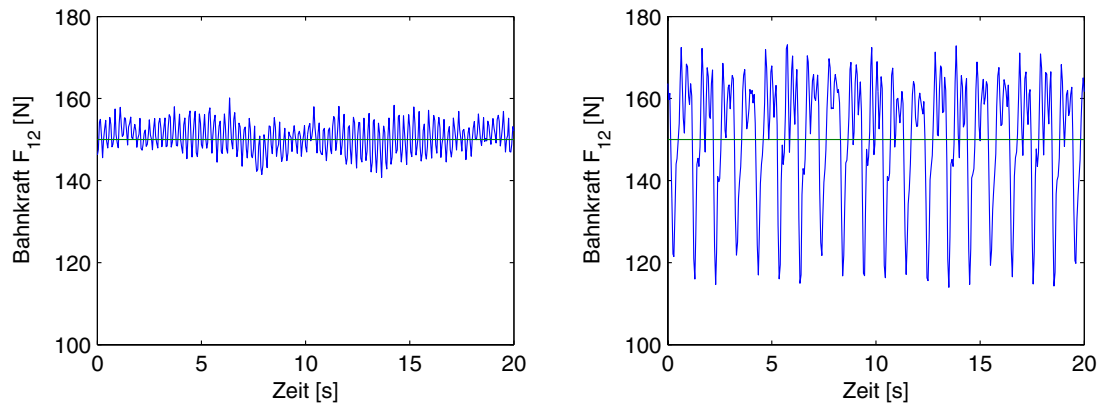


Bild 4.21: Kraftschwankungen ohne (links) und mit Unrundheit (rechts) bei $v_2 = 2 \text{ m/s}$

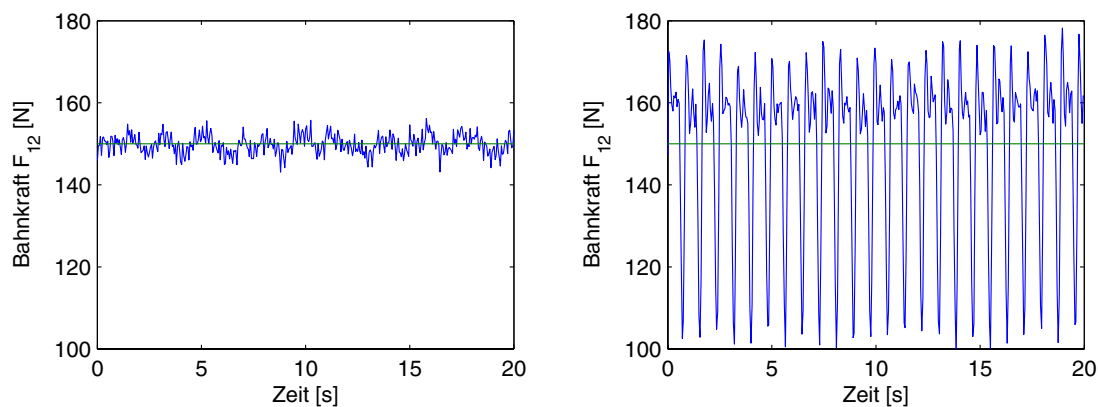


Bild 4.22: Kraftschwankungen ohne (links) und mit Unrundheit (rechts) bei $v_2 = 2.5 \text{ m/s}$

Erhöhung der Bahngeschwindigkeit auf $v_2 = 2.5 \text{ m/s}$ reduzieren sich die Kraftschwankungen ohne Unrundheit (Bild 4.22 links) gringfügig, wohingegen die Kraftschwankungen mit Unrundheit auf ein Band von 70 N ansteigen (Bild 4.22 rechts). Mit zunehmender Bahngeschwindigkeit nehmen die durch eine Unrundheit verursachten Kraftschwankungen zu, wodurch auch der Bedarf nach Kompensation steigt. Die Kraftschwankungen ohne künstliche Unrundheit sind als ideales Ziel der Kompensation durch den selbstlernenden Regler zu verstehen.

Wird der selbstlernende Regler auf die Unrundheitskompensation angewandt, so ergibt sich der in Bild 4.23 dargestellte Lernvorgang. Nach einer Lerndauer von 60 s haben die Kraftschwankungen ihr Minimum erreicht und schwanken mit einer Amplitude von ca. 12 N . Vergleicht man dies mit dem Idealverlauf aus Bild 4.21 links (Schwankungsbreite 10 N), so wird deutlich, dass der Effekt der künstlichen Unrundheit nahezu vollständig kompensiert wurde. Die verbleibenden Kraftschwankungen sind auf nichtmodellerte Effekte wie Messrauschen und Gerüstschwingungen des mechanischen Aufbaus zurückzuführen. Bei $t = 90 \text{ s}$ wurde die Kompensation kurzzeitig abgeschaltet, ohne das Lernergebnis des GRNN zurückzusetzen. Dadurch wird der Unterschied zwischen den Schwankungen ohne und mit Kompensation deutlicher. In Bild 4.23 rechts ist der Verlauf des Kompensations-

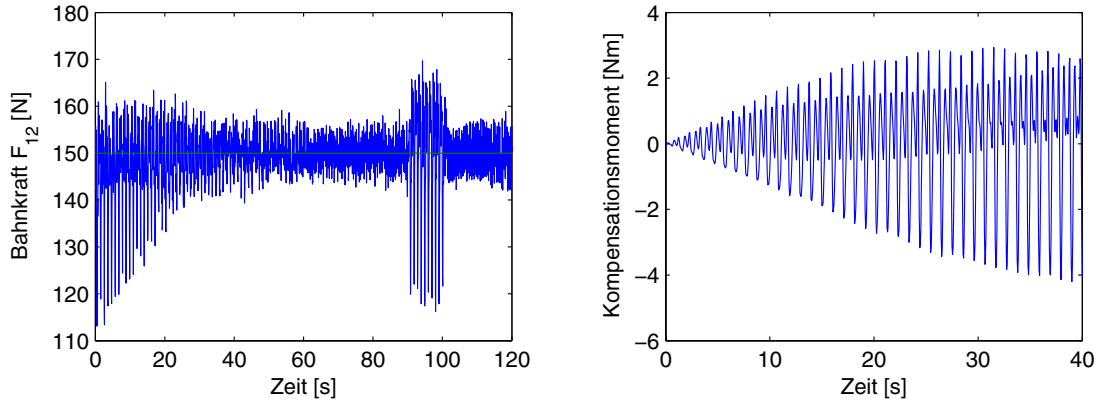


Bild 4.23: Kraftschwankungen (links) und Kompensationsmoment q während eines Lernvorgangs bei $v_2 = 2 \text{ m/s}$

moments q , das durch das GRNN durch Online-Adaption erzeugt wird, während des Lernvorgangs dargestellt. Das periodische Kompensationsmoment hat eine Amplitude von ca. 6 Nm , was von den verwendeten Asynchronmotoren mit einem Nennmoment von 140 Nm problemlos aufgebracht werden kann. Durch die Kompensation werden keine unmöglichen Momentenanforderungen an den Antrieb bzgl. Amplitude und Anstieg gestellt.

Bild 4.24 zeigt den gleichen Lernvorgang bei einer Papiergeschwindigkeit von $v_2 = 2.5 \text{ m/s}$. Durch Ab- und Zuschalten des Kompensationssignals bei $t = 90 \text{ s}$ wird der Gewinn durch

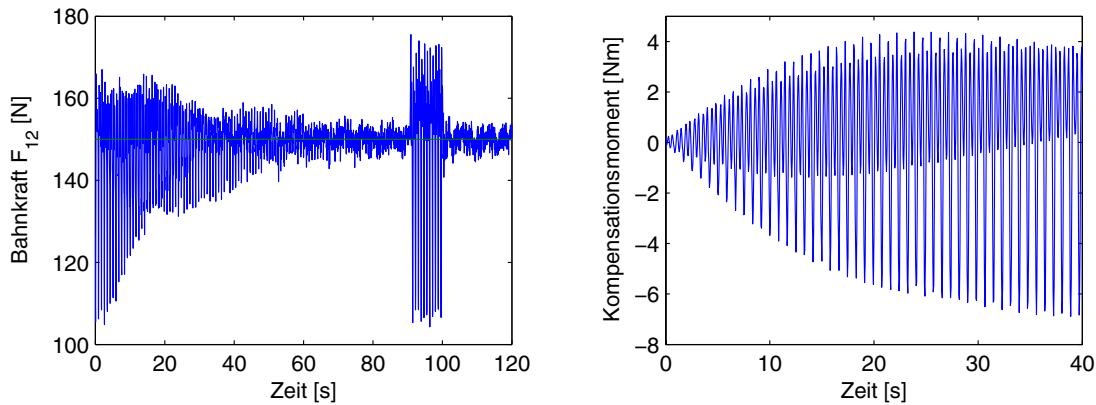


Bild 4.24: Kraftschwankungen (links) und Kompensationsmoment q während eines Lernvorgangs bei $v_2 = 2.5 \text{ m/s}$

die Kompensation besonders deutlich. Die Kraftschwankungen können von 70 N auf 10 N reduziert werden und erreichen den Kraftverlauf aus Bild 4.22. Auch hier kann die Auswirkung der künstlichen Unrundheit vollständig kompensiert werden. Das Kompensationsmoment steigt auf eine Amplitude von 10 Nm , was immer noch eine problemlose Anforderung an den Wickelmotor darstellt. Der Anstieg der Amplitude des Kompensationsmoments lässt sich anschaulich folgendermaßen erklären: Die Kraftschwankungen werden ideal kompensiert, wenn die Abwickelgeschwindigkeit des Papiers trotz einer Radiuserhöhung konstant bleibt. Dazu muss die Drehzahl des Wickels abgesenkt werden. Bei idealer Kompensation

ergibt sich dann ein pulsierender Drehzahlverlauf. Dazu sind Abbrems- und Beschleunigungsvorgänge nötig. Je höher die Papiergeschwindigkeit (und damit auch die Wickeldrehzahl) ist, desto kürzer ist die Zeitspanne, die zum Abbremsen und anschließenden Beschleunigen zur Verfügung steht. Dazu sind auch höhere Momente der Antriebsmaschine nötig.

Das Kompensationssignal $q(\phi)$ wird durch ein GRNN als Funktion der Winkelstellung ϕ des Wickels zwischen $0 \leq \phi \leq 2\pi$ approximiert. Das Lernergebnis des GRNN nach Ende der Messung bei $t = 120\text{ s}$ ist in Bild 4.25 für eine Bahngeschwindigkeit von $v_2 = 2\text{ m/s}$ dargestellt. Mit zunehmender Lerndauer nähert sich der Verlauf $q(\phi)$ einer festen Funktion

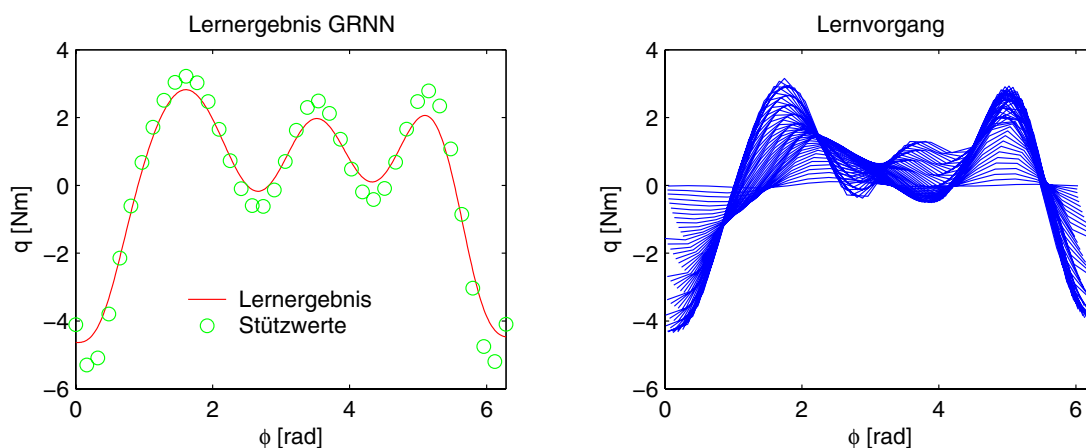


Bild 4.25: Kompensationssignal $q(\phi)$ nach einem Lernvorgang von 120 s (links) und während des Lernvorgangs (rechts) bei $v_2 = 2\text{ m/s}$

an. Damit ist auch experimentell gezeigt, dass $q(\phi)$ als statische Funktion dargestellt und damit durch ein GRNN approximiert werden kann.

Die Kraftschwankungen durch Wicklerunrundheiten wirken sich nicht nur im aktuellen Bahnabschnitt aus, sondern pflanzen sich über die Koppelung durch die Papierbahn in alle weiteren Bahnabschnitte fort. Eine Unrundheitskompensation am Wickler bedeutet demnach geringere Kraftschwankungen in der gesamten Anlage. Bild 4.26 zeigt die Auswirkungen von Wicklerunrundheiten und deren Kompensation auf den nachfolgenden Bahnabschnitt. In Bild 4.26 rechts ist zu erkennen, dass die Schwankungen von F_{23} im nachfolgenden Bahnabschnitt durch die Unrundheitskompensation um den Faktor drei zurückgehen.

Nun wird das Verhalten der Unrundheitskompensation bei einem Arbeitspunktwechsel der Bahngeschwindigkeit untersucht. Bei einem Geschwindigkeitswechsel ändert sich die Periodendauer der Nichtlinearität und das Kompensationssignal muss nachgelernt werden (siehe Kap. 4.2.3). Bild 4.27 zeigt den Bahnkraft und Geschwindigkeitsverlauf bei einer Geschwindigkeitsänderung von $v_2 = 2\text{ m/s}$ auf $v_2 = 2.5\text{ m/s}$. Während des Geschwindigkeitsübergangs wurde die Adaption des GRNN abgeschaltet, da alle Annahmen über die Nichtlinearität und zur Fehlerbildung nicht mehr zutreffen. Insbesondere handelt es sich während des transienten Vorgangs nicht mehr um eine periodische Nichtlinearität. Die hohen Abweichungen der Bahnkraft während der Beschleunigung resultieren aus nicht idealen

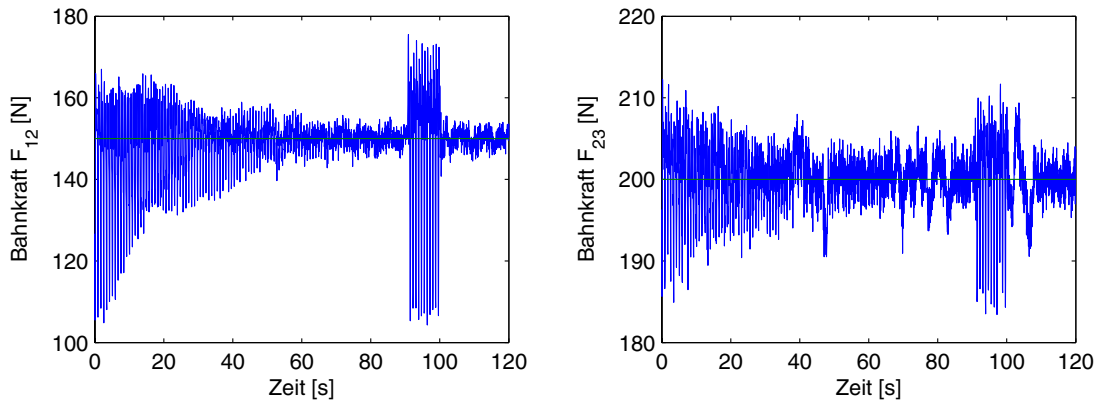


Bild 4.26: Bahnkräfte F_{12} und F_{23} während des Lernvorgangs bei $v_2 = 2.5 \text{ m/s}$

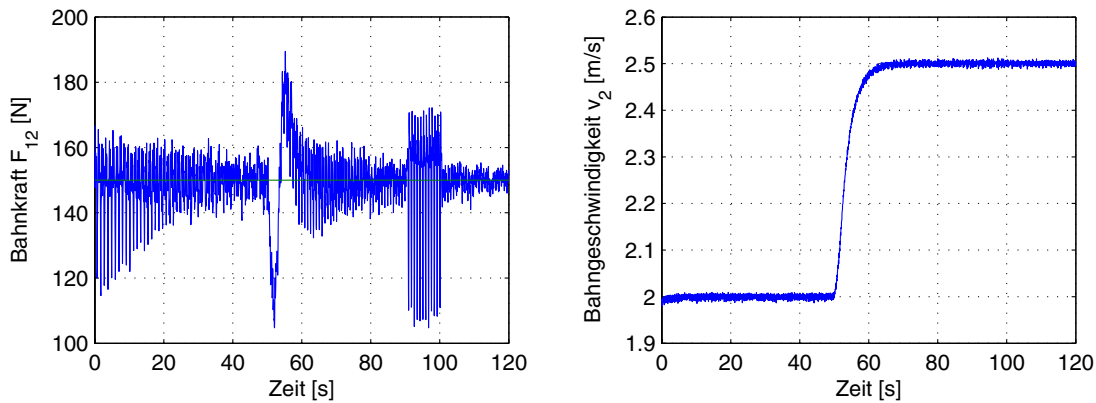


Bild 4.27: Kraft und Geschwindigkeitsverlauf bei einem Arbeitspunktwechsel

Vorsteuerungen der Momente und Drehzahlen. Nach Abschluss der Beschleunigungsphase (bei $t = 60 \text{ s}$) wird die Adaption des GRNN wieder aktiviert. Es sind zunächst größere Kraftschwankungen als vor Beginn des Arbeitspunktwechsels erkennbar. Mit weiterer Lerndauer nehmen diese jedoch stetig weiter ab. Bei $t = 90 \text{ s}$ wird kurzzeitig die Kompensation abgeschaltet.

Durch die Anwendung des selbstlernenden Reglers auf ein komplexes und praxisnahes System ist dessen Tauglichkeit zur Unterdrückung von Kraftschwankungen in kontinuierlichen Fertigungsanlagen nachgewiesen worden. Es haben sich die folgenden Vorteile ergeben:

- Sehr gute Kompensation von Kraftschwankungen bei konstanten Bahngeschwindigkeiten bereits nach einer Lerndauer von 60 s .
- Arbeitspunktwechsel werden durch Nachlernen der Netzwerkgewichte behandelt. Dies geschieht im selben Zeithorizont wie die erstmalige Adaption auch.
- Die bestehende Reglerstruktur der Anlage musste nicht verändert werden. Der selbstlernende Regler unterstützt die vorhandenen PI-Regler.

5 Modellgestützte prädiktive Regelungen

Die Idee der modellbasierten prädiktiven Regelung (MPC) reicht nach [70, 50] bis in die Zeit um 1960 zurück. Zadeh und Whalen erkannten 1962 den Zusammenhang zwischen dem *minimum time optimal control problem* und der linearen Programmierung. Ein Jahr später schlug Propoi den zurückweichenden Horizont als *open-loop optimal feedback-Strategie* vor. Dennoch gilt es in der Literatur als anerkannt, dass die Entstehung (bzw. die Wiederentdeckung) der MPC auf Mitte bis Ende der 70er Jahre datiert wird. In dieser Zeit wurde ein prädiktives Regelungsverfahren in der chemischen und verfahrenstechnischen Industrie erstmals angewendet. Erst lange nach dem erfolgreichen Einsatz in der Industrie fand dieses Verfahren Mitte der 80er Jahre den Weg in die Universitäten und erhielt dort ein tieferes akademisches Verständnis. Etwa 20 Jahre nach der Einführung der MPC konnte der erste Stabilitätsbeweis für prädiktiv geregelte Prozesse mit Beschränkungen angegeben werden.

Die langsamen Abtastraten bei chemischen und thermischen Prozessen wie bei katalytischen Krackanlagen, Destillationssäulen und Glasöfen ließen eine vergleichsweise rechenaufwändige prädiktive Regelung zu. Die vorteilhafte Eigenschaft der MPC, Prozessbeschränkungen während des Reglerentwurfes direkt berücksichtigen zu können, wurde nach [20] aufgrund der Energiekrise im Winter 1973/74 erkannt und später besonders in der Erdölindustrie mit Erfolg eingesetzt, um effizientere Ausbeuten zu erhalten. Seither verbreitete sich die MPC in der verfahrenstechnischen Industrie und gilt heute als de-facto-Standard für high-performance-Regelungen. Erst mit der steigenden Rechenleistung moderner Mikroprozessoren erweiterte sich das Anwendungsgebiet dieser Regelstrategie auch auf *schnelle* Prozesse wie Roboter und elektrische Antriebe.

Die modellgestützte prädiktive Regelung (*Model Predictive Control*, MPC) bezeichnet eine Klasse modellgestützter Regelungsverfahren, die ein Prozessmodell zur Prädiktion relevanter zukünftiger Prozessgrößen nutzt. Die zukünftigen Auswirkungen der aktuellen und zukünftigen Stellgrößen werden abgeschätzt und im Regelalgorithmus optimiert. Der prädiktive Regelungsansatz zeichnet sich gegenüber anderen Verfahren durch seine vielseitigen Einsatzmöglichkeiten aus. Das Konzept ist sowohl auf lineare als auch auf nichtlineare Prozesse anwendbar. Durch Online-Optimierung der Stellgröße können Prozess- und Stellgrößenbeschränkungen berücksichtigt werden. Sind Störgrößen messbar oder können gar in der Zukunft vorhergesagt werden, so können diese Kenntnisse direkt in das Regelungskonzept eingebunden werden, um ein bestmögliches Regelergebnis zu erzielen. Die Einordnung prädiktiver Regler in lineare und nichtlineare Verfahren wird anhand der verwendeten Prozessmodelle vorgenommen. Begrenzungen werden als Nebenbedingungen in den Optimierungsalgorithmus eingebunden und bestimmen zusammen mit dem Prozessmodell den verwendeten Optimierungsalgorithmus.

5.1 Prinzipielle Funktionsweise

Das Grundprinzip modellbasierter prädiktiver Regelungen besteht in der Auswertung des zukünftigen Prozessverhaltes mit dem Ziel, einen optimalen Stellgrößenverlauf zu erzeugen. Prädiktive Regelungen basieren auf den folgenden Kernpunkten:

1. Um die zukünftige Regelgüte beurteilen zu können, muss zuerst eine **Referenztrajektorie** bestimmt werden. Diese stellt den zukünftigen Wunschverlauf der Regelgröße dar (zukünftiger Sollwertverlauf).
2. Auf der Grundlage eines Prozessmodells wird als Funktion zukünftiger Stellgrößenverläufe das Prozessverhalten prädiziert. Die **Prädiktion der Regelgrößen** liefert den funktionellen Zusammenhang zwischen Stellgrößen und Regelgrößen in der Zukunft. Zur Bestimmung des zukünftigen Prozessverhaltens können neben Stellgrößen auch bekannte Störgrößen eingebunden werden.
3. Um eine möglichst genaue Prädiktion zu gewährleisten, muss die Schätzung zukünftiger Prozessgrößen auf möglichst aktueller Information über den Prozess beruhen. Dazu muss bei jedem Zeitschritt der geschätzte Prozesszustand mit dem tatsächlich vorhandenen abgeglichen werden. Dieser Abgleich ist wegen unbekannter Störgrößen und nicht exakter Modellierung des Prozesses nötig. Die **Einbindung aktueller Prozessinformationen** gewährleistet eine möglichst realistische Vorhersage.
4. Die **Definition der Gütefunktion** legt die Regelziele fest. Sie setzt sich aus mehreren Kostenanteilen zusammen, die jeweils ein spezielles Regelziel repräsentieren. Entsprechend ihrer Wichtigkeit werden diese Anteile unterschiedlich stark gewichtet.
5. MPC-Verfahren arbeiten üblicherweise zeitdiskret. Die **Optimierung des Stellgrößenverlaufs** bzgl. des gewählten Gütefunktional wird zu jedem Abtastschritt mit der jeweils aktuellen Prozessinformation neu durchgeführt. Aus dem optimierten Stellgrößenverlauf wird nur das erste Element tatsächlich auf den Prozess geschaltet, alle weiteren Elemente bleiben ungenutzt. Dies nennt man die Strategie des zurückweichenden Horizonts (*receding horizon*).

Bei prädiktiven Regelungen handelt es sich meist um zeitdiskret formulierte Algorithmen, wobei jedoch unter erhöhtem Aufwand und ohne relevante Vorteile auch eine zeitkontinuierliche Beschreibung des Regelgesetzes denkbar wäre. Allen prädiktiven Regelungen ist gemeinsam, dass – im Gegensatz zu herkömmlichen Regelstrategien – nicht vergangene und aktuelle Soll- und Istwerte für die Berechnung der Stellgröße verwendet werden, vielmehr bilden **zukünftige**, geschätzte Regeldifferenzen die Basis des Regelgesetzes, welches sich aus der Minimierung von Kosten ergibt, die durch eine Gütefunktion beschrieben werden. Die Gütefunktion, auch Kostenfunktion genannt, bestraft zukünftige Regeldifferenzen, indem diesen Abweichungen entsprechende Kosten zugeordnet werden. In diesem Sinne reiht sich die MPC in die Gruppe der Optimal-Steuerungen bzw. Optimal-Regelungen ein. Mit Hilfe der Gütefunktion können verschiedene Regelziele formuliert werden, wie zum Beispiel die Forderung nach gutem Folgeverhalten oder die Einsparung

von Stellenergie. Um diese gegensätzlichen Forderungen einhalten zu können, müssen Gewichtungen der Regelziele vorgenommen werden, aufgrund derer ein Optimierer die beste Lösung errechnen kann. Die Verwendung eines geeigneten Prädiktionsmodelles zur Vorhersage des zukünftigen Prozessverhaltens ist für prädiktive Regelungen charakteristisch und stellt ein wesentliches Erkennungsmerkmal für diese Regelungsstrategie dar. Abhängig von noch nicht bestimmten, aktuellen und zukünftigen Stellgrößen wird der Verlauf des Streckenausganges mit einem Prozessmodell prädiziert. Die gesuchte, aktuelle Stellgröße ergibt sich als das Ergebnis einer Optimierungsaufgabe. Unter Verwendung numerischer Optimierungsalgorithmen können Prozessbeschränkungen als Nebenbedingungen (NB) in der Minimierung direkt berücksichtigt werden, wenn diese als Gleichungsnebenbedingungen (GNB) oder Ungleichungsnebenbedingungen (UNB) formuliert werden. Bei quadratischen Gütefunktionen und linearen Prozessmodellen entsteht dadurch ein Problem der *quadratischen Programmierung*, das mit leistungsfähigen Standardalgorithmen gelöst werden kann [86]. Bild 5.1 zeigt die grundsätzliche Struktur eines prädiktiven Regelkreises.

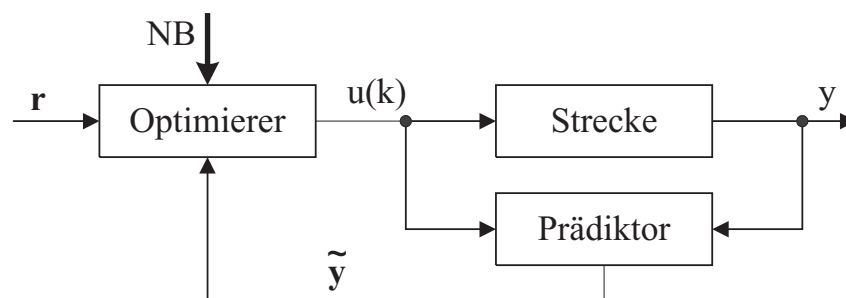


Bild 5.1: Struktur eines prädiktiven Regelkreises

Der Optimierer erhält den zukünftigen Sollwertverlauf $\mathbf{r}$, die Nebenbedingungen NB und den vorhergesagten Prozessausgang $\tilde{\mathbf{y}}$ als Eingangsinformationen und berechnet daraus die optimale Stellgrößenfolge $\mathbf{u}$, deren erstes Element $u(k)$ dem Prozesseingang als aktuelle Stellgröße aufgeschaltet wird. Eine direkte Rückkopplung des Prozessausganges besteht dabei nicht. Der Regelkreis wird durch Verwendung der prädizierten Ausgangssignale $\tilde{\mathbf{y}}$ geschlossen, die vom Prädiktor geschätzt werden. Dieser berechnet die zukünftige Entwicklung des Prozessausganges, abhängig von zukünftigen Stellgrößen und vom aktuellen Prozesszustand.

Die zu optimierende Kostenfunktion berücksichtigt alle Regeldifferenzen innerhalb eines festgelegten Vorhersagezeitraumes, der sich vom unteren bis zum oberen Prädiktionshorizont erstreckt. Der untere Prädiktionshorizont liegt N_1 Zeitschritte in der Zukunft. Für die spezielle Wahl $N_1 = 0$ beginnt der Vorhersagezeitraum zum gegenwärtigen Zeitpunkt k . Der Vorhersagezeitraum endet nach N_2 Zeitschritten zum Zeitpunkt $k + N_2$. Durch das Fortschreiten des aktuellen Zeitpunktes auf der diskreten Zeitachse und durch die festgelegten Horizonte bewegt sich auch der Vorhersagezeitraum entlang der Zeitachse mit. Dies findet Ausdruck in den üblichen Bezeichnungen *gleitender Horizont* und *zurückweichender Horizont* bzw. *receding horizon*. Um die Freiheitsgrade des Optimierungsproblem und damit auch den Rechenaufwand zu reduzieren, wird ein Stellhorizont N_u eingeführt, der

kleiner oder gleich dem oberen Prädiktionshorizont ist. Nur bis zum Zeitpunkt $k + N_u$ werden Stellgrößenänderungen angenommen, anschließend bleibt die Stellgröße konstant. Dies entspricht dem Ausfall des Stellgliedes, der ein Verharren der Stellgröße in ihrem aktuellen Wert bedingt. Der Ansatz eines Stellhorizonts N_u liegt auch deshalb nahe, da nur das erste Element der berechneten Stellgrößenfolge auf den Prozess geschaltet wird. Beim nächsten Zeitschritt wird die Optimierung wiederholt. Es ist daher nicht nötig N_2 Stellgrößen zu berechnen und nur eine einzige davon tatsächlich zu verwenden. Die Einschränkung der Freiheitsgrade führt zwar zu einer suboptimalen Lösung, allerdings kann der geringe Qualitätsverlust in der Regelgüte bei einem hinreichend langen Stellhorizont ($N_u = 3 \dots 7$) in praktischen Anwendungen meist in Kauf genommen werden.

Die Lösung der Minimierungsaufgabe ergibt eine Stellgrößenfolge, welche im Sinne einer Steuerung auf den Prozess aufgeschaltet werden könnte. Innerhalb des Vorhersagezeitraumes wäre diese Stellfolge im Sinne der Kostenfunktion optimal. Allerdings können hierdurch Störungen und Modellunsicherheiten nicht berücksichtigt werden. Weiterhin besteht das Problem, dass die weitere Sollwerttrajektorie nach Ablauf des aktuellen Vorhersagezeitraumes in der Optimierung nicht beachtet worden ist. Deshalb wird nur die aktuelle Stellgrößenänderung tatsächlich auf die Regelstrecke geschaltet. Alle weiteren Stellgrößenänderungen werden verworfen. Wegen des zurückweichenden Horizontes muss im nächsten Zeitschritt am Ende des Vorhersagezeitraumes ein zusätzlicher Sollwert berücksichtigt werden, so dass ein abgeändertes Optimierungsproblem gelöst werden muss.

Zur Beschreibung der bei der prädiktiven Regelung verwendeten Größenverläufe wird folgende Schreibweise gewählt: Bei allen Variablen mit Tilde, z.B. $\tilde{x}$, handelt es sich um prädizierte Größen. Ferner muss manchmal unterschieden werden, zu welchem Zeitpunkt eine Prädiktion durchgeführt wurde. $\tilde{x}(k+1|k)$ bezeichnet eine prädizierte Größe für den Zeitpunkt $k+1$, berechnet zum Zeitpunkt k , wogegen $\tilde{x}(k+1|k-1)$ die selbe Größe bezeichnet, die zum Zeitpunkt $k-1$ berechnet wurde. Dieser Unterschied ist wichtig, da als Basis der Prädiktion jeweils unterschiedliche Information verwendet wurde. Wenn eine prädizierte Größe mit $\tilde{x}(k+1)$ bezeichnet wird, ist der Einfachheit halber immer die zum Zeitpunkt k berechnete Größe gemeint (z.B. $\tilde{x}(k+1) \equiv \tilde{x}(k+1|k)$).

Ein geeignetes Gütekriterium für einen Single-Input Single-Output (SISO) Prozess ist eine quadratische Gütefunktion:

$$J(\Delta \mathbf{u}) = \sum_{j=N_1}^{N_2} [r(k+j) - \tilde{y}(k+j)]^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) + \mu \cdot \sum_{j=0}^{N_u} u^2(k+j) \quad (5.1)$$

mit $\lambda, \mu \geq 0$. Sie bewertet die Kosten der Regeldifferenz und des Stelleingriffes. Die gewählten Gewichtungsfaktoren dürfen keine negativen Werte aufweisen, da sich ansonsten Stellgrößen kostenmindernd auswirken können und eine indefinite Kostenfunktion mit einem Minimum bei $-\infty$ entsteht. Der zukünftige zeitdiskrete Sollwertverlauf ist durch die Elemente $r(k+j)$ beschrieben, das zukünftige prädizierte Prozessverhalten durch $\tilde{y}(k+j)$. Die Variable u steht für die Stellgröße, Δu bedeutet die Stellgrößenänderung zwischen zwei Zeitpunkten. Die gleichzeitige Minimierung der Regeldifferenz und der Stellgröße stellen gegensätzliche Forderungen dar. Die quantitative Beschreibung der Bedeutung der unterschiedlichen Regelziele muss durch eine Gewichtung mit den Faktoren λ und μ vorgenommen werden. Die Berücksichtigung der Stellgröße u führt im Allgemeinen zu einer bleiben-

den Regeldifferenz im stationären Zustand [44, 58]. Daher wird im Folgenden die Stellenergie nicht berücksichtigt, μ wird zu null gesetzt. Die Gewichtung der Stellgrößenänderung dagegen wird zugelassen, da hierdurch die stationäre Genauigkeit nicht beeinflusst wird. Es wird lediglich die Geschwindigkeit der Stellgrößenänderung beeinflusst. Weiterhin besteht die Möglichkeit, die Regeldifferenz zeitabhängig zu gewichten. Aufgrund von Modellunsicherheiten entsteht ein Fehler in der Prädiktion des Prozessverhaltens, der für zeitlich weiter entfernte Ausgangswerte ansteigt. Daher kann eine Gewichtungsmatrix $\mathbf{Q}$ eingeführt werden, welche weiter in der Zukunft liegende Regeldifferenzen schwächer gewichtet. Bei sehr gutem Prozessmodell ist jedoch auch eine stärkere Gewichtung durch die Matrix $\mathbf{Q}$ für entferntere Regelabweichungen denkbar, die anfangs große Abweichungen zulässt, um dann später bessere Genauigkeit zu erzielen. Eine derartige Gütefunktion

$$\begin{aligned} J(\Delta \mathbf{u}) &= \sum_{j=N_1}^{N_2} q_j [r(k+j) - \tilde{y}(k+j)]^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) = \\ &= (\mathbf{r} - \tilde{\mathbf{y}})^T \mathbf{Q} (\mathbf{r} - \tilde{\mathbf{y}}) + \lambda \Delta \mathbf{u}^T \Delta \mathbf{u} \end{aligned} \quad (5.2)$$

mit

$$\begin{aligned} \tilde{\mathbf{y}} &= [\tilde{y}(k+N_1), \tilde{y}(k+N_1+1), \dots, \tilde{y}(k+N_2)] && \in \mathbb{R}^{N_2-N_1+1} \\ \mathbf{r} &= [r(k+N_1), r(k+N_1+1), \dots, r(k+N_2)] && \in \mathbb{R}^{N_2-N_1+1} \\ \Delta \mathbf{u} &= [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_u)] && \in \mathbb{R}^{N_u+1} \\ \mathbf{Q} &= \text{diag}(q_j) \quad q_j \geq 0 && \in \mathbb{R}^{N_2-N_1+1 \times N_2-N_1+1} \end{aligned} \quad (5.3)$$

mit den $N_2 - N_1 + 1$ abnehmenden oder ansteigenden Gewichtungsfaktoren q_j bringt zwar zusätzliche Einstellparameter, die jedoch kaum neue Freiheitsgrade liefern, da durch entsprechende Wahl der Horizonte N_1 und N_2 ein ähnlicher Effekt erzielt werden kann. Für den speziellen Fall der Wahl $\mathbf{Q} = \mathbf{E}$ ergibt sich die Gütefunktion

$$J(\Delta \mathbf{u}) = \sum_{j=N_1}^{N_2} [r(k+j) - \tilde{y}(k+j)]^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (5.4)$$

die für alle Anwendungen in dieser Arbeit Verwendung fand. Anhand von Bild 5.2 soll die Bedeutung von Gleichung (5.4) verdeutlicht werden, auf deren Minimierung das prädiktive Regelgesetz basiert. Zum aktuellen Zeitpunkt k muss die Stellgrößenänderung $\Delta u(k)$ berechnet werden. Dazu wird mit Hilfe eines adäquaten Prädiktionsmodells der Verlauf des Prozessausganges, abhängig von den Stellgrößenänderungen $\Delta u(k), \Delta u(k+1), \Delta u(k+2), \dots, \Delta u(k+N_u)$, vorhergesagt. Die Größe N_u wird als Stellhorizont bezeichnet, da Stellgrößenänderungen nur bis dorthin zugelassen werden. Ab dem Zeitpunkt $k + N_u$ wird die Stellgröße als konstant angenommen. Wie aus dem Gütefunktional (5.4) hervorgeht, erstreckt sich der Vorhersagezeitraum vom Zeitpunkt $k + N_1$ bis $k + N_2$. Die Regelabweichung wird nur innerhalb dieses in der Zukunft liegenden Zeitraumes bewertet. Da jedoch der zukünftige Verlauf des Prozessausganges zum gegenwärtigen Zeitpunkt noch nicht feststehen kann, muss ein Prozessmodell zur Prädiktion der Ausgangstrajektorie verwendet werden. Die Unterscheidung der vorhergesagten Werte von den zu späteren Zeitpunkten tatsächlich erreichten Werten ist durchaus sinnvoll, da die prädizierten Werte nur dann erreicht würden, wenn die gesamte berechnete Stellgrößenfolge $\Delta \mathbf{u}$ gemäß einer

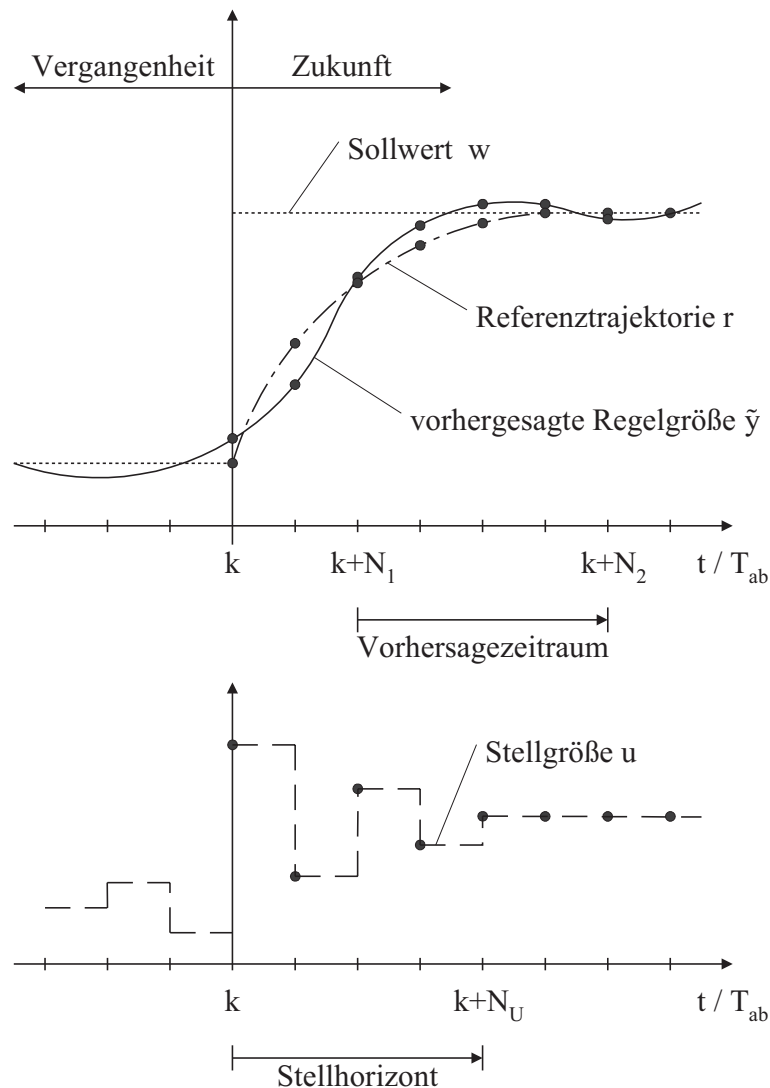


Bild 5.2: Konzept der prädiktiven Regelung

Optimalsteuerung auf den Prozess geschaltet würde, wobei weder Modellunsicherheiten noch Störungen auftreten dürften. Im Allgemeinen weicht auch die Stellgröße $\Delta u(k|k-1)$, die im vergangenen Zeitschritt berechnet wurde, von der Stellgröße $\Delta u(k|k)$ ab, die im aktuellen Zeitschritt berechnet wird, obwohl beide Größen den Prozesseingang zum gleichen Zeitpunkt k beschreiben. Dies liegt an der unterschiedlichen Prozessinformation zum Zeitpunkt der Berechnung. Die Optimierung der Gütefunktion (5.4) ergibt den Vektor $\Delta \mathbf{u}$, der N_u Stellgrößenänderungen enthält. Modellunsicherheiten und die Unkenntnis von zukünftigen Störgrößen verhindern eine exakte Prädiktion. Daher wird im Sinne einer closed-loop-Regelung der Prozess nur mit dem ersten Element $\Delta u(k)$ des Vektors $\Delta \mathbf{u}$ beaufschlagt. Der gesamte Vektor $\Delta \mathbf{u}$ wird dann im nächsten Zeitschritt wieder neu berechnet. Weiterhin wird der zukünftige Sollwertverlauf zur Differenzbildung im ersten Summanden des Gütefunktionalen benötigt. Die Verwendung von zukünftigen Sollwerten ist eine typische Eigenschaft prädiktiver Regelungen. Dies ermöglicht eine frühzeitige Reaktion des

geregelten Systems auf Sollwertänderungen. Jedoch ist die Anwendbarkeit des prädiktiven Regelungskonzepts keineswegs an diese Information gebunden. Wenn kein Wissen über den zukünftigen Sollwertverlauf vorliegt, kann die zukünftige Sollwerttrajektorie als konstant angenommen werden. Der benötigte Sollwertvektor $\mathbf{r}$ enthält dann in all seinen Elementen den aktuellen und damit bekannten Sollwert.

Die Berücksichtigung von Eingangs- und Zustandsbeschränkungen des Prozesses im Reglerentwurf ist eine weitere charakteristische Eigenschaft von prädiktiven Regelungen. Für Eingangsgrößen können sowohl Stellgrößen- als auch Stellgrößenänderungsbeschränkungen berücksichtigt werden.

$$u_{min} \leq u(k) \leq u_{max} \quad (5.5)$$

$$\Delta u_{min} \leq \Delta u(k) \leq \Delta u_{max} \quad (5.6)$$

Bei Zustandsbeschränkungen reicht in der Praxis die Berücksichtigung von Absolutwertbeschränkungen aus.

$$\mathbf{x}_{min} \leq \mathbf{x}(k) \leq \mathbf{x}_{max} \quad (5.7)$$

Bisher existiert kein anderes Regelungsverfahren, das gleichzeitig Zustands und Eingangsbeschränkungen berücksichtigen kann. Dies liegt daran, dass bei Existenz beider Begrenzungen Zustände des Prozesses erreicht werden können, die die Verletzung einer Begrenzung unumgänglich machen. Diese kritischen Zustände kann der prädiktive Regelalgorithmus dadurch umgehen, dass Begrenzungen nicht nur auf die aktuellen Werte, sondern alle Werte im Prädiktionshorizont angewendet werden. Dadurch kann der prädiktive Regler rechtzeitig gegensteuern. Unter Einbeziehung von Beschränkungen ergibt sich das Optimierungsproblem unter Nebenbedingungen (siehe Kap. 6.5), das in keinem Fall mehr analytisch lösbar ist. Der Einsatz numerischer Verfahren ist hier unausweichlich. Aus mathematischer Sicht begrenzen Eingangsbeschränkungen den Definitionsbereich der Gütefunktion, während Zustandsbegrenzungen den Wertebereich der Prädiktion einschränken. In [88] wird die Problematik diskutiert, dass die strikte Einhaltung von Begrenzungen eine Lösung des Optimierungsproblems unmöglich machen kann.

Eine Problematik der prädiktiven Regelung ist deren Stabilitätsnachweis. Dies liegt am endlichen Prädiktionshorizont. Es erfolgt immer nur eine Optimierung des Prozessverhaltens innerhalb des Horizonts, während das Verhalten außerhalb außer Acht gelassen wird. Da Stabilität aber eine Eigenschaft für $t \rightarrow \infty$ ist, kann mit einem endlichen Gütefunktional generell keine Stabilitätsaussage getroffen werden. Die Stabilitätsproblematik wird in Kap. 7 ausführlicher betrachtet.

5.2 Prädiktive Regelverfahren auf Basis linearer Prozessmodelle

Die folgenden prädiktiven Regelverfahren beruhen auf linearen Prozessmodellen. Jedes der dargestellten Verfahren ist prinzipiell für Ein- oder Mehrgrößensysteme anwendbar. Die Darstellung erfolgt der Einfachheit halber für Eingrößensysteme (SISO), da hier bereits alle charakteristischen Eigenschaften deutlich werden. Zwei Regelverfahren mit grundsätzlich

unterschiedlichen Modellen werden vorgestellt: Das *Dynamic Matrix Control* Verfahren verwendet ein FIR-Impulsantwortmodell (siehe Kap. 2.2.1), wogegen beim *Generalized Predictive Control* Verfahren ein Übertragungsfunktionsmodell zum Einsatz kommt. Ein guter Überblick über lineare prädiktive Regelungsverfahren ist in [6, 84] enthalten.

5.2.1 Dynamic Matrix Control

Das *Dynamic Matrix Control* (DMC) Verfahren kommt vor allem in der verfahrenstechnischen Industrie zum Einsatz. Das Prädiktionsmodell der Ausgangsgröße basiert auf dem zeitdiskreten stabilen FIR-Impulsantwortmodell.

$$\begin{aligned} \tilde{y}(k+j) = & \sum_{i=1}^j h(i)\Delta u(k+j) + \sum_{i=j+1}^{m-1} h(i)\Delta u(k+j-i) \\ & + h(m)u(k+j-m) + d(k+j) \end{aligned} \quad (5.8)$$

Die vier Summanden beschreiben (in dieser Reihenfolge) den Einfluss von zukünftigen Stellgrößen, vergangenen Stellgrößen, Anfangsbedingungen und Störungen. Die $h(i)$ sind die Gewichte des zeitdiskreten Impulsantwortmodells. Während die vergangenen Stellgrößen zum Zeitpunkt k bekannt sind, stellen die zukünftigen Stellgrößen die Freiheitsgrade des Optimierungsproblems dar. Im DMC Verfahren wird angenommen, dass die zukünftigen Störgrößen alle gleich der aktuellen Störgröße sind, die sich aus dem aktuellen Prädiktionsfehler ergibt.

$$d(k+j) = d(k) = y(k) - \sum_{i=1}^{m-1} h(i)\Delta u(k-i) \quad (5.9)$$

Den Namen *Dynamic Matrix Control* verdankt das Verfahren der sog. dynamischen Matrix $\mathbf{H}$, die den Einfluss der zu bestimmenden N_u Stellgrößenänderungen auf die prädizierte Regelgröße $\tilde{y}$ beschreibt.

$$\mathbf{H} = \begin{bmatrix} h(1) & 0 & \dots & 0 \\ h(2) & h(1) & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ h(N_u) & \dots & \dots & h(1) \\ \vdots & \vdots & \vdots & \vdots \\ h(N_2) & \dots & \dots & h(N_2 - N_u + 1) \end{bmatrix} \quad (5.10)$$

Der Vektor $\tilde{\mathbf{y}}$ der prädizierten Ausgangsgrößen (von 1 bis N_2) ergibt sich unter Verwendung des Stellgrößenvektors $\Delta \mathbf{u}$ (von 1 bis N_u) zu

$$\tilde{\mathbf{y}} = \mathbf{f}_D + \mathbf{H}\Delta \mathbf{u} \quad (5.11)$$

Die freie Antwort $\mathbf{f}_D$ beinhaltet den Einfluss vergangener Eingangsgrößen und den Schätzwert der Störung aus Gleichung (5.9). Alle DMC-Verfahren verwenden das Prädiktionsmodell aus Gleichung (5.8). Der ursprüngliche DMC-Regler aus [66] und der später in

[70] vorgestellte *Quadratic Programming Dynamic Matrix Control* (QDMC) Algorithmus verwenden quadratische Gütefunktionen.

$$J(\Delta \mathbf{u}) = [\tilde{\mathbf{y}} - \mathbf{r}]^T \mathbf{\Gamma} [\tilde{\mathbf{y}} - \mathbf{r}] + \Delta \mathbf{u}^T \mathbf{\Lambda} \Delta \mathbf{u} \quad (5.12)$$

Nebenbedingungen werden in Form von linearen Ungleichungen berücksichtigt. Ohne Nebenbedingungen ist eine analytische Lösung der Gütefunktion (5.12) möglich; mit Nebenbedingungen kann das effiziente Verfahren der quadratischen Programmierung verwendet werden [86].

Im Gegensatz zu QDMC, wird im *Linear Dynamic Matrix Control* (LDMC) ein lineares Optimierungsproblem aufgebaut [81]. Dafür wird im Gütefunktional eine L_1 -Norm verwendet. Die Minimierung bei gleichzeitiger Beachtung von Nebenbedingungen wird durch ein lineares Programm in einer garantiert endlichen Anzahl an Iterationen gelöst. Dies ist vor allem bei Prozessen mit kleiner Abtastzeit interessant. Aufgrund der Modellbeschreibung durch ein FIR-Modell, können mit dem DMC-Verfahren nur stabile Prozesse behandelt werden. Die Auswirkungen der Einstellparameter und die Eigenschaften der Regelung wurden in [76] untersucht.

5.2.2 Generalized Predictive Control

Das wohl bekannteste prädiktive Regelungsverfahren ist der von Clarke vorgestellte *Generalized Predictive Control* Algorithmus [64, 65]. Dieses Verfahren nutzt ein CARIMA (Controlled Auto-Regressive Integrated and Moving Average) Modell zur Prozessnachbildung. Es handelt sich um ein Übertragungsfunktionsmodell mit zusätzlichem Rauschanteil.

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})} u(k) + \frac{Z(z^{-1})}{\Delta A(z^{-1})} \xi(k) \quad (5.13)$$

A , B und Z stellen Polynome im Ein-Schritt-Verschiebeoperator z^{-1} dar.

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \dots + b_{nb} z^{-nb} \\ Z(z^{-1}) &= z_0 + z_1 z^{-1} + \dots + z_{nz} z^{-nz} \end{aligned} \quad (5.14)$$

ξ kennzeichnet zeitdiskretes weißes Rauschen und $\Delta = 1 - z^{-1}$ ist der zeitdiskrete Differenzieroperator. Durch die allgemeine Wahl des Rauschsignalfilters sind alle wichtigen Störmodelle enthalten: Der autoregressive Prozess (AR) für $z_0 \xi(k)$, der integrierende Prozess (I) für $1/\Delta \xi(k)$ und der Prozess mit gleitendem Mittel (moving average, MA) für $Z(z^{-1}) \xi(k)$.

Das Prozessmodell in Gleichung (5.13) kann in den Zustandsraum transformiert werden. Dies erleichtert die Erweiterung auf nichtlineare Prozessmodelle (Kap. 6.1.2) und Mehrgrößensysteme (Kap. 9). Die Transformation wurde in [44] ausführlich durchgeführt und die Äquivalenz beider Darstellungen gezeigt. Dabei wird ein zusätzlicher Zustand $u(k-1) = u(k) - \Delta u(k)$ eingeführt, um die Zustandsdarstellung mit Δu als Eingangsgröße

anschreiben zu können. Die Zustandsbeschreibung ergibt sich zu

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A} \mathbf{x}(k) + \mathbf{b} \Delta u(k) + \mathbf{z} \xi(k) \\ y &= \mathbf{c}^T \mathbf{x}(k) + v(k)\end{aligned}\quad (5.15)$$

Dabei wurde ein weiteres Rauschsignal $v(k)$ eingeführt, das der Modellierung von Messrauschen dient. Beide Signale treten in der Prädiktion des Ausgangssignals nicht auf, in das sie mit ihrem Erwartungswert von null eingehen [44]. Die prädizierten Ausgangssignale ergeben sich dann zu:

$$\begin{aligned}\tilde{y}(k+j) &= \mathbf{c}^T \mathbf{A}^j \mathbf{x}(k) + \sum_{i=1}^j \mathbf{c}^T \mathbf{A}^{i-1} \mathbf{b} \Delta u(k+j-i) \\ &= \mathbf{F}(j) \mathbf{x}(k) + \sum_{i=1}^j \mathbf{H}(i) \Delta u(k+j-i)\end{aligned}\quad (5.16)$$

Bei diesem Prädiktionsmodell ist eine Messung oder Beobachtung (z.B. mittels Kalman-Filter [44]) des kompletten Zustandsvektors nötig. Durch eine Zusammenfassung der Variablen gemäß Gleichung (5.3) kann die Prädiktionsgleichung kompakt dargestellt werden.

$$\tilde{\mathbf{y}} = \mathbf{F} \mathbf{x}(k) + \mathbf{H} \Delta \mathbf{u} \quad (5.17)$$

Der erste Term in Gleichung (5.17) gibt die freie Prozessantwort an und hängt nur vom momentanen Zustand $\mathbf{x}(k)$ ab. Der zweite Term beinhaltet den partikulären Anteil der Prädiktion, wodurch der Einfluss des zukünftigen Stellverlaufs wiedergespiegelt wird. Die Matrizen $\mathbf{F}$ und $\mathbf{H}$ ergeben sich zu

$$\mathbf{F} = \begin{bmatrix} \mathbf{c}^T \mathbf{A}^{N_1} \\ \vdots \\ \mathbf{c}^T \mathbf{A}^{N_2} \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} h(N_1) & \dots & h(N_1 - N_u) \\ \vdots & \ddots & \vdots \\ h(N_2) & \dots & h(N_2 - N_u) \end{bmatrix} \quad (5.18)$$

$$h(i) = \begin{cases} \mathbf{c}^T \mathbf{A}^{i-1} \mathbf{b} & : i \geq 1 \\ 0 & : i < 1 \end{cases} \quad (5.19)$$

Als Gütefunktional wird beim GPC-Verfahren eine quadratische Funktion mit den diagonalen Gewichtungsmatrizen $\mathbf{\Gamma}$ und $\mathbf{\Lambda}$ angesetzt. Der Vektor $\mathbf{r}$ stellt die Referenztrajektorie im Prädiktionshorizont dar.

$$J(\Delta \mathbf{u}) = [\tilde{\mathbf{y}} - \mathbf{r}]^T \mathbf{\Gamma} [\tilde{\mathbf{y}} - \mathbf{r}] + \Delta \mathbf{u}^T \mathbf{\Lambda} \Delta \mathbf{u} \quad (5.20)$$

Die Minimierung kann durch Nullsetzen der Ableitung $dJ/d\Delta \mathbf{u}$ erfolgen und liefert den optimalen Stellgrößenvektor.

$$\Delta \mathbf{u} = [\mathbf{H}^T \mathbf{\Gamma} \mathbf{H} + \mathbf{\Lambda}]^{-1} \mathbf{H}^T \mathbf{\Gamma} (\mathbf{r} - \mathbf{F} \mathbf{x}(k)) \quad (5.21)$$

Die Stellgrößenberechnung in Gleichung (5.21) entspricht einem linearen Zustandsregler, der jedoch auch zukünftige Sollwerte berücksichtigt. Bei Berücksichtigung von Begrenzungen muss zur Berechnung von $\Delta \mathbf{u}$ auf das Verfahren der quadratischen Programmierung zurückgegriffen werden, das jedoch als Standard-Programmbibliothek zur Verfügung

steht [48, 86]. Es existieren zahlreiche Abwandlungen des ursprünglichen GPC-Verfahrens. So wurde die GPC-Regelung in [91] auf Mehrgrößensysteme ausgedehnt und ein Basisfunktionsansatz für den Verlauf der Stellgröße eingeführt. Unter dem Titel *Multivariable Continuous Generalized Predictive Control* wurde in [67] ein GPC-Verfahren auf Basis zeitkontinuierlicher Prozessmodelle präsentiert. Einen guten Überblick über verschiedene GPC-Verfahren liefern [6, 89].

5.3 Nichtlineare prädiktive Regelungen

Zu Beginn der Entwicklung prädiktiver Regelungsverfahren lag das Hauptinteresse auf linearen Mehrgrößensystemen mit Begrenzungen. Inzwischen steht die prädiktive Regelung auf Basis nichtlinearer Modelle im Mittelpunkt des Forschungsinteresses. Das liegt zum einen daran, dass der Wunsch besteht, Systeme näher an den technischen Grenzen zu betreiben um die Effizienz und Ausbeute zu erhöhen, zum anderen auch daran, dass durch neue Technologien zusätzliche Herausforderungen für die Regelungstechnik entstehen. Beides erfordert eine genauere, und damit meist nichtlineare Modellbildung. Da allerdings vom Übergang von linearen auf nichtlineare Modelle die Vielfalt der Strukturen in einem Maße zunimmt, dass eine einfache Einteilung nicht mehr möglich ist, werden hier nur grobe Einteilungen durchgeführt. Die Basis stellt das nichtlineare SISO-Zustandsraummodell mit den stetigen nichtlinearen Funktionen $\mathbf{f}$ und g dar.

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{f}(\mathbf{x}(k), u(k)) \\ y(k) &= g(\mathbf{x}(k))\end{aligned}\tag{5.22}$$

Die erste Möglichkeit zum Aufbau eines Prädiktionsmodells stellt die Linearisierung von Gleichung (5.22) um den jeweils aktuellen Zustand dar. Dadurch ergibt sich ein, zu jedem Abtastpunkt unterschiedliches, lineares zeitinvariantes System. Die Prädiktion erfolgt mit einem konstanten Modell und kann daher auch nur kleine Abweichungen vom aktuellen Arbeitspunkt in guter Qualität vorhersagen. In [70] wird eine nichtlineare Weiterentwicklung des DMC-Verfahrens vorgestellt. Es basiert auf der Modellbeschreibung in Gleichung (5.22) und spaltet die Prädiktion in die freie Prozessantwort und einen partikulären Anteil auf. Die freie Antwort wird aus dem nichtlinearen Modell berechnet, während die partikuläre Antwort aus dem, um den aktuellen Arbeitspunkt, linearisierten Prozessmodell bestimmt wird. Die dynamische Matrix $\mathbf{H}$ wird nun zu jedem Abtastschritt neu aufgebaut. Es wird die gleiche Gütefunktion wie beim linearen DMC-Algorithmus verwendet, so dass sich auch bei ihrer Optimierung keine Änderungen ergeben. Insbesondere ist ohne Begrenzungen eine analytische Lösung möglich, und bei Berücksichtigung von Begrenzungen kann auf die quadratische Programmierung zurückgegriffen werden.

In [21, 62] wird als Prozessmodell ein nichtlineares Black-Box-Modell auf Basis eines *Multilayer Perceptron Netzes* verwendet und zur Prädiktion um den jeweils aktuellen Arbeitspunkt linearisiert. Es eignet sich insbesondere für unbekannte Prozesse, da nur sehr wenig Vorwissen zur Identifikation benötigt wird. In [58] wird für Wiener- und Hammersteinmodelle eine nichtlineare prädiktive Regelung entwickelt. Die nichtlinearen Blöcke werden als Polynome, die lineare Dynamik als endliche FIR-Modelle dargestellt. Dieses Verfahren

kommt ohne Linearisierung aus, setzt aber stabile Regelstrecken voraus. Das Verfahren wird erfolgreich auf eine Neutralisationsstrecke und einen Rührkesselreaktor angewandt. Schließlich wird in [39] eine prädiktive Regelung auf Basis von Fuzzy-Modellen vorgestellt.

Das allgemeine nichtlineare Zustandsraummodell (5.22) wird in [61, 78, 79, 85] behandelt. Dies stellt ohne Zweifel die genaueste Prozessmodellierung dar, benötigt jedoch, bereits ohne Berücksichtigung von Beschränkungen, nichtlineare numerische Verfahren zur Optimierung der Gütefunktion. Dadurch ergibt sich ein wesentlich erhöhter Rechenaufwand, der die Anwendung auf *schnelle* technische Systeme schwierig macht. Außerdem sind wegen des nichtlinearen Charakters lokale Minima der Gütefunktion möglich, zu deren Vermeidung zusätzliche Maßnahmen ergriffen werden müssen. Die Unterschiede der nichtlinearen MPC-Verfahren zeigen sich hauptsächlich in den Stabilitätsnachweisen des geschlossenen Regelkreises und den verwendeten Optimierungsalgorithmen. In [38] wird ein Stabilitätsbeweis basierend auf zeitkontinuierlichen nichtlinearen Zustandsraummodellen hergeleitet. Einen Überblick über aktuelle nichtlineare prädiktive Regelungsverfahren gibt [3].

Die Verfahren, die eine Arbeitspunktlinearisierung zur Prädiktion einsetzen, besitzen bei großen Arbeitsbereichen des Prozesses nur mäßige Genauigkeit, da sie die nichtlinearen Effekte in der Zukunft nicht abbilden. Die Lösung des Optimierungsproblems kann allerdings analytisch oder mit dem effizienten Verfahren der quadratischen Programmierung gelöst werden. Ferner existiert als Lösung nur genau ein Minimum. Die Verfahren ohne Linearisierung besitzen zwar höhere Genauigkeit, jedoch ist deren Online-Berechnung durch den Einsatz nichtlinearer Optimierungsalgorithmen viel aufwändiger. Außerdem ist die Existenz genau eines Minimums nicht ohne weiteres gesichert.

Das in Kap. 6 vorgestellte prädiktive Regelungsverfahren für nichtlineare Prozesse verbessert die Genauigkeit des Prädiktionsmodells durch Linearisierung um die Referenztrajektorie. Dadurch werden nichtlineare Effekte in der Zukunft berücksichtigt. Die Berechnung der optimalen Stellgröße ist aber nach wie vor analytisch (ohne Begrenzungen) bzw. mit dem Verfahren der quadratischen Programmierung (mit Begrenzungen) lösbar. Es wird somit die Genauigkeit des Prädiktionsmodells gesteigert, was sich vor allem bei transienten Vorgängen bemerkbar macht, und gleichzeitig die effiziente Berechenbarkeit gewahrt.

6 Prädiktive Regelung mit nichtlinearen Zustandsmodellen

Die Berücksichtigung nichtlinearer Prozesseigenschaften bei der Modellierung gewinnt zunehmend an Bedeutung, da hierdurch genauere Prozessmodelle entstehen, welche ein besseres Abbild der Realität darstellen. Dies führt zu verbesserten Regelungen, die den geregelten Prozess näher und exakter an seinen zulässigen oder gewünschten Grenzwerten betreiben können. Die meisten industriellen Prozesse weisen nichtlinearen Charakter auf. Daher sind lineare Prozessmodelle für präzise Regelungsaufgaben häufig nicht ausreichend genau. Besonders in regelungstechnischen Aufgabenstellungen, in welchen der nichtlineare Prozess in einem weiten Bereich des Zustandsraumes betrieben wird, zeigt sich der Vorteil von nichtlinearen Modellen. Allerdings muss für die Verbesserung der Genauigkeit ein erhöhter Rechenaufwand in Kauf genommen werden. Einen Mittelweg zwischen Präzision und Anforderung an die benötigte Rechenleistung beschreitet die Linearisierung des Prozessmodells entlang einer Referenztrajektorie, so dass ein **lineares**, aber **zeitvariantes Prädiktionsmodell** entsteht. Dadurch kann einerseits nichtlineares Verhalten berücksichtigt werden, andererseits stellt die geforderte Rechenleistung keine unüberwindbare Hürde für eine Regelung in Echtzeit dar.

Die betrachteten dynamischen Prozessmodelle zeichnen sich durch eine isolierte statische Nichtlinearität aus. Genaue Definitionen zur betrachteten Systemklasse finden sich in Kap. 3.3.1. Da bei der nichtlinearen Modellbildung in Kap. 3.3 von einem zeitkontinuierlichen Prozessmodell ausgegangen wird, muss zuerst eine Diskretisierung erfolgen. Dies kann für den linearen Anteil gemäß Kap. 2.2 erfolgen, die Nichtlinearität wird unverändert auch im diskretisierten Modell benutzt. Zur Diskretisierung stehen Standardwerkzeuge in Softwarepaketen zur Verfügung [4].

Zunächst werden nur Eingrößensysteme (SISO) untersucht, eine Erweiterung auf den Mehrgrößenfall (MIMO) erfolgt in Kap. 9. Die Vektoren und Matrizen der kontinuierlichen Streckenbeschreibung werden mit dem Index c gekennzeichnet. Die zeitkontinuierliche Streckenbeschreibung lautet:

$$\dot{\mathbf{x}}_c = \mathbf{A}_c \mathbf{x}_c + \mathbf{b}_c u_c + \mathbf{k}_c \mathcal{NL}(\mathbf{x}_c) \quad (6.1)$$

$$y_c = \mathbf{c}_c^T \mathbf{x}_c + d_c u_c \quad (6.2)$$

Nach der Diskretisierung ergibt sich die zeitdiskrete Zustandsdarstellung mit isolierter Nichtlinearität in Gleichung (6.3).

$$\bar{\mathbf{x}}(k+1) = \bar{\mathbf{A}} \cdot \bar{\mathbf{x}}(k) + \bar{\mathbf{b}} \cdot u(k) + \bar{\mathbf{k}} \cdot \mathcal{NL}(\bar{\mathbf{x}}) \quad (6.3)$$

$$y(k) = \bar{\mathbf{c}}^T \cdot \bar{\mathbf{x}}(k) + \bar{d} \cdot u(k)$$

Die Wirkung der nichtlinearen Streckendynamik ist durch einen zusätzlichen Eingang beschrieben, der über den Vektor $\bar{\mathbf{k}}$ eingekoppelt wird. Das Argument der Nichtlinearität $\mathcal{NL}$ in Gleichung (6.3) kann eine beliebige Untermenge des Zustandsvektors $\bar{\mathbf{x}}$ sein. Darin sind

auch gewichtete Linearkombinationen der einzelnen Prozesszustände enthalten, was einer Abhängigkeit der Nichtlinearität vom Ausgangssignal entspricht.

Um den Prozess mit der Eingangsgröße Δu darzustellen, wird eine Erweiterung um einen Zustand $u(k-1) = u(k) - \Delta u(k)$ vorgenommen. Dies erleichtert die spätere Gewichtung der Stelländerung im Gütefunktional der prädiktiven Regelung. Das erweiterte Modell ist in Gleichung (6.4) angegeben.

$$\underbrace{\begin{bmatrix} \bar{\mathbf{x}}(k+1) \\ u(k) \end{bmatrix}}_{\mathbf{x}(k+1)} = \underbrace{\begin{bmatrix} \bar{\mathbf{A}} & \bar{\mathbf{b}} \\ \mathbf{0} & 1 \end{bmatrix}}_{\mathbf{A}} \cdot \underbrace{\begin{bmatrix} \bar{\mathbf{x}}(k) \\ u(k-1) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} \bar{\mathbf{b}} \\ 1 \end{bmatrix}}_{\mathbf{b}} \cdot \Delta u(k) + \underbrace{\begin{bmatrix} \bar{\mathbf{k}} \\ 0 \end{bmatrix}}_{\mathbf{k}} \cdot \mathcal{NL}(\mathbf{x}(k)) \quad (6.4)$$

$$y(k) = \underbrace{\begin{bmatrix} \bar{\mathbf{c}}^T & \bar{d} \end{bmatrix}}_{\mathbf{c}^T} \cdot \begin{bmatrix} \bar{\mathbf{x}}(k) \\ u(k-1) \end{bmatrix} + \underbrace{\bar{d}}_d \cdot \Delta u(k) \quad (6.5)$$

Wenn das ursprüngliche Modell n Zustände hatte, so erhöht sich die Zahl der Zustände in Gleichung (6.4) auf $N = n + 1$. Für den Entwurf des prädiktiven Regelgesetzes wird im Folgenden der erweiterte Prozess angesetzt.

$$\mathbf{x}(k+1) = \mathbf{A} \cdot \mathbf{x}(k) + \mathbf{b} \cdot \Delta u(k) + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}(k)) \quad (6.6)$$

$$y(k) = \mathbf{c}^T \cdot \mathbf{x}(k) + d \cdot \Delta u(k) \quad (6.7)$$

Notwendigkeit der Linearisierung

Ein prädiktives Regelgesetz erfordert die Vorhersage des zukünftigen Prozessverhaltens. Wird dafür Gleichung (6.6) verwendet, ergibt sich folgendes Gleichungssystem:

$$\tilde{\mathbf{x}}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{b}\Delta u(k) + \mathbf{k}\mathcal{NL}(\mathbf{x}(k)) \quad (6.8)$$

$$\begin{aligned} \tilde{\mathbf{x}}(k+2) &= \mathbf{A}^2\mathbf{x}(k) + \mathbf{A}\mathbf{b}\Delta u(k) + \mathbf{b}\Delta u(k+1) + \mathbf{A}\mathbf{k}\mathcal{NL}(\mathbf{x}(k)) + \\ &+ \mathbf{k}\mathcal{NL}(\mathbf{A}\mathbf{x}(k) + \mathbf{b}\Delta u(k) + d\Delta u(k+1) + \mathbf{k}\mathcal{NL}(\mathbf{x}(k))) \end{aligned}$$

Darin ist die gesuchte Stellgröße $\Delta u(k)$ als Argument in der nichtlinearen Funktion $\mathcal{NL}$ enthalten. Es ist im Allgemeinen nicht möglich, daraus ein nach der Stellgröße aufgelöstes Gleichungssystem anzugeben. Somit kann die Stellgröße nur mit rechenaufwändigen numerischen Algorithmen ermittelt werden. Diese stellen einerseits hohe Ansprüche an die erforderliche Rechenleistung und können andererseits wegen evtl. mehrdeutiger Lösung nicht immer ein eindeutiges Ergebnis liefern. Daher wird hier der Ansatz verfolgt, Gleichung (6.6) erst zu linearisieren, bevor diese dann zur Prädiktion herangezogen wird. Dies entspricht den Forderungen aus Kap. 5.3 nach Berücksichtigung der nichtlinearen Effekte unter Beibehaltung der effizienten Lösbarkeit.

Zunächst wird die Abhängigkeit der Nichtlinearität $\mathcal{NL}$ vom Prozessausgang behandelt. In Kap. 6.2 wird das Verfahren auf die Abhängigkeit der Nichtlinearität auf beliebige innere Zustände erweitert. Diese Unterscheidung ist sinnvoll, da sich Unterschiede in der Durchführung der Linearisierung ergeben.

6.1 Abhängigkeit der Nichtlinearität vom Prozessausgang

Zunächst werden Prozesse betrachtet, deren **Ausgang** über einen nichtlinearen Zusammenhang auf den Zustandsvektor einwirkt, wie auch aus Bild 6.1 ersichtlich ist. Ein technisch-physikalischer Prozess dieser Art ist z.B. ein mechanisches Antriebssystem mit zwei rotierenden Körpern, dessen Ausgangssignal (Drehzahl eines Körpers) aufgrund von Reibung die Änderung eines Systemzustandes (resultierendes Beschleunigungsmoment) beeinflusst. Die Zustandsbeschreibung derartiger Systeme ergibt sich zu:

$$\mathbf{x}(k+1) = \mathbf{A} \cdot \mathbf{x}(k) + \mathbf{b} \cdot \Delta u(k) + \mathbf{k} \cdot \mathcal{NL}(y(k)) \quad (6.9)$$

$$y(k) = \mathbf{c}^T \cdot \mathbf{x}(k) + d \cdot \Delta u(k) \quad (6.10)$$

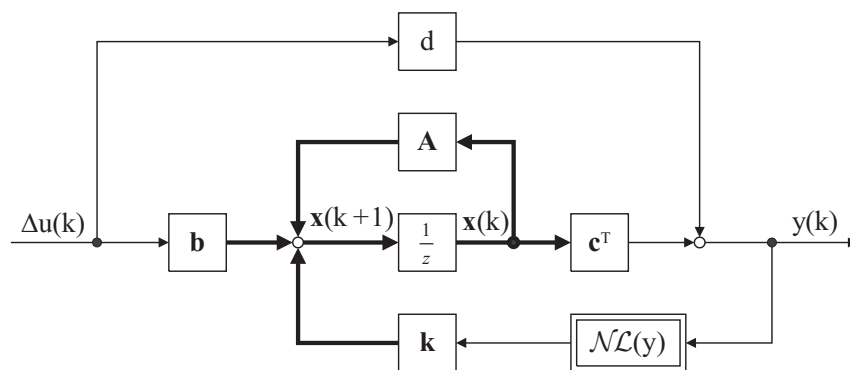


Bild 6.1: Signalflussplan mit nichtlinearer Abhängigkeit vom Prozessausgang

Um Rechenzeit einzusparen, kann eine Linearisierung nullter Ordnung durchgeführt werden (Kap. 6.1.1). Der eigentliche Kern der Arbeit stellt jedoch eine Linearisierung erster Ordnung entlang der Referenztrajektorie dar (Kap. 6.1.2). Dadurch wird die Berücksichtigung des nichtlinearen Charakters auch bei der Prädiktion möglich.

6.1.1 Linearisierung nullter Ordnung

Das problematische Auftreten der Stellgrößenänderung im Argument der Nichtlinearität bei der Prädiktion kann durch Linearisierung umgangen werden. Dabei wird implizit angenommen, dass der Prozessausgang, bedingt durch die Regelung, der Sollwerttrajektorie eng genug folgen wird. $\mathcal{NL}$ wird durch eine Linearisierung nullter Ordnung um die Referenztrajektorie $r(k+j)$ angenähert.

$$\mathcal{NL}(y(k+j)) \approx \mathcal{NL}(r(k+j)) = v(k+j) \quad (6.11)$$

Folglich kann $r(k+j)$ anstelle von $y(k+j)$ als Argument in die nichtlineare Funktion eingesetzt werden, wodurch deren Abhängigkeit von Δu verschwindet. Die Nichtlinearität hat nach der Linearisierung nullter Ordnung um die Referenztrajektorie die Wirkung einer

Störgröße, die jedoch in der Zukunft bekannt ist. Somit kann eine Prädiktion des Prozessverhaltens

$$\begin{aligned}
\tilde{\mathbf{x}}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{b}\Delta u(k) + \mathbf{k}v(k) \\
\tilde{\mathbf{x}}(k+2) &= \mathbf{A}^2\mathbf{x}(k) + \mathbf{A}\mathbf{b}\Delta u(k) + \mathbf{b}\Delta u(k+1) + \mathbf{A}\mathbf{k}v(k) + \mathbf{k}v(k+1) \\
\tilde{\mathbf{x}}(k+3) &= \mathbf{A}^3\mathbf{x}(k) + \mathbf{A}^2\mathbf{b}\Delta u(k) + \mathbf{A}\mathbf{b}\Delta u(k+1) + \mathbf{b}\Delta u(k+2) + \\
&\quad + \mathbf{A}^2\mathbf{k}v(k) + \mathbf{A}\mathbf{k}v(k+1) + \mathbf{k}v(k+2) \\
&\vdots \\
\tilde{\mathbf{x}}(k+j) &= \mathbf{A}^j\mathbf{x}(k) + \sum_{i=0}^{j-1} \mathbf{A}^{j-i-1}\mathbf{b}\Delta u(k+i) + \sum_{i=0}^{j-1} \mathbf{A}^{j-i-1}\mathbf{k}v(k+i) \quad (6.12)
\end{aligned}$$

durchgeführt werden, bei der eine *lineare* Abhängigkeit von der Stellgröße besteht. Eine Umrechnung des prädizierten Zustandsvektors auf den interessierenden Ausgangswert nimmt Gleichung (6.7) vor, woraus sich allgemein für einen beliebigen zukünftigen Zeitpunkt $k+j$ ergibt:

$$\tilde{y}(k+j) = \mathbf{c}^T \mathbf{A}^j \mathbf{x}(k) + \sum_{i=0}^{j-1} \mathbf{c}^T \mathbf{A}^{j-i-1} \mathbf{b} \Delta u(k+i) + \sum_{i=0}^{j-1} \mathbf{c}^T \mathbf{A}^{j-i-1} \mathbf{k} v(k+i) + d \Delta u(k+j) \quad (6.13)$$

Der Vorhersagezeitraum erstreckt sich von $k+N_1$ bis $k+N_2$. Es werden also $N_2 - N_1 + 1$ Ausgangswerte $\tilde{y}(k+j)$ in diesem Zeitintervall benötigt. Deswegen muss Gleichung (6.13) für alle $N_1 \leq j \leq N_2$ ausgewertet werden. Wenn die prädizierten Größen im Ausgangsvektor

$$\tilde{\mathbf{y}} = [\tilde{y}(k+N_1), \tilde{y}(k+N_1+1), \dots, \tilde{y}(k+N_2)]^T \in \mathbb{R}^{N_2-N_1+1} \quad (6.14)$$

zusammengefasst werden, kann die Berechnung der zukünftigen Ausgangswerte in einer einzigen vektoriellen Gleichung erfolgen.

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (6.15)$$

Die dazu notwendige Bildung der Prädiktionsmatrizen in Gleichung (6.15) geht aus Gleichung (6.13) hervor. Die Matrix

$$\mathbf{F} = \begin{bmatrix} \mathbf{c}^T \mathbf{A}^{N_1} \\ \mathbf{c}^T \mathbf{A}^{N_1+1} \\ \vdots \\ \mathbf{c}^T \mathbf{A}^{N_2} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N} \quad (6.16)$$

beschreibt die freie Systemantwort des linearen Prozessanteils, also die Weiterentwicklung des Ausgangsverlaufes aufgrund des Anfangszustandes $\mathbf{x}(k)$. Durch die Matrix

$$\mathbf{H} = \begin{bmatrix} h_{N_1-1} & h_{N_1-2} & \dots & h_{N_1-N_u-1} \\ \vdots & \ddots & \ddots & \vdots \\ h_{N_2-1} & h_{N_2-2} & \dots & h_{N_2-N_u-1} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_u+1} \quad (6.17)$$

mit den Elementen

$$h_p = \begin{cases} \mathbf{c}^T \mathbf{A}^p \mathbf{b} & \text{für } p \geq 0 \\ d & \text{für } p = -1 \\ 0 & \text{für } p < -1 \end{cases} \quad (6.18)$$

wird der Einfluss der zukünftigen Stellgrößen berücksichtigt. Falls kein Durchgriff existiert ($d = 0$) und der Stellhorizont $N_u = N_2$ gewählt wurde, besteht die letzte Spalte der Matrix $\mathbf{H}$ in Gleichung (6.17) nur aus Nullen und kann gestrichen werden. Für diesen Fall gilt:

$$\mathbf{H} = \begin{bmatrix} h_{N_1-1} & h_{N_1-2} & \cdots & h_{N_1-N_u} \\ \vdots & \ddots & \ddots & \vdots \\ h_{N_2-1} & h_{N_2-2} & \cdots & h_{N_2-N_u} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_u} \quad (6.19)$$

mit den Elementen:

$$h_p = \begin{cases} \mathbf{c}^T \mathbf{A}^p \mathbf{b} & \text{für } p \geq 0 \\ 0 & \text{für } p < 0 \end{cases} \quad (6.20)$$

Da der Stellhorizont N_u normalerweise wesentlich kleiner gewählt wird als der obere Prädiktionshorizont N_2 , tritt dieser Fall nicht sehr häufig auf. Deshalb wird, wenn nicht ausdrücklich etwas anderes vereinbart wird, von einem Stellhorizont $N_u < N_2$ ausgegangen, so dass Gleichung (6.17) verwendet werden kann.

Die Auswirkung der Nichtlinearität auf die Systemantwort ist in der Matrix

$$\mathbf{G} = \begin{bmatrix} g_{N_1-1} & g_{N_1-2} & \cdots & g_{N_1-N_2} \\ \vdots & \ddots & \ddots & \vdots \\ g_{N_2-1} & g_{N_2-2} & \cdots & g_{N_2-N_2} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_2} \quad (6.21)$$

enthalten. Ähnlich wie bei der $\mathbf{H}$ -Matrix setzen sich die einzelnen Elemente folgendermaßen zusammen:

$$g_p = \begin{cases} \mathbf{c}^T \mathbf{A}^p \mathbf{k} & \text{für } p \geq 0 \\ 0 & \text{für } p < 0 \end{cases} \quad (6.22)$$

Die zukünftige Stellgrößenfolge berechnet sich aus der Minimierung der Gütefunktion (5.4) und wird durch den Vektor $\Delta \mathbf{u}$ beschrieben, die Auswirkung der Nichtlinearität ist im Vektor $\mathbf{v}$ enthalten. Da die Nichtlinearität nicht direkt auf das Ausgangssignal y wirkt, leistet $v(k + N_2)$ keinen Beitrag zur Ausgangsgröße $y(k + N_2)$. Es genügt daher, $v(k + j)$ im Bereich $0 \leq j \leq N_2 - 1$ zu berücksichtigen.

$$\Delta \mathbf{u} = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_u)]^T \in \mathbb{R}^{N_u+1} \quad (6.23)$$

$$\mathbf{v} = [v(k), v(k+1), \dots, v(k+N_2-1)]^T \in \mathbb{R}^{N_2} \quad (6.24)$$

mit

$$v(k+j) = \mathcal{NL}(r(k+j)) \quad (6.25)$$

Diese Linearisierung hat die Eigenschaft, dass die Matrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ nicht zeitvariant sind und daher in einer Initialisierungsphase des späteren Regelalgorithmus offline berechnet werden können.

6.1.2 Linearisierung erster Ordnung

Zur genaueren Nachbildung der Abweichung des Ausgangssignals von der Referenztrajektorie, wird nun eine Linearisierung erster Ordnung von Gleichung (6.6) durchgeführt. Die Linearisierung um die Referenztrajektorie erfolgt hier durch Entwicklung der Nichtlinearität in eine Taylorreihe, die nach dem linearen Term abgebrochen wird.

$$\begin{aligned}
 \mathcal{NL}(y(k)) &\approx \mathcal{NL}(r(k)) + \left. \frac{d\mathcal{NL}}{dy} \right|_{y=r(k)} \cdot (y(k) - r(k)) = \\
 &= \mathcal{NL}(r(k)) + \underbrace{\left. \frac{d\mathcal{NL}}{dy} \right|_{y=r(k)}}_{\mathcal{NL}_y(r(k))} \cdot (\mathbf{c}^T \mathbf{x}(k) + d \cdot \Delta u(k) - r(k)) = \\
 &= \mathcal{NL}(r(k)) + \mathcal{NL}_y(r(k)) \cdot (\mathbf{c}^T \mathbf{x}(k) + d \cdot \Delta u(k) - r(k))
 \end{aligned} \tag{6.26}$$

Die Taylorentwicklung wird nun in die Zustandsgleichung (6.6) eingesetzt. Es ergibt sich daraus folgende Zustandsprädiktion, die wiederum eine lineare Abhängigkeit von der Stellgrößenänderung $\Delta \mathbf{u}$ aufweist:

$$\begin{aligned}
 \tilde{\mathbf{x}}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{b}\Delta u(k) + \mathbf{k}\mathcal{NL}(y(k)) \approx \\
 &\approx \underbrace{[\mathbf{A} + \mathbf{k}\mathbf{c}^T \mathcal{NL}_y(r(k))]}_{\mathbf{A}(k)} \mathbf{x}(k) + \underbrace{[\mathbf{b} + \mathbf{k} d \mathcal{NL}_y(r(k))]}_{\mathbf{b}(k)} \Delta u(k) + \\
 &\quad + \mathbf{k} \underbrace{[\mathcal{NL}(r(k)) - r(k)\mathcal{NL}_y(r(k))]}_{v(k)}
 \end{aligned} \tag{6.27}$$

$$\begin{aligned}
 \tilde{\mathbf{x}}(k+2) &= \mathbf{A}(k+1)\mathbf{A}(k)\mathbf{x}(k) + \\
 &\quad + \mathbf{A}(k+1)\mathbf{b}(k)\Delta u(k) + \mathbf{b}(k+1)\Delta u(k+1) + \\
 &\quad + \mathbf{A}(k+1)\mathbf{k}v(k) + \mathbf{k}v(k+1)
 \end{aligned} \tag{6.28}$$

$$\begin{aligned}
 \tilde{\mathbf{x}}(k+3) &= \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{A}(k)\mathbf{x}(k) + \\
 &\quad + \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{b}(k)\Delta u(k) + \\
 &\quad + \mathbf{A}(k+2)\mathbf{b}(k+1)\Delta u(k+1) + \mathbf{b}(k+2)\Delta u(k+2) + \\
 &\quad + \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{k}v(k) + \mathbf{A}(k+2)\mathbf{k}v(k+1) + \mathbf{k}v(k+2)
 \end{aligned} \tag{6.29}$$

$\vdots$

$$\begin{aligned}
 \tilde{\mathbf{x}}(k+j) &= \left[\prod_{n=j-1}^0 \mathbf{A}(k+n) \right] \cdot \mathbf{x}(k) + \\
 &\quad + \sum_{i=0}^{j-2} \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \mathbf{b}(k+i)\Delta u(k+i) + \\
 &\quad + \mathbf{b}(k+j-1)\Delta u(k+j-1) +
 \end{aligned} \tag{6.30}$$

$$+ \sum_{i=0}^{j-2} \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \cdot \mathbf{k} \cdot v(k+i) + \mathbf{k} \cdot v(k+j-1)$$

Als allgemeine Berechnungsvorschrift für das Verhalten des Ausgangssignals y in j Zeitschritten folgt aus den Gleichungen (6.30) und (6.7):

$$\begin{aligned} \tilde{y}(k+j) &= \mathbf{c}^T \left[\prod_{n=j-1}^0 \mathbf{A}(k+n) \right] \cdot \mathbf{x}(k) + \\ &+ \sum_{i=0}^{j-2} \mathbf{c}^T \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \mathbf{b}(k+i) \Delta u(k+i) + \\ &+ \mathbf{c}^T \mathbf{b}(k+j-1) \Delta u(k+j-1) + d \Delta u(k+j) + \\ &+ \sum_{i=0}^{j-2} \mathbf{c}^T \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \cdot \mathbf{k} \cdot v(k+i) + \mathbf{c}^T \mathbf{k} \cdot v(k+j-1) \end{aligned} \quad (6.31)$$

Die Verwendung der Approximation (6.26) führt auf ein **lineares zeitvariantes Prädiktionsmodell** (LTV). Wie aus Gleichung (6.27) hervorgeht, gilt für die jetzt zeitvarianten Zustandsmatrizen:

$$\mathbf{A}(k+j) = \mathbf{A} + \mathbf{k} \mathbf{c}^T \mathcal{N}_{\mathcal{L}_y}(r(k+j)) \quad (6.32)$$

$$\mathbf{b}(k+j) = \mathbf{b} + \mathbf{k} d \mathcal{N}_{\mathcal{L}_y}(r(k+j)) \quad (6.33)$$

$$v(k+j) = \mathcal{N}_{\mathcal{L}}(r(k+j)) - r(k+j) \mathcal{N}_{\mathcal{L}_y}(r(k+j)) \quad (6.34)$$

Es ist wichtig zu erkennen, dass die Zeitvarianz der Zustandsmatrizen in der Zukunft bekannt ist. $\mathbf{A}$, $\mathbf{b}$ und v hängen nur von der Referenztrajektorie r ab, die innerhalb des Prädiktionshorizonts bekannt ist. Gleichung (6.31) muss für alle Zeitpunkte innerhalb des Prädiktionszeitraumes ausgewertet werden, d.h. für $N_1 \leq j \leq N_2$. Auch hier wird das prädizierte Ausgangssignal zu jedem Zeitpunkt im Vektor $\tilde{\mathbf{y}}$ zusammengefasst. Somit ergibt sich formal die selbe Gleichung wie in Kap. 6.1.1 für die Vorhersage des Systemverhaltens.

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (6.35)$$

Allerdings ändert sich die Bildung der Prädiktionsmatrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$, deren Bedeutung bleibt dagegen unverändert. Die zeitvariante Eigenschaft wird durch das LTV-Prädiktionsmodell eingebracht, das in jedem Abtastschritt sollwertabhängig neu berechnet werden muss. Deswegen können die Prädiktionsmatrizen nicht mehr vorab aufgebaut werden, sondern müssen zu jedem Abtastzeitpunkt neu bestimmt werden, weswegen ein bedeutend höherer Rechenaufwand entsteht. Die Bildungsvorschrift der Prädiktionsmatrizen lautet:

$$\mathbf{F} = \begin{bmatrix} \mathbf{c}^T \prod_{n=N_1-1}^0 \mathbf{A}(k+n) \\ \mathbf{c}^T \prod_{n=N_1}^0 \mathbf{A}(k+n) \\ \vdots \\ \mathbf{c}^T \prod_{n=N_2-1}^0 \mathbf{A}(k+n) \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N} \quad (6.36)$$

$$\mathbf{H} = \begin{bmatrix} h_{N_1-1, N_1-1} & \cdots & h_{N_1-1, N_1-N_u-1} \\ \vdots & & \vdots \\ h_{N_2-1, N_2-1} & \cdots & h_{N_2-1, N_2-N_u-1} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_u+1} \quad (6.37)$$

mit

$$h_{p,q} = \begin{cases} \mathbf{c}^T \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{b}(k+p-q) & \text{für } q > 0 \\ \mathbf{c}^T \mathbf{b}(k+p-q) & \text{für } q = 0 \\ d & \text{für } q = -1 \\ 0 & \text{für } q < -1 \end{cases} \quad (6.38)$$

$$\mathbf{G} = \begin{bmatrix} g_{N_1-1, N_1-1} & \cdots & g_{N_1-1, N_1-N_2} \\ \vdots & & \vdots \\ g_{N_2-1, N_2-1} & \cdots & g_{N_2-1, N_2-N_2} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_2} \quad (6.39)$$

mit

$$g_{p,q} = \begin{cases} \mathbf{c}^T \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{k} & \text{für } q > 0 \\ \mathbf{c}^T \mathbf{k} & \text{für } q = 0 \\ 0 & \text{für } q < 0 \end{cases} \quad (6.40)$$

Bei der Matrix $\mathbf{H}$ ergibt sich für $N_u = N_2$ und $d = 0$ in Gleichung (6.37) die Besonderheit, dass die letzte Spalte nur aus Nullen besteht und somit entfallen kann. Die Stellgrößenfolge und die Einwirkung des nichtlinearen Prozesscharakters werden in den Vektoren $\Delta \mathbf{u}$ und $\mathbf{v}$ zusammengefasst.

$$\Delta \mathbf{u} = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_u)]^T \in \mathbb{R}^{N_u+1} \quad (6.41)$$

$$\mathbf{v} = [v(k), v(k+1), \dots, v(k+N_2-1)]^T \in \mathbb{R}^{N_2} \quad (6.42)$$

Durch die Näherung erster Ordnung werden Abweichungen von der Referenztrajektorie im Prädiktionsmodell berücksichtigt. Diese zusätzliche Genauigkeit erkaufte man sich dadurch, dass nun die Prädiktionsmatrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ bei jedem Abtastschritt online neu berechnet werden müssen. Zusätzlich wird neben der Nichtlinearität selbst, auch noch deren erste Ableitung benötigt. Diese kann nach erfolgter Modellbildung analytisch berechnet werden oder vom lernfähigen Zustandsraummodell aus Kap. 3.3 online zur Verfügung gestellt werden. Die online Berechnung der Ableitung wird in Kap. 8.6 behandelt.

6.2 Abhängigkeit der Nichtlinearität vom Zustandsvektor

Nicht jedes technisch relevante System wird eine Abhängigkeit der Nichtlinearität vom Ausgangssignal aufweisen. Deshalb wird nun eine allgemeinere Abhängigkeit der Nichtlinearität von **beliebigen Elementen des Zustandsvektors** zugelassen. Es werden Prozesse betrachtet, die sich durch Gleichung (6.43) beschreiben lassen.

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A} \cdot \mathbf{x}(k) + \mathbf{b} \cdot \Delta u(k) + \mathbf{k} \cdot \mathcal{NL}(\mathbf{x}(k)) \\ y(k) &= \mathbf{c}^T \cdot \mathbf{x}(k) + d \cdot \Delta u(k) \end{aligned} \quad (6.43)$$

Bild 6.2 zeigt den zugehörigen Signalflussplan. Das Argument der nichtlinearen Funktion

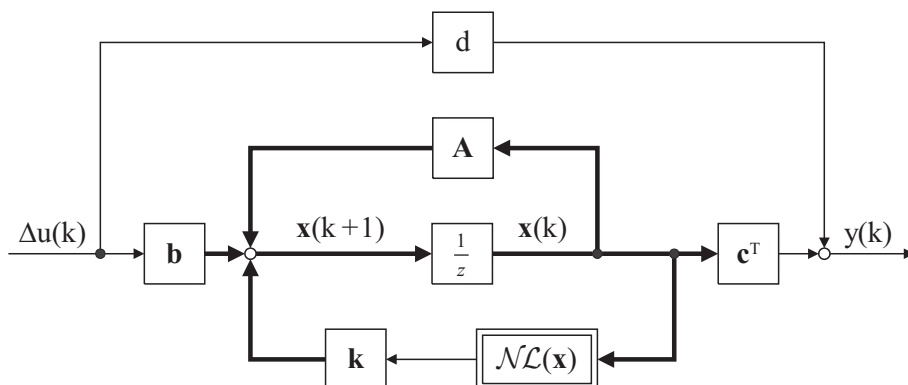


Bild 6.2: Signalflussplan mit nichtlinearer Abhängigkeit vom Zustandsvektor

ist in diesem Fall der Zustandsvektor $\mathbf{x}$. Welche Zustände sich im Einzelnen nichtlinear auf die Dynamik auswirken, beschreibt die Struktur der nichtlinearen Funktion $\mathcal{NL}$. Wenn keine Abhängigkeit der Nichtlinearität vom i -ten Zustand x_i besteht, wird dieser in der Nichtlinearität mit null bewertet. Dadurch kann sowohl eine Abhängigkeit von einzelnen Zuständen, als auch vom gesamten Zustandsvektor bestehen. Als Näherung wird nur die Linearisierung erster Ordnung durchgeführt.

Für die Linearisierung von $\mathcal{NL}(x(k))$ ist eine Referenztrajektorie $\mathbf{x}^{soll}(k+j)$ erforderlich, welche den Sollverlauf der Prozesszustände beschreibt. Bevor auf die Generierung einer solchen Trajektorie eingegangen wird, soll diese zunächst als bekannt angenommen und eine Linearisierung erster Ordnung angesetzt werden. Dazu wird die Nichtlinearität mit N -dimensionalem Eingangsraum in eine Taylorreihe entwickelt, die nach dem linearen Term abgebrochen wird:

$$\begin{aligned}
 \mathcal{NL}(\mathbf{x}(k)) &\approx \mathcal{NL}(\mathbf{x}^{soll}) + \left. \frac{\partial \mathcal{NL}}{\partial x_1} \right|_{\mathbf{x}=\mathbf{x}^{soll}} \cdot (x_1 - x_1^{soll}) \\
 &+ \left. \frac{\partial \mathcal{NL}}{\partial x_2} \right|_{\mathbf{x}=\mathbf{x}^{soll}} \cdot (x_2 - x_2^{soll}) \\
 &\vdots \\
 &+ \left. \frac{\partial \mathcal{NL}}{\partial x_N} \right|_{\mathbf{x}=\mathbf{x}^{soll}} \cdot (x_N - x_N^{soll})
 \end{aligned} \tag{6.44}$$

Die Summe über die Produkte (Ableitung mal Abstand) in der Taylorreihe kann in ein Skalarprodukt zusammengefasst werden. Damit ergibt sich eine kompaktere Schreibweise:

$$\mathcal{NL}(\mathbf{x}) \approx \mathcal{NL}(\mathbf{x}^{soll}) + \underbrace{\left[\frac{\partial \mathcal{NL}(\mathbf{x}^{soll})}{\partial x_1}, \dots, \frac{\partial \mathcal{NL}(\mathbf{x}^{soll})}{\partial x_N} \right]}_{\text{grad}^T(\mathcal{NL})} \cdot \underbrace{\begin{bmatrix} x_1 - x_1^{soll} \\ \vdots \\ x_N - x_N^{soll} \end{bmatrix}}_{\mathbf{x} - \mathbf{x}^{soll}} \tag{6.45}$$

Setzt man Gleichung (6.45) in (6.43) ein, so folgt die linearisierte Prozessbeschreibung, die

als Prädiktionsmodell verwendet wird.

$$\begin{aligned} \mathbf{x}(k+1) = & \underbrace{[\mathbf{A} + \mathbf{k} \operatorname{grad}^T(\mathcal{NL}(\mathbf{x}^{soll}(k)))]}_{\mathbf{A}(k)} \mathbf{x}(k) + \mathbf{b} \Delta u(k) + \\ & \mathbf{k} \underbrace{[\mathcal{NL}(\mathbf{x}^{soll}(k)) - \operatorname{grad}^T(\mathcal{NL}(\mathbf{x}^{soll}(k))) \cdot \mathbf{x}^{soll}(k)]}_{v(k)} \end{aligned} \quad (6.46)$$

Wenn sich nicht alle Zustände auf den Funktionswert der Nichtlinearität auswirken, beinhaltet der Gradient an den entsprechenden Positionen eine Null. Gleichung (6.44) kann als mehrdimensionale Verallgemeinerung von Gleichung (6.26) aufgefasst werden. Nun kann eine Vorhersage des zukünftigen Prozessausgangs bis zum oberen Prädiktionshorizont erfolgen.

$$\begin{aligned} \tilde{y}(k+1) &= \mathbf{c}^T \tilde{\mathbf{x}}(k+1) = \mathbf{c}^T [\mathbf{A}(k) \mathbf{x}(k) + \mathbf{b} \Delta u(k) + \mathbf{k} v(k)] \\ \tilde{y}(k+2) &= \mathbf{c}^T [\mathbf{A}(k+1) \mathbf{A}(k) \mathbf{x}(k) + \mathbf{A}(k+1) \mathbf{b} \Delta u(k) + \mathbf{b} \Delta u(k+1) \\ &\quad + \mathbf{A}(k+1) \mathbf{k} v(k) + \mathbf{k} v(k+1)] \\ &\vdots \\ \tilde{y}(k+j) &= \mathbf{c}^T \left[\prod_{n=j-1}^0 \mathbf{A}(k+n) \right] \cdot \mathbf{x}(k) + \\ &\quad + \sum_{i=0}^{j-2} \mathbf{c}^T \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \cdot \mathbf{b} \cdot \Delta u(k+i) + \\ &\quad + \mathbf{c}^T \mathbf{b} \cdot \Delta u(k+j-1) + d \Delta u(k+j) + \\ &\quad + \sum_{i=0}^{j-2} \mathbf{c}^T \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \cdot \mathbf{k} \cdot v(k+i) + \mathbf{c}^T \mathbf{k} \cdot v(k+j-1) \end{aligned} \quad (6.47)$$

Im Gegensatz zur Prädiktion in Kap. 6.1.2 ist hier der Einkoppelvektor $\mathbf{b}$ zeitinvariant. Das liegt daran, dass hier um einen Zustandssollverlauf statt um einen Ausgangssollverlauf linearisiert wurde. Für einen verschwindenden Durchgriff $d = 0$ sind die Prädiktionen jedoch in beiden Fällen identisch. Lediglich die Struktur der Systemmatrix $\mathbf{A}(k+j)$ und des nichtlinearen Einflusses $v(k+j)$ sind unterschiedlich, wie aus der zeitvarianten Zustandsdarstellung (6.46) des Prädiktionsmodelles hervorgeht.

$$\mathbf{A}(k+j) = \mathbf{A} + \mathbf{k} \operatorname{grad}^T(\mathcal{NL}(\mathbf{x}^{soll}(k+j))) \quad (6.48)$$

$$v(k+j) = \mathcal{NL}(\mathbf{x}^{soll}(k+j)) - \operatorname{grad}^T(\mathcal{NL}(\mathbf{x}^{soll}(k+j))) \cdot \mathbf{x}^{soll}(k+j) \quad (6.49)$$

Die Auswertung der Gleichung (6.47) im gesamten Prädiktionshorizont $N_1 \leq j \leq N_2$ führt wiederum auf die bereits bekannte Prädiktionsgleichung.

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (6.50)$$

Die darin enthaltenen Matrizen werden nach den folgenden Bildungsvorschriften aufgebaut:

$$\mathbf{F} = \begin{bmatrix} \mathbf{c}^T \prod_{n=N_1-1}^0 \mathbf{A}(k+n) \\ \mathbf{c}^T \prod_{n=N_1}^0 \mathbf{A}(k+n) \\ \vdots \\ \mathbf{c}^T \prod_{n=N_2-1}^0 \mathbf{A}(k+n) \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N} \quad (6.51)$$

$$\mathbf{H} = \begin{bmatrix} h_{N_1-1, N_1-1} & \cdots & h_{N_1-1, N_1-N_u-1} \\ \vdots & & \vdots \\ h_{N_2-1, N_2-1} & \cdots & h_{N_2-1, N_2-N_u-1} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_u+1} \quad (6.52)$$

$$\text{mit } h_{p,q} = \begin{cases} \mathbf{c}^T \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{b} & \text{für } q > 0 \\ \mathbf{c}^T \mathbf{b} & \text{für } q = 0 \\ d & \text{für } q = -1 \\ 0 & \text{für } q < -1 \end{cases} \quad (6.53)$$

Nach dem selben Schema setzt sich die $\mathbf{G}$ -Matrix zusammen. An die Stelle des Vektors $\mathbf{b}$ tritt der Vektor $\mathbf{k}$.

$$\mathbf{G} = \begin{bmatrix} g_{N_1-1, N_1-1} & \cdots & g_{N_1-1, N_1-N_2} \\ \vdots & & \vdots \\ g_{N_2-1, N_2-1} & \cdots & g_{N_2-1, N_2-N_2} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_2} \quad (6.54)$$

mit

$$g_{p,q} = \begin{cases} \mathbf{c}^T \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{k} & \text{für } q > 0 \\ \mathbf{c}^T \mathbf{k} & \text{für } q = 0 \\ 0 & \text{für } q < 0 \end{cases} \quad (6.55)$$

Zunächst wurde von einem bekannten Soll-Zustandsverlauf $\mathbf{x}^{soll}$ ausgegangen, um das Prinzip der Prädiktion auch bei mehrdimensionaler Nichtlinearität klar zu machen. In der Realität ist dies nicht immer der Fall, so dass Maßnahmen zur Bestimmung eines geeigneten Soll-Zustandsverlaufs ergriffen werden müssen.

6.2.1 Sollverlauf des Zustandsvektors

Die Abhängigkeit der nichtlinearen Funktion vom Zustandsvektor (im Vergleich zu einer Abhängigkeit vom Ausgangssignal) erschwert im Allgemeinen die Linearisierung erheblich, da nicht mehr die gegebene Sollwerttrajektorie $\mathbf{r}$ als Referenz für die Linearisierung verwendet werden kann. Es muss in diesem Fall für eine Linearisierung erst ein geeigneter Sollverlauf des Zustandsvektors generiert werden. Davon ausgenommen sind Systeme, für

die eine Zustandstrajektorie $\mathbf{x}^{soll}$ als Sollwert vorgegeben ist, entlang derer eine Linearisierung möglich ist. In Chargenprozessen der chemischen Industrie beispielsweise können physikalische Systemzustände wie Füllstand, Druck oder Temperatur durch die Rezeptur als Sollwerte vorgeschrieben sein. In solchen Fällen existiert kein echter physikalischer und verfahrenstechnisch relevanter Prozessausgang, so dass dessen reine systemtheoretische Definition überflüssig erscheint. Dann bietet sich eine Abwandlung der Gütefunktion an, die dadurch als Regelfehler die Differenz zwischen Sollverlauf der Prozesszustände von dem prädierten Zustandsvektor mit der positiv definiten Diagonalmatrix $\mathbf{Q}$ gewichtet.

$$J(\Delta \mathbf{u}) = \sum_{j=N_1}^{N_2} \|\mathbf{x}^{soll}(k+j) - \tilde{\mathbf{x}}(k+j)\|_{\mathbf{Q}}^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (6.56)$$

Wenn dagegen kein Referenzverlauf $\mathbf{x}^{soll}$ der Zustände gegeben ist und der Prozessausgang einer gegebenen Sollwerttrajektorie folgen soll, dann entsteht dabei das Problem, dass aus dem Sollverlauf für den Ausgang nicht eindeutig auf den Verlauf des Zustandsvektors zurückgeschlossen werden kann. Aufgrund der Skalarproduktbildung $\mathbf{c}^T \mathbf{x}$ in Gleichung (6.43) existieren nämlich unendlich viele Zustände, welche die Bedingung $r(k) \stackrel{!}{=} y(k) = \mathbf{c}^T \mathbf{x}(k) + d\Delta u(k)$ erfüllen. Dies liegt darin begründet, dass eine Zustandsdarstellung niemals eindeutig bezüglich des Ein- Ausgangsverhaltens ist [15]. Auf den ersten Blick ist jeder Zustand als Sollwert geeignet, der den entsprechenden Wert am Ausgang des Prozesses erzeugt. Allerdings muss zusätzlich bedacht werden, dass der Prozess dem vorausberechneten Verlauf der Systemzustände auch folgen kann. Somit wird die Menge der geeigneten Sollwerttrajektorien auf die Menge der realisierbaren Trajektorien eingeschränkt. Eine sinnvolle Linearisierung erfordert die Kenntnis einer Referenztrajektorie, in deren Nähe die tatsächlichen Werte liegen werden, damit der Approximationsfehler möglichst gering bleibt.

Das Grundprinzip für die Berechnung der Zustandstrajektorie besteht darin, a-priori möglichst realistische Annahmen über den zukünftigen Stellgrößenverlauf zu treffen, das heißt, eine Schätzung des weiteren Verlaufes vor der eigentlichen Berechnung der Stellgröße vorzunehmen. Aufgrund dieser Annahme kann mit dem nichtlinearen Prozessmodell nach Gleichung (6.43) ein Verlauf des Zustandsvektors errechnet werden, der als Solltrajektorie $\mathbf{x}^{soll}$ dient. In Bild 6.3 ist die Funktionsweise des Trajektoriengenerators grafisch dargestellt. Ei-

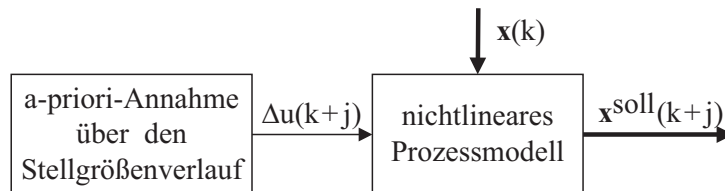


Bild 6.3: Prinzip eines Solltrajektorienplaners

ne geeignete Annahme über den zukünftigen Stellgrößenverlauf und der aktuelle Zustandsvektor des Prozesses bilden die benötigten Eingangsinformationen für das Prozessmodell, mit dem anschließend der Zustandsverlauf bestimmt werden kann. Dieser Zustandsverlauf wird als Referenztrajektorie $\mathbf{x}^{soll}$ für die Linearisierung des Prädiktionsmodells verwendet. Die Linearisierung entlang dieser Trajektorie führt dann auf das zeitvariante lineare

Prädiktionsmodell aus Gleichung (6.46). In den folgenden Abschnitten werden verschiedene a-priori-Annahmen über den Stellgrößenverlauf untersucht.

6.2.2 Aktueller Zustand als Referenzpunkt

Die Linearisierung erfolgt nicht mehr entlang einer Trajektorie, sondern nur noch um den aktuellen Arbeitspunkt. Die Solltrajektorie ist somit kein echter Verlauf mehr, sondern degeneriert zu einem einzigen Punkt, nämlich dem aktuellen Zustand $\mathbf{x}(k)$. Dieser Zustand wird als Sollzustand innerhalb des gesamten Vorhersagezeitraumes verwendet. Anstelle der zukünftigen, und damit von $\Delta \mathbf{u}$ abhängigen, Prozesszustände wird lediglich der aktuelle Zustandsvektor als Argument in die nichtlineare Funktion eingesetzt, welcher nur von vergangenen – und damit bekannten – Stellgrößen abhängt. Somit folgt für den angenommenen Sollverlauf des Zustandsvektors:

$$\mathbf{x}^{soll}(k+j) = \mathbf{x}(k) \quad (6.57)$$

Dies bedeutet für die Linearisierung erster Ordnung von $\mathcal{NL}$ innerhalb des Prädiktionshorizonts:

$$\mathcal{NL}(\mathbf{x}(k+j)) \approx \mathcal{NL}(\mathbf{x}(k)) + \text{grad}^T(\mathcal{NL}(\mathbf{x}(k))) \cdot (\mathbf{x}(k+j) - \mathbf{x}(k)) \quad (6.58)$$

Gleichung (6.58) wird nun für den gesamten Prädiktionshorizont angewendet. Dies ergibt nach Gleichung (6.46) ein lineares Prädiktionsmodell, dessen Systemmatrizen während der Prädiktion konstant sind, die aber bei jedem Abtastschritt neu bestimmt werden müssen. In der Prädiktionsgleichung (6.47) werden die Matrix $\mathbf{A}(k+j)$ und das Signal $v(k+j)$ zeitinvariant.

$$\mathbf{A}(k+j) = \mathbf{A}(k) = \mathbf{A} + \mathbf{k} \text{grad}^T(\mathcal{NL}(\mathbf{x}(k))) \quad (6.59)$$

$$v(k+j) = v(k) = \mathcal{NL}(\mathbf{x}(k)) - \text{grad}^T(\mathcal{NL}(\mathbf{x}(k))) \cdot \mathbf{x}(k) \quad (6.60)$$

Dies bewirkt, dass die Nichtlinearität und ihr Gradient nur einmal pro Abtastschritt ausgewertet werden muss. Die Produkte über die Matrizen $\mathbf{A}(k+j)$ in Gleichung (6.47) werden dann zu einer einfachen Potenz der konstanten Matrix $\mathbf{A}(k)$. Außerdem besteht der Vektor $\mathbf{v}$ aus lauter gleichen Einträgen.

Diese Näherung ist sicher nur dann brauchbar, wenn sich der Zustandsvektor innerhalb des Vorhersagezeitraumes bis $k + N_2$ wenig verändert. Dies widerspricht der Forderung, dass die Länge des Vorhersagezeitraumes das Erfassen relevanter Prozessantworten erlauben muss. Somit ist eine deutliche Abweichung der prädizierten Prozesszustände vom aktuellen Zustand $\mathbf{x}(k)$ zu erwarten, um den linearisiert wird, so dass diese Variante nur als grobe Näherung aufgefasst werden kann. Jedoch zeichnet sich das Verfahren durch geringen Rechenaufwand und universelle Anwendbarkeit aus. Es werden keine weiteren Forderungen an das System gestellt.

6.2.3 Solltrajektorie bei Prozessen mit Relativgrad eins

Hier wird das Problem der Rückrechnung von einem Sollverlauf der Ausgangsgröße auf einen Sollverlauf des Zustandsvektors für den Spezialfall von Prozessen mit dem Rela-

tivgrad eins betrachtet. Dies schließt Prozesse mit einem Durchgriff $d \neq 0$ aus.¹⁾ Das bedeutet, eine veränderte Eingangsgröße wirkt sich bereits nach einem Abtastschritt im Ausgangssignal aus. Um einen gewünschten Wert $r(k)$ am Ausgang zu erzeugen, gibt es zunächst unendlich viele Zustandskombinationen, die alle auf einer n -dimensionalen Hyperebene liegen, die durch die Gleichung

$$r(k) \stackrel{!}{=} \mathbf{c}^T \mathbf{x}(k) \quad (6.61)$$

festgelegt ist. Allerdings existiert nur *eine Kombination*, die sowohl den gewünschten Ausgangswert liefert und die der Prozess durch eine Stellgrößenänderung innerhalb *eines* Zeitschritts erreichen kann. Für die Berechnung dieser Zustandskombination wird die Forderung aufgestellt, dass der Prozessausgang zu jedem Zeitpunkt mit der Referenztrajektorie übereinstimmen muss. Um dies zu erreichen, wird die zugehörige Stellgröße Δu^{exakt} bestimmt. Mit dieser Stellgröße lässt sich die nächste Zustandskombination bestimmen, die $y(k+1) = r(k+1)$ erfüllt. Dieses Vorgehen wird für den gesamten Prädiktionshorizont fortgesetzt. Man erhält so eine Stellgrößenfolge $\Delta \mathbf{u}^{exakt}$, die, wenn man sie auf den Prozess schalten würde, einen Zustandsverlauf generiert, für den $y(k+j) = r(k+j)$ gilt. Die Folge $\Delta \mathbf{u}^{exakt}$ wird im Trajektorienplaner in Bild 6.4 einem Prozessmodell zugeführt, welches den zugehörigen Verlauf $\mathbf{x}^{soll}$ als Referenztrajektorie der Zustandsgrößen für die Linearisierung liefert. Ohne Beschränkung der Allgemeinheit wird hier ein System mit $N = 2$ Zuständen

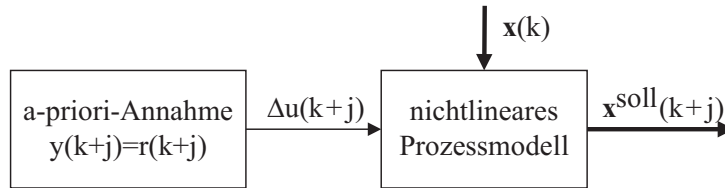


Bild 6.4: Solltrajektorienplaner für sprungfähige Prozesse

betrachtet, weil das Vorgehen dann geometrisch in einer zweidimensionalen Ebene veranschaulicht werden kann.

Das Skalarprodukt $r(k+1) \stackrel{!}{=} \mathbf{c}^T \mathbf{x}(k+1)$ beschreibt eine Geradengleichung $\mathbf{g}(k+1)$ in der Zustandsebene. Die Steigung dieser Geraden beträgt $-c_1/c_2$. Der Achsabschnitt hängt von r ab und beträgt $r(k+1)/c_2$. Also kann der Sollwert $r(k+1)$ am Ausgang durch alle Zustände $\mathbf{x}(k+1)$ erzeugt werden, die auf dieser Geraden liegen.

In Bild 6.5 wird der Übergang vom Zustand $\mathbf{x}(k)$ in den Zustand $\mathbf{x}(k+1)$ betrachtet, der in zwei Schritten vollzogen wird. Ausgehend von $\mathbf{x}(k)$ läuft zuerst die Zustandsänderung ab, die durch die freie Bewegung erfolgt. Diese ist die Antwort des nichtlinearen Prozesses auf den Anfangszustand $\mathbf{x}(k)$ ohne Änderung der Stellgröße ($\Delta u(k) = 0$). Dieser neue Zustand soll $\mathbf{x}^{drift}(k)$ genannt werden. Von dort aus kann nun durch ein $\Delta u(k) \neq 0$ der neue Zustand $\mathbf{x}(k+1)$ erreicht werden, der auf der durch $r(k+1)$ bestimmten Geraden liegen soll. Allerdings kann der Zustand $\mathbf{x}^{drift}(k)$ nur in *einer* Richtung verlassen werden, die durch den Einkoppelvektor $\mathbf{b}$ festgelegt ist. Somit existiert ein eindeutiges $\Delta u^{exakt}(k)$,

¹⁾ Eine Erweiterung auf Prozesse mit $d \neq 0$ ist möglich, wird zum besseren Verständnis hier aber nicht ausgeführt.

welches den neuen Zustand $\mathbf{x}(k+1)$ genau auf der gewünschten Geraden zu liegen kommen lässt. Der gesuchte Sollwert $\mathbf{x}^{soll}(k+1)$, der später zur Linearisierung dient, wird als der Schnittpunkt der Richtungsgeraden $\mathbf{b}$ mit der Geraden $\mathbf{g}(k+1)$ definiert. Daraus wird sofort deutlich, dass ein solcher Schnittpunkt nur dann existiert, wenn die Gerade $\mathbf{g}(k)$ und der Richtungsvektor $\mathbf{b}$ nicht parallel sind. Dies entspricht genau der eingangs erwähnten Forderung nach einem Relativgrad von eins.

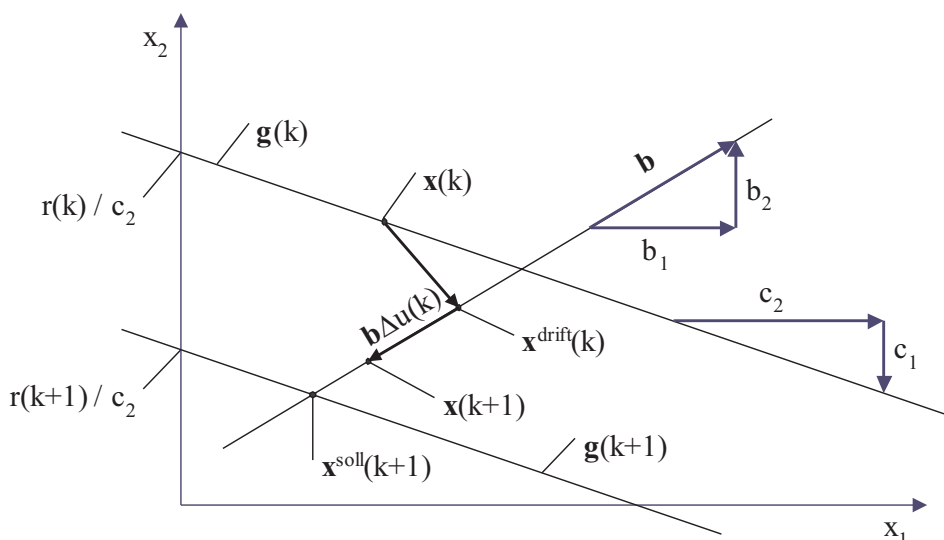


Bild 6.5: Berechnung der Solltrajektorie für sprungfähige Prozesse

Ebenso lässt sich die geschilderte Vorgehensweise auch mathematisch formulieren. Das Ausmultiplizieren des Skalarproduktes der Auskoppelgleichung $r(k+1) = y(k+1) = \mathbf{c}^T \mathbf{x}(k+1)$ ergibt die Ebenengleichung:

$$E : c_1 x_1 + \dots + c_n x_n \stackrel{!}{=} r(k+1) \quad (6.62)$$

Die Richtungsgerade in Parameterform ist gegeben durch den Aufpunkt $\mathbf{x}^{drift}(k)$ und den Richtungsvektor $\mathbf{b}$.

$$\mathbf{x} = \mathbf{x}^{drift} + \mathbf{b} \Delta u \quad (6.63)$$

Der Vektor $\mathbf{x}^{drift}$ berechnet sich aus der freien Antwort des Prozessmodells.

$$\mathbf{x}^{drift}(k) = \mathbf{A} \mathbf{x}(k) + \mathbf{k} \mathcal{L}(\mathbf{x}(k)) \quad (6.64)$$

Durch Einsetzen der Geradengleichung (6.63) in die Ebenengleichung (6.62) kann die gesuchte Stellgröße

$$\Delta u^{exakt}(k) = \frac{r(k+1) - \mathbf{c}^T \mathbf{x}^{drift}(k)}{\mathbf{c}^T \mathbf{b}} \quad (6.65)$$

berechnet werden, welche notwendig ist, um als Ausgangsgröße exakt den Sollwert $r(k+1)$ zu erreichen, d.h. den Regelfehler zu null werden zu lassen. Mit diesem Wert könnte also die Regelstrecke bereits beaufschlagt werden. Dann ist aber die Beachtung des Gütefunktional

(5.4) nicht mehr gegeben, da der Term $\sum \Delta u^2(k+j)$, der die Stellgrößenänderungen gewichtet, nicht berücksichtigt wurde. Daher wird Δu^{exakt} nicht direkt als Stellgröße verwendet, sondern dient lediglich zur Berechnung einer Zustandssolltrajektorie. Der gesuchte Sollwert $\mathbf{x}^{soll}$ ist der Schnittpunkt der Geraden mit der Hyperebene und ergibt sich zu:

$$\mathbf{x}^{soll}(k+1) = \mathbf{x}^{drift}(k) + \mathbf{b}\Delta u^{exakt}(k) \quad (6.66)$$

Für die Prädiktion muss $\mathbf{x}^{soll}(k+j)$ für alle $N_1 \leq j \leq N_2$ bestimmt werden. Dies geschieht durch wiederholtes Auswerten der Gleichungen (6.64), (6.65) und (6.66).

Diese Variante der Erzeugung der Solltrajektorie ist auf Prozesse eingeschränkt, für deren Beschreibung gilt:

$$\mathbf{c}^T \mathbf{b} \neq 0 \quad (6.67)$$

Andernfalls ist der Nenner von Gleichung (6.65) gleich null und Δu^{exakt} strebt gegen Unendlich. Die zugrunde liegende systemtechnische Erklärung ist, dass durch die Stellgröße nur Zustände direkt beeinflusst werden können, die nicht ausgekoppelt werden. D.h. im darauffolgenden Zeitschritt kann die Ausgangsgröße durch $\Delta u(k)$ nicht beeinflusst werden, sie kann sich nur aufgrund der freien Antwort verändern. Solche Prozesse sind jedoch keineswegs selten. In der Antriebstechnik beispielsweise tritt bei Mehrmassensystemen genau dieser Fall auf. Die Bedingung (6.67) stellt zwar eine hohe Anforderung an den zu regelnden Prozess dar, wenn sie aber erfüllt ist, kann eine realistische Zustandssolltrajektorie zur Linearisierung um diese Trajektorie erzeugt werden. Die Prädiktion kann dann gemäß Kap. 6.2 durchgeführt werden.

6.2.4 Verwendung des Stellgrößenvektors des vorhergehenden Zeitschrittes

Da die Forderung (6.67) nur für eine geringe Anzahl an technisch relevanten Systemen erfüllt ist, wird nun ein allgemeingültigeres Verfahren zur Erzeugung einer realistischen Referenztrajektorie für den Zustandsverlauf vorgestellt.

Üblicherweise wird zu jedem Abtastzeitpunkt aus der Optimierung einer Kostenfunktion eine Stellfolge ermittelt, von der nur das erste Element $\Delta u(k)$ als aktuelle Stellgrößenänderung genutzt wird. Alle weiteren Werte $\Delta u(k+1), \dots, \Delta u(k+N_u)$ werden verworfen, da diese im Sinne einer Regelung beim nächsten Zeitschritt erneut berechnet werden. Die Stellgröße $\Delta u(k+1|k)$, welche zum aktuellen Zeitpunkt k berechnet wird, entspricht wegen des zurückweichenden Horizontes also zeitlich der Stellgröße $\Delta u(k+1|k+1)$, die im darauf folgenden Zeitschritt $k+1$ errechnet wird. Allerdings ist hierbei zu beachten, dass diese beiden Stellgrößen, die zu unterschiedlichen Zeitpunkten berechnet werden, nicht notwendigerweise die selben Zahlenwerte aufweisen müssen, obwohl beide die Stellgrößenänderung für den selben Zeitpunkt beschreiben. Die Abweichung der beiden Zahlenwerte liegt in der Tatsache begründet, dass die Berechnung von $\Delta u(k+1|k)$ zum Zeitpunkt k auf der Messung des Zustandsvektors $\mathbf{x}(k)$ basiert, wohingegen die Berechnung von $\Delta u(k+1|k+1)$ auf dem Zustandsvektor zum darauf folgenden Zeitpunkt beruht. Selbst bei idealer Übereinstimmung des Prädiktionsmodelles mit dem realen Prozess und bei Abwesenheit von Störungen tritt ein Unterschied zwischen beiden Größen auf. Diese Tatsache wird vom Sollwert $r(k+N_2+1)$ verursacht, der wegen des zurückweichenden

Horizonts zum Zeitpunkt $k + 1$ in die Gütefunktion eingeht, zum Zeitpunkt k aber noch nicht berücksichtigt wurde. Aus demselben Grund fehlt der Sollwert $r(k + N_1)$ im nächsten Zeitschritt in der Kostenfunktion, so dass sich das Optimierungsproblem zwischen zwei aufeinander folgenden Abtastzeitpunkten verändert. In Tabelle 6.1 ist die zeitliche Korre-

Zeitpunkt	Δu
k	$\Delta u(k k) \quad \Delta u(k+1 k), \quad \dots, \quad \Delta u(k+N_u k)$
	$\downarrow \qquad \qquad \qquad \downarrow$
$k+1$	$\Delta u(k+1 k+1), \quad \dots, \quad \Delta u(k+N_u k+1), \quad \Delta u(k+N_u+1 k+1)$

Tabelle 6.1: Korrespondenz der Stellfolgen $\Delta u(k+j|k)$ und $\Delta u(k+j|k+1)$

spondenz der Stellgrößenfolgen $\Delta u(k+j|k)$ und $\Delta u(k+j|k+1)$ dargestellt. Zwei untereinander stehende Größen beschreiben die Stelländerung für einen identischen Zeitpunkt. Die Werte der oberen Zeile sind zum Zeitpunkt k berechnet worden, die Werte der unteren Zeile zum Zeitpunkt $k+1$. Es liegt nun nahe, die folgende Näherung durchzuführen:

$$\Delta u(k+j|k+1) \approx \Delta u(k+j|k) \quad (6.68)$$

Zum Zeitpunkt $k+1$ sind damit aus der Berechnung von $\Delta u(k+j|k)$ zum Zeitpunkt k alle Stelländerungen bis $k+N_u$ vorhanden. Die letzte Stelländerung $\Delta u(k+N_u+1|k+1)$ ist aus der vorigen Berechnung noch nicht bekannt und wird mit null aufgefüllt. Daraus ist ersichtlich, dass zur näherungsweisen Vorhersage des Zustandsverlaufes zum Zeitpunkt $k+1$ die Stellfolge $\Delta u(k+j|k)$ des vorangegangenen Zeitschrittes verwendet werden kann. Dies geschieht durch die folgenden Gleichungen:

$$\begin{aligned}
 \mathbf{x}^{soll}(k+1) &= \mathbf{x}(k+1) \\
 \mathbf{x}^{soll}(k+2) &= \mathbf{A}\mathbf{x}(k+1) + \mathbf{b}\Delta u(k+1|k) + \mathbf{k}\mathcal{NL}(\mathbf{x}(k+1)) \\
 \mathbf{x}^{soll}(k+3) &= \mathbf{A}\mathbf{x}^{soll}(k+2) + \mathbf{b}\Delta u(k+2|k) + \mathbf{k}\mathcal{NL}(\mathbf{x}^{soll}(k+2)) \\
 &\vdots \\
 \mathbf{x}^{soll}(k+N_2+1) &= \mathbf{A}\mathbf{x}^{soll}(k+N_2) + \mathbf{b}\Delta u(k+N_2|k) + \\
 &\quad + \mathbf{k}\mathcal{NL}(\mathbf{x}^{soll}(k+N_2))
 \end{aligned} \quad (6.69)$$

In Gleichung (6.69) ist zu beachten, dass sich für die Linearisierung die Zustandstrajektorie vom Zeitpunkt $k+1$ bis zum Zeitpunkt $k+N_2+1$ erstrecken muss. Die Verwendung der Stellfolge des letzten Zeitschrittes führt jedoch nur auf eine Trajektorie, die bis zum Zeitpunkt $k+N_u$ reicht. Für den allgemeinen Fall $N_u < N_2$ sind zu wenig Stelländerungen bekannt, um den gesamten Zustandsverlauf präzisieren zu können. Deshalb werden die unbekannten Stelländerungen ab dem Zeitpunkt $k+N_u+1$ zu null angenommen, was der Einführung des Stellhorizontes entspricht, der ab $k+N_u$ Stellgrößenänderungen unterbindet. Wie aus

den obigen Ausführungen hervorgeht, stellt die Prädiktion der Zustandstrajektorie $\mathbf{x}^{soll}$ basierend auf den Werten $\Delta u(k+j|k)$ lediglich eine Näherung dar, deren Genauigkeit jedoch sehr gut ist. Diese Vorgehensweise ermöglicht die Prädiktion einer Trajektorie, die *unabhängig* ist von den zu berechnenden Stellgrößenänderungen $\Delta u(k+j|k+1)$. Es werden die nicht zur Regelung verwendeten Elemente des Stellgrößenänderungsvektors $\Delta \mathbf{u}$ für die Schätzung einer Solltrajektorie $\mathbf{x}^{soll}$ für den darauffolgenden Zeitpunkt genutzt.

Zusammenfassend ist festzuhalten, dass diese Methode die Stellfolge des vorhergehenden Zeitschrittes verwendet, welche auf einer optimal-Steuerung basiert, um eine Zustandssolltrajektorie zu berechnen. Mit dieser Trajektorie kann eine Linearisierung der Prozessbeschreibung gemäß Kap. 6.2 vorgenommen werden, was zum linearisierten, zeitvarianten Prädiktionsmodell führt. Anschließend wird die Minimierung der darauf aufbauenden Kostenfunktion vorgenommen, die eine neue optimale Stellfolge ergibt, von der im Sinne einer Regelung nur das erste Element $\Delta u(k)$ als Stellgrößenänderung aufgeschaltet wird. Die restlichen Stellgrößenänderungen werden nun nicht verworfen, sondern dienen wiederum zur Vorhersage der nächsten Zustandssolltrajektorie.

Es sind auch einfachere a-priori Annahmen über den zukünftigen Stellgrößenverlauf möglich. In [55] werden die Annahmen $\Delta u = const$ und $u = const$ untersucht. In Simulationen hat sich jedoch gezeigt, dass die Güte des damit erzeugten Sollzustandsverlaufs $\mathbf{x}^{soll}$ erheblich vom tatsächlichen Verlauf der Referenztrajektorie abhängt. Daher wurden diese beiden Methoden nicht weiter ausgeführt.

6.3 Regelgesetz ohne Begrenzungen

Die Prädiktion zukünftiger Ausgangssignale kann auf Basis der linearisierten Zustandsraummodelle trotz unterschiedlicher Abhängigkeiten der Nichtlinearität formal immer durch die selbe Gleichung ausgedrückt werden. Die prädizierten Ausgangssignale ergeben sich zu:

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (6.70)$$

Die Bildungsvorschriften der Matrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ unterscheiden sich allerdings. Bei einer Abhängigkeit der Nichtlinearität von inneren Zustandsgrößen, muss vor dem Aufbau der Matrizen noch ein Solltrajektorienplaner vorgeschaltet werden. Die unterschiedlichen Bildungsvorschriften der Matrizen sind in den Kap. 6.1 und 6.2 erläutert.

Nachdem das Ausgangssignal innerhalb des gesamten Vorhersagezeitraumes prädiziert werden muss, der sich vom unteren bis zum oberen Prädiktionshorizont erstreckt, müssen immer genau $N_2 - N_1 + 1$ Prädiktionsgleichungen berücksichtigt werden. Daher weisen alle drei Prädiktionsmatrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ $N_2 - N_1 + 1$ Zeilen auf. Die Spaltenzahlen der drei Matrizen hingegen sind unterschiedlich und hängen von der Ordnung des Prozesses, dem unteren und oberen Prädiktionshorizont und dem Stellhorizont ab. Zusätzlich muss bei der Wahl $N_u = N_2$ und $d = 0$ eine Spalte der Matrix $\mathbf{H}$ gestrichen werden (siehe Gleichung (6.19)). Die Auswirkung der Nichtlinearität muss ab dem aktuellen Zeitpunkt bis zum Ende des Vorhersagezeitraumes berücksichtigt werden. Eine Verringerung der Spaltenzahl der Matrix $\mathbf{G}$ analog zur Einführung eines Stellhorizonts ist daher nicht zulässig.

Das prädiktive Regelgesetz ergibt sich aus der wiederholten optimalen Lösung eines Steuerungsproblems, die zu jedem Abtastschritt neu berechnet wird. Die optimale Lösung leitet sich aus der Forderung ab, dass die Stellgröße die Gütefunktion (6.71) minimiert.

$$\begin{aligned}
 J(\Delta \mathbf{u}) &= \sum_{j=N_1}^{N_2} [r(k+j) - \tilde{y}(k+j)]^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) = \\
 &= (\mathbf{r} - \tilde{\mathbf{y}})^T \cdot (\mathbf{r} - \tilde{\mathbf{y}}) + \lambda \cdot \Delta \mathbf{u}^T \Delta \mathbf{u} = \\
 &= \Delta \mathbf{u}^T (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E}) \Delta \mathbf{u} - 2 \mathbf{d}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{d}^T \mathbf{d} \\
 \text{mit } \mathbf{d} &= \mathbf{r} - \mathbf{F} \mathbf{x}(k) - \mathbf{G} \mathbf{v}
 \end{aligned} \tag{6.71}$$

Die Vektoren $\Delta \mathbf{u}$, $\mathbf{v}$ und $\tilde{\mathbf{y}}$ sind gemäß den Gleichungen (5.3) und (6.24) definiert, $\mathbf{E}$ kennzeichnet die Einheitsmatrix. Die freien Entscheidungsvariablen stellen die Elemente von $\Delta \mathbf{u}$ dar. Mit $\Delta \mathbf{u}^*$ ist die gesuchte optimale Stellgrößenfolge gekennzeichnet, für die das Gütefunktional (6.71) den minimalen Wert annimmt. Wenn keinerlei Beschränkungen (der Zustandsgrößen, der Stellgrößen und der Stellgrößenänderungen) zu berücksichtigen sind, wird die Optimierungsaufgabe nicht durch Nebenbedingungen eingeschränkt und ist damit analytisch lösbar. Es ist möglich, das Regelgesetz in einer geschlossenen Form darzustellen. Die erste notwendige Optimalitätsbedingung erfordert das Nullsetzen des Gradienten der Gütefunktion:

$$\frac{\partial J(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}} = 2(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E}) \Delta \mathbf{u} - 2 \mathbf{H}^T \mathbf{d} \stackrel{!}{=} 0 \tag{6.72}$$

$$\Rightarrow \Delta \mathbf{u}^* = (\mathbf{H}^T \mathbf{H} + \lambda \cdot \mathbf{E})^{-1} \mathbf{H}^T \mathbf{d} \tag{6.73}$$

Als zweite Bedingung muss für ein Minimum die zweite Ableitung

$$\frac{\partial^2 J(\Delta \mathbf{u})}{\partial^2 \Delta \mathbf{u}} = 2(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E}) \tag{6.74}$$

positiv definit sein, was mit einer positiv definiten Matrix $\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E}$ erfüllt ist. Unter der Annahme $\lambda > 0$ ist dies gesichert (siehe Beweis Kap. 6.5, Gleichung (6.127)).

Nachdem kein Steuergesetz angewendet wird, sondern eine closed-loop-Regelung implementiert werden soll, ist nur das erste, aktuelle Element $\Delta u^*(k)$ des Lösungsvektors $\Delta \mathbf{u}^*$ relevant und wird dem Prozess als aktuelle Stellgröße aufgeschaltet. Alle zukünftigen Stellgrößenänderungen werden verworfen und beim nächsten Abtastschritt erneut berechnet. Dies geschieht ausgehend vom später tatsächlich erreichten Zustand. Die gesuchte Stellgröße kann aus dem Vektor $\Delta \mathbf{u}^*$ durch Multiplikation mit einem *Ausblendvektor* extrahiert werden.

$$\begin{aligned}
 \Delta u^*(k) &= [1, 0, \dots, 0] \cdot \Delta \mathbf{u}^* \\
 &= \underbrace{[1, 0, \dots, 0] \cdot (\mathbf{H}^T \mathbf{H} + \lambda \cdot \mathbf{E})^{-1} \mathbf{H}^T}_{\mathbf{T}} \cdot \mathbf{d} \\
 &= \mathbf{T} \cdot (\mathbf{r} - \mathbf{F} \cdot \mathbf{x}(k) - \mathbf{G} \cdot \mathbf{v})
 \end{aligned} \tag{6.75}$$

Um eine abkürzende Schreibweise des Regelgesetzes zu erhalten, können die Matrixmultiplikationen in Gleichung (6.75) zusammengefasst werden.

$$\mathbf{K} = \mathbf{T} \cdot \mathbf{F} \quad \mathbf{L} = \mathbf{T} \cdot \mathbf{G} \quad (6.76)$$

Damit folgt schließlich das prädiktive Regelgesetz in Gleichung (6.77), welches wegen des Terms $\mathbf{K} \cdot \mathbf{x}$ zur Klasse der Zustandsregler gezählt werden kann. Allerdings ist zu beachten, dass die Matrizen $\mathbf{K}$, $\mathbf{L}$ und $\mathbf{T}$ zeitvariant sind und zu jedem Abtastschritt neu berechnet werden. Man kann demnach von einem zeitvarianten Zustandsregler sprechen, der jedoch zukünftige Sollwerte im Vektor $\mathbf{r}$ berücksichtigt.

$$\Delta u^*(k) = \mathbf{T} \cdot \mathbf{r} - \mathbf{K} \cdot \mathbf{x}(k) - \mathbf{L} \cdot \mathbf{v} \quad (6.77)$$

Der Anteil $\mathbf{K} \cdot \mathbf{x}(k)$ stellt eine Zustandsrückführung dar, $\mathbf{T} \cdot \mathbf{r}$ berücksichtigt zukünftige Sollwerte und $\mathbf{L} \cdot \mathbf{v}$ stellt den Anteil der Nichtlinearität dar. Gleichung (6.77) wird online bei jedem Abtastschritt ausgewertet. Bild 6.6 zeigt den Signalflussplan des prädiktiv geregelten Prozesses auf Basis der linearisierten Zustandsraummodelle.

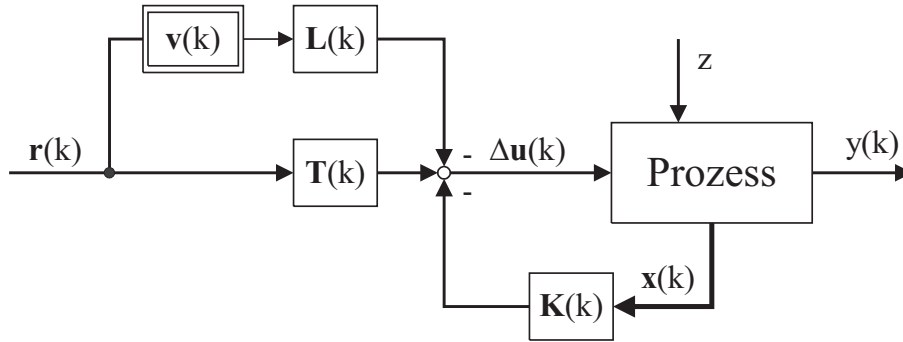


Bild 6.6: Prädiktiver Regelkreis ohne Berücksichtigung von Prozessbeschränkungen

Zur Berechnung des Regelgesetzes für den unbeschränkten Fall ist die Invertierung der Matrix $(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E})$ notwendig. Es soll im Folgenden gezeigt werden, dass die geforderte Invertierbarkeit stets gegeben ist.

Wenn eine beliebige, symmetrische Matrix $\mathbf{M}$ positiv definit ist, dann sind ihre sämtlichen Eigenwerte Λ_i positiv [16]:

$$\mathbf{M} \text{ ist positiv definit} \Leftrightarrow \text{alle Eigenwerte } \Lambda_i > 0 \quad (6.78)$$

Die Determinante einer Matrix kann aus dem Produkt der Eigenwerte berechnet werden und ist daher für positiv definite Matrizen größer null:

$$\det \mathbf{M} = \Lambda_1 \Lambda_2 \cdots \Lambda_K > 0 \quad (6.79)$$

Nicht invertierbare Matrizen heißen *singulär* und zeichnen sich durch eine Determinante gleich null aus. Alle *regulären* (invertierbaren) Matrizen dagegen sind durch eine Determinante ungleich null gekennzeichnet:

$$\det \mathbf{M} = 0 \Leftrightarrow \mathbf{M}^{-1} \text{ existiert nicht} \quad (6.80)$$

$$\det \mathbf{M} \neq 0 \Leftrightarrow \mathbf{M}^{-1} \text{ existiert} \quad (6.81)$$

Für die Matrix $(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E})$ existiert demnach immer eine Inverse, wenn sie positiv definit ist. In Kap. 6.5, Gleichung (6.127) ist diese Eigenschaft für alle $\lambda > 0$ gezeigt, so dass die Existenz des Regelgesetzes (6.77) gesichert ist.

6.3.1 Stationäre Genauigkeit bei Störgrößen

Bisher wurden unbekannte Störgrößen in keiner Weise beim Regelungsentwurf berücksichtigt. Da eine Hauptforderung an eine Regelung jedoch die stationäre Unterdrückung von konstanten Störungen ist, wird die Regelung aus Bild 6.6 diesbezüglich untersucht und gegebenenfalls erweitert. Es wird dafür ein stationärer Zustand des geregelten Systems untersucht. Dies bedeutet, dass sich das System im eingeschwungenen Zustand befindet und eine konstante Referenztrajektorie vorliegt. Ohne Beschränkung der Allgemeinheit sei der stationäre Zustand im Ursprung und die Nichtlinearität nehme dort den Wert null an. Damit gilt:

$$\mathbf{x}_0 = 0 \quad y_0 = 0 \quad \Delta u = 0 \quad r(k+j) = \text{const} = 0 \quad (6.82)$$

Wegen der Konstanz der Referenztrajektorie ist die Zustandsmatrix $\mathbf{A}$ konstant und das die Nichtlinearität beschreibende Signal v verschwindet. Da die Zustandsrückführung $\mathbf{K}$ nur vom Verlauf der Referenztrajektorie abhängt und diese konstant ist, muss auch die Rückführung $\mathbf{K}$ konstant sein.

$$\mathbf{A} = \text{const.} \quad \mathbf{b} = \text{const} \quad v(k+j) = \text{const} = 0 \quad \mathbf{K} = \text{const} \quad (6.83)$$

Die Terme $\mathbf{L} \cdot \mathbf{v}$ und $\mathbf{T} \cdot \mathbf{r}$ des Regelgesetzes entsprechen einer Vorsteuerung, die zukünftige Sollwerte und den Einfluss der Nichtlinearität berücksichtigt. Diese Anteile können unbekannte Störgrößen nicht unterdrücken, da keine Rückkopplung vom Prozess vorhanden ist (siehe Bild 6.6). Die Störgröße z greift über einen zusätzlichen Einkoppelvektor $\mathbf{k}_z$ in das System ein. Die Systemgleichungen lauten dann:

$$\mathbf{x}(k+1) = \mathbf{A} \mathbf{x}(k) + \mathbf{b} \Delta u + \mathbf{k} v(k) + \mathbf{k}_z z(k) \quad y(k) = \mathbf{c}^T \mathbf{x}(k) \quad (6.84)$$

$$\Delta u = \mathbf{T} \cdot \mathbf{r} - \mathbf{K} \cdot \mathbf{x}(k) - \mathbf{L} \cdot \mathbf{v} \quad (6.85)$$

Der neue stationäre Zustand, der sich aufgrund der Störgröße z einstellt, kann mit der Bedingung $\mathbf{x}(k+1) = \mathbf{x}(k) = \mathbf{x}_z$ berechnet werden.

$$\mathbf{x}_z = \mathbf{A} \mathbf{x}_z - \mathbf{b} \mathbf{K} \mathbf{x}_z + \mathbf{k}_z z \quad (6.86)$$

Aus Gleichung (6.86) wird der neue stationäre Zustand $\mathbf{x}_z$ und die zugehörige stationäre Ausgangsgröße y_z bestimmt.

$$\mathbf{x}_z = (\mathbf{E} - \mathbf{A} + \mathbf{b} \mathbf{K})^{-1} \mathbf{k}_z \cdot z \quad \Rightarrow \quad y_z = \mathbf{c}^T \mathbf{x}_z = \underbrace{\mathbf{c}^T (\mathbf{E} - \mathbf{A} + \mathbf{b} \mathbf{K})^{-1} \mathbf{k}_z}_{V_{stat}} \cdot z \quad (6.87)$$

Die stationäre Störverstärkung V_{stat} ist im Allgemeinen ungleich null, so dass sich eine stationäre Abweichung zwischen r und y_z einstellen wird. Das Regelgesetz in der Form (6.77) arbeitet bei konstanten Störgrößen nicht stationär genau. Eine Erweiterung mit dem Ziel stationärer Genauigkeit wird an Kap. 6.4 hergeleitet.

6.4 Prädiktive Regelung mit Integralanteil

Bei den bisherigen Betrachtungen wurde die Auswirkung von unbekannten Störgrößen z auf das Systemverhalten außer Acht gelassen. In Kap. 6.3.1 wurde gezeigt, dass ohne weitere Maßnahmen eine stationäre Regelabweichung bestehen bleibt. Als Störungen werden in diesem Zusammenhang nicht messbare und unbekannte Eingangssignale des Systems betrachtet (z.B. Lastmomente in elektrischen Antrieben). Dieses Kapitel befasst sich mit einer Erweiterung des prädiktiven Regelungsverfahrens zur Unterdrückung konstanter Störgrößen.

Die linearisierte Prozessbeschreibung ergibt bei Linearisierung erster Ordnung ein zeitvariantes System. Mit der unbekannten skalaren Störgröße $z(k)$ und dem zugehörigen Einkoppelvektor $\mathbf{k}_z$ ist die Prozessbeschreibung nach Gleichung (6.88) gegeben.

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A} \cdot \mathbf{x}(k) + \mathbf{b}_u \cdot \Delta u(k) + \mathbf{k} \cdot v(k) + \mathbf{k}_z \cdot z(k) \\ y(k) &= \mathbf{c}^T \cdot \mathbf{x}(k) + d \cdot \Delta u(k)\end{aligned}\quad (6.88)$$

Die Ausregelung konstanter und unbekannter Störgrößen kann nach [24] durch die Erweiterung des Systems um einen Integrator erreicht werden. Dieser integriert die aktuelle Regeldifferenz in der zusätzlich entstandenen Zustandsgröße auf. Der integrierte Regelfehler wird in einem erweiterten Gütefunktional bewertet. Nach [24] werden folgende zwei Bedingungen vorausgesetzt (vor allem für Mehrgrößensysteme relevant):

1. Die Anzahl der Ausgangsgrößen ist nicht höher als die Anzahl der Eingangsgrößen.
2. Der Rang der Matrix des quadratischen Anteils der Gütefunktion in Gleichung (6.120) besitzt vollen Rang. In Kapitel 6.5 wird auf die Bildung und die Eigenschaften dieser Matrix näher eingegangen.

Mit diesen Annahmen wird ein zeitdiskreter Integrator mit der Zustandsvariablen x_s angesetzt, der die Regeldifferenz $r(k) - y(k)$ aufintegriert und die Ordnung des Prozesses um eins erhöht.

$$x_s(k+1) = x_s(k) + r(k) - \mathbf{c}^T \mathbf{x}(k) - d \Delta u(k) \quad (6.89)$$

N sei die Ordnung des erweiterten Systems. Die aufsummierte Regeldifferenz wird vom Zeitschritt $k+1$ bis zum oberen Prädiktionshorizont N_2 gebildet. Der Vektor der Referenztrajektorie $\mathbf{r}_2$ enthält, im Gegensatz zur Trajektorie $\mathbf{r}$, Einträge von $k+1$ bis $k+N_2$. In Bild 6.7 ist das um einen LQI-Regler erweiterte System dargestellt. Der Name *LQI* rührt daher, dass ein quadratisches Gütefunktional mit Integralanteil zugrunde liegt. Die Bedeutung der in Bild 6.7 auftretenden Matrix $\mathbf{T}_2$ wird weiter unten deutlich. Die Zustandsrückführung $\mathbf{K}$ ist in die beiden Anteile $\mathbf{K}_P$ und K_I , entsprechend den realen Zuständen des Prozesses und dem hinzugefügten Zustand x_s aufgeteilt worden. Die resultierende Zustandsbeschreibung des linearisierten Systems ergibt sich zu:

$$\underbrace{\begin{bmatrix} \mathbf{x}(k+1) \\ x_s(k+1) \end{bmatrix}}_{\mathbf{x}(k+1)} = \underbrace{\begin{bmatrix} \mathbf{A}(k) & \mathbf{0} \\ -\mathbf{c}^T & 1 \end{bmatrix}}_{\mathbf{A}(k)} \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ x_s(k) \end{bmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{bmatrix} \mathbf{b}(k) \\ -d \end{bmatrix}}_{\mathbf{b}(k)} \Delta u(k)$$

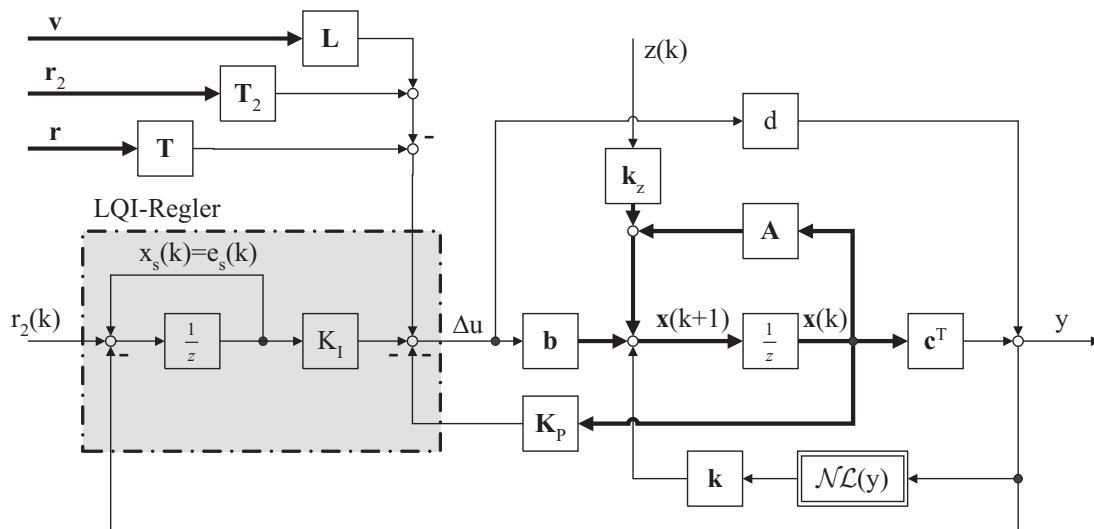


Bild 6.7: Prädiktive Regelung mit Integralanteil

$$+ \underbrace{\begin{bmatrix} \mathbf{k} \\ 0 \end{bmatrix}}_{\mathbf{k}} v(k) + \underbrace{\begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}}_{\mathbf{b}_w} r(k) + \underbrace{\begin{bmatrix} \mathbf{k}_z \\ 0 \end{bmatrix}}_{\mathbf{k}_z} z(k) \quad (6.90)$$

$$y(k) = \underbrace{\begin{bmatrix} \mathbf{c}^T & 0 \end{bmatrix}}_{\mathbf{c}^T} \begin{bmatrix} \mathbf{x}(k) \\ x_s(k) \end{bmatrix} + d \Delta u(k) \quad (6.91)$$

$$e_s(k) = \underbrace{\begin{bmatrix} 0^T & 1 \end{bmatrix}}_{\mathbf{c}_s^T} \begin{bmatrix} \mathbf{x}(k) \\ x_s(k) \end{bmatrix} = x_s(k) \quad (6.92)$$

Im Gütefunktional (6.93) wird neben der quadratischen Differenz zwischen Trajektorie und Ausgangswert $r(k+j) - \tilde{y}(k+j)$ und der quadratischen Stellgrößenänderung $\Delta u(k+j)$ noch zusätzlich der quadrierte aufsummierte Regelfehler $\tilde{e}_s(k+j)$ gewichtet.

$$J(\Delta u) = \sum_{j=N_1}^{N_2} (r(k+j) - \tilde{y}(k+j))^2 + \lambda \sum_{j=0}^{N_u} \Delta u^2(k+j) + \alpha \sum_{j=N_1}^{N_2} \tilde{e}_s^2(k+j) \quad (6.93)$$

Bei der Festlegung des dynamischen Verhaltens des Regelkreises wird neben dem Gewichtungsfaktor λ ein weiterer Gewichtungsfaktor α eingeführt, der für die Dynamik der Ausregelung des stationären Regelfehlers verantwortlich ist.

Neben dem Ausgangssignal $y(k)$ existiert nun eine zusätzliche Ausgangsgleichung $e_s(k)$, in der der aufsummierte Regelfehler ausgekoppelt wird. Die Prädiktion des Ausgangssignals und des aufsummierten Fehlers erfolgt in der bekannten Vektor-Matrix-Schreibweise:

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (6.94)$$

$$\tilde{\mathbf{e}}_s = \mathbf{F}_s \cdot \mathbf{x}(k) + \mathbf{H}_u \cdot \Delta \mathbf{u} + \mathbf{H}_w \cdot \mathbf{r}_2 + \mathbf{G}_s \cdot \mathbf{v} \quad (6.95)$$

Die verwendeten Vektoren sind wie folgt definiert:

$$\Delta \mathbf{u} = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_u)]^T \quad (6.96)$$

$$\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T \quad (6.97)$$

$$\tilde{\mathbf{y}} = [\tilde{y}(k+N_1), \tilde{y}(k+N_1+1), \dots, \tilde{y}(k+N_2)]^T \quad (6.98)$$

$$\mathbf{r} = [r(k+N_1), r(k+N_1+1), \dots, r(k+N_2)]^T \quad (6.99)$$

$$\mathbf{r}_2 = [r(k+1), r(k+2), \dots, r(k+N_2)]^T \quad (6.100)$$

$$\mathbf{e}_s = [e_s(k+N_1), e_s(k+N_1+1), \dots, e_s(k+N_2)]^T \quad (6.101)$$

Die Matrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ werden gemäß den Gleichungen (6.51) bis (6.55) gebildet. Es ist zu beachten, dass dabei die erweiterten Systemmatrizen benutzt werden müssen. Die Matrizen $\mathbf{F}_s$, $\mathbf{H}_u$ und $\mathbf{G}_s$ werden wie die Matrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ gebildet, der Unterschied besteht lediglich darin, dass statt des Auskoppelvektors $\mathbf{c}^T$, der Vektor $\mathbf{c}_s^T$ bei der Bildung verwendet wird. Die Matrix $\mathbf{H}_w$ entsteht wie die Matrix $\mathbf{H}$, es wird jedoch der Auskoppelvektor $\mathbf{c}_s^T$ und der zusätzliche Einkoppelvektor $\mathbf{b}_w$ verwendet. Es ist ferner zu beachten, dass sich der zeitliche Horizont der Matrix $\mathbf{H}_w$ zwischen 1 und N_2 bewegt.

$$\mathbf{H}_w = \begin{bmatrix} h_{N_1-1, N_1-1} & \dots & h_{N_1-1, N_1-N_2} \\ \vdots & & \vdots \\ h_{N_2-1, N_2-1} & \dots & h_{N_2-1, N_2-N_2} \end{bmatrix} \in \mathbb{R}^{N_2-N_1+1 \times N_2} \quad (6.102)$$

mit

$$h_{p,q} = \begin{cases} \mathbf{c}_s^T \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{b}_w & \text{für } q > 0 \\ \mathbf{c}_s^T \mathbf{b}_w & \text{für } q = 0 \end{cases} \quad (6.103)$$

Zum Nachweis der stationären Genauigkeit wird ein stationärer Arbeitspunkt untersucht. Die Referenztrajektorie sei im gesamten Prädiktionshorizont konstant ($r(k+j) = \text{const}$). Die Argumentation erfolgt mit den gemäß LQI-Regelung erweiterten Systemmatrizen. Ein stationärer Arbeitspunkt ist dadurch gekennzeichnet, dass keine zeitliche Veränderung der Zustandsgrößen stattfindet, dass also gilt:

$$\mathbf{x}(k+1) = \mathbf{x}(k) \quad (6.104)$$

Die letzte Zeile von Gleichung (6.90) lautet:

$$x_s(k+1) = x_s(k) + r(k) - y(k) \quad (6.105)$$

Im stationären Zustand müssen $x_s(k+1)$ und $x_s(k)$ gleich sein. Dies kann nur erfüllt werden, wenn $y(k)$ und $r(k)$ ebenfalls gleich sind. Damit ist gezeigt, dass jeder stationäre Punkt eines mit dem vorgeschlagenen LQI-Regler geregelten Systems auch stationär genau bezüglich des Ausgangssignals ist. Beim Beweis tauchte die Störgröße $z(k)$ nicht auf, d.h. die geforderte stationäre Genauigkeit gilt insbesondere auch für beliebige konstante Störgrößen. Dies gilt nur für $\alpha > 0$, da nur dann eine Gewichtung im Gütefunktional vorgenommen wird. Für $\alpha = 0$ verursacht der Zustand x_s keine Kosten im Gütefunktional

und der Einkoppelkoeffizient K_I in Bild 6.7 wird gleich null sein, da eine Abhängigkeit der Stellgröße von x_s nicht besteht. In diesem Fall ist der Zustand $\mathbf{x}_s$ im Ausgangssignal y nicht beobachtbar.

Die Untersuchung des Einstellparameters α und dessen Auswirkung auf die Dynamik des Regelkreises wird in Kap. 8 an einem Versuchsstand praktisch untersucht.

6.4.1 Regelgesetz ohne Begrenzungen

Die Berechnung der Stellgröße wird durch Minimierung eines erweiterten Gütefunktional durchgeführt. Es erfolgt eine zusätzliche Bewertung des aufsummierten Regelfehlers e_s . Das Gütefunktional in Gleichung (6.106) wird zur Stellgrößenoptimierung herangezogen.

$$J(\Delta u) = \sum_{j=N_1}^{N_2} (r(k+j) - \tilde{y}(k+j))^2 + \lambda \sum_{j=0}^{Nu} \Delta u^2(k+j) + \alpha \sum_{j=N_1}^{N_2} \tilde{e}_s^2(k+j) \quad (6.106)$$

Unter Verwendung der Vektoren der Gleichungen (6.96) bis (6.101) entsteht die Vektorschreibweise der Gütefunktion.

$$J = \underbrace{(\mathbf{r} - \tilde{\mathbf{y}})^T (\mathbf{r} - \tilde{\mathbf{y}})}_{J_1} + \underbrace{\lambda \Delta \mathbf{u}^T \Delta \mathbf{u}}_{J_2} + \underbrace{\alpha \tilde{\mathbf{e}}_s^T \tilde{\mathbf{e}}_s}_{J_3} \quad (6.107)$$

Werden in Gleichung (6.107) die Prädiktionsgleichungen (6.94) und (6.95) eingesetzt und ausmultipliziert, so kann das Minimum von $J = J_1 + J_2 + J_3$ durch Nullsetzen des Gradienten bestimmt werden.

$$\begin{aligned} J_1(\Delta \mathbf{u}) &= (\mathbf{r} - (\mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v}))^T \\ &\quad \cdot (\mathbf{r} - (\mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G} \cdot \mathbf{v})) \\ \frac{\partial J_1(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}} &= -2 \mathbf{H}^T \mathbf{r} + 2 \mathbf{H}^T \mathbf{F} \mathbf{x}(k) + 2 \mathbf{H}^T \mathbf{H} \Delta \mathbf{u} + 2 \mathbf{H}^T \mathbf{G} \mathbf{v} \end{aligned} \quad (6.108)$$

$$\begin{aligned} J_2(\Delta \mathbf{u}) &= \lambda \Delta \mathbf{u}^T \Delta \mathbf{u} \\ \frac{\partial J_2(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}} &= 2 \lambda \Delta \mathbf{u} \end{aligned} \quad (6.109)$$

$$\begin{aligned} J_3(\Delta \mathbf{u}) &= \alpha (\mathbf{F}_s \cdot \mathbf{x}(k) + \mathbf{H}_u \cdot \Delta \mathbf{u} + \mathbf{H}_w \cdot \mathbf{r}_2 + \mathbf{G}_s \cdot \mathbf{v})^T \\ &\quad \cdot (\mathbf{F}_s \cdot \mathbf{x}(k) + \mathbf{H}_u \cdot \Delta \mathbf{u} + \mathbf{H}_w \cdot \mathbf{r}_2 + \mathbf{G}_s \cdot \mathbf{v}) \\ \frac{\partial J_3(\Delta \mathbf{u})}{\partial \Delta \mathbf{u}} &= 2 \alpha \mathbf{H}_u^T \mathbf{F}_s \mathbf{x}(k) + 2 \alpha \mathbf{H}_u^T \mathbf{H}_u \Delta \mathbf{u} + 2 \alpha \mathbf{H}_w^T \mathbf{H}_w \mathbf{r}_2 \end{aligned} \quad (6.110)$$

Die differenzierten Terme (6.108) bis (6.110) werden wieder zusammengesetzt und zu null gesetzt, um das Minimum zu berechnen.

$$\begin{aligned} &2 (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u) \Delta \mathbf{u} + 2 \mathbf{H}^T (-\mathbf{r} + \mathbf{F} \mathbf{x}(k) + \mathbf{G} \mathbf{v}) \\ &+ 2 \mathbf{H}_u^T (\mathbf{F}_s \mathbf{x}(k) + \mathbf{H}_w \mathbf{r}_2 + \mathbf{G}_s \mathbf{v}) \stackrel{!}{=} 0 \end{aligned} \quad (6.111)$$

Das Auflösen von Gleichung (6.111) nach der Stellgrößenänderung ergibt schließlich die optimale Stellfolge $\Delta \mathbf{u}^*$ bezüglich der gewählten Kostenfunktion.

$$\Delta \mathbf{u}^* = \underbrace{(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u)}_{\mathbf{C}}^{-1} \cdot [\mathbf{H}^T \underbrace{(\mathbf{r} - \mathbf{F} \mathbf{x}(k) - \mathbf{G} \mathbf{v})}_{\mathbf{d}} - \alpha \mathbf{H}_u^T \underbrace{(\mathbf{F}_s \mathbf{x}(k) + \mathbf{H}_w \mathbf{r}_2 + \mathbf{G}_s \mathbf{v})}_{\mathbf{h}}] \quad (6.112)$$

Das Minimum kann bestätigt werden, wenn die zweite Ableitung der Gütefunktion existiert und positiv ist.

$$\frac{\partial^2 J(\Delta \mathbf{u})}{\partial^2 \Delta \mathbf{u}} = 2 (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u) > 0 \quad (6.113)$$

Die zweite Ableitung von J ist unabhängig von $\Delta \mathbf{u}$ für $\lambda, \alpha > 0$ stets positiv definit (Beweis in Kap. 6.5, Gleichung (6.127)). Zur Berechnung der Matrix $\mathbf{C}^{-1}$ wurde bei der praktischen Realisierung in Kap. 8 das Gauß-Jordan-Verfahren verwendet. Die Matrixdimension ist relativ klein ($N_u \times N_u$), so dass mit einer Vorwärtselimination schnell die inverse Matrix bestimmt werden kann [16].

Das erste Element des optimalen Stellvektors wird auf die Regelstrecke geschaltet. Es berechnet sich durch Multiplikation mit einem *Ausblendvektor*.

$$\Delta u(k)^* = [1, 0, \dots, 0] \cdot \Delta \mathbf{u}^* = \mathbf{T} \cdot \mathbf{r} - \mathbf{T}_2 \cdot \mathbf{r}_2 - \mathbf{K} \cdot \mathbf{x}(k) - \mathbf{L} \cdot \mathbf{v} \quad (6.114)$$

Die Matrizen $\mathbf{T}$, $\mathbf{T}_2$, $\mathbf{K}$ und $\mathbf{L}$ ergeben sich zu:

$$\begin{aligned} \mathbf{T}' &= [1, 0, \dots, 0] \cdot (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u)^{-1} \\ \mathbf{T} &= \mathbf{T}' \cdot \mathbf{H}^T \end{aligned} \quad (6.115)$$

$$\mathbf{T}_2 = \mathbf{T}' \cdot \alpha \mathbf{H}_u^T \mathbf{H}_w \quad (6.116)$$

$$\mathbf{K} = \mathbf{T}' \cdot (\mathbf{H}^T \mathbf{F} + \alpha \mathbf{H}_u^T \mathbf{F}_s) \quad (6.117)$$

$$\mathbf{L} = \mathbf{T}' \cdot (\mathbf{H}^T \mathbf{G} + \alpha \mathbf{H}_u^T \mathbf{G}_s) \quad (6.118)$$

Das Regelgesetz mit Führungsintegrator ist ähnlich zu Gleichung (6.77). Der zusätzliche Term $\mathbf{T}_2 \cdot \mathbf{r}_2$ rührt vom aufsummierten Regelfehler her. Es handelt sich wieder um ein Zustandsregelgesetz, das jedoch zukünftige Sollwerte und den Einfluss der Nichtlinearität berücksichtigt.

6.5 Optimierung unter Berücksichtigung von Beschränkungen

Eine charakteristische Stärke der prädiktiven Regelung ist die Möglichkeit, verschiedene Prozessbeschränkungen schon während des Reglerentwurfes direkt berücksichtigen zu können. Nach [58, 70, 87] ist die MPC die einzige praxistaugliche Regelstrategie, die hierzu in der Lage ist. Im Allgemeinen existieren in jeder technischen Regelungsaufgabe kritische Prozesszustände, in welchen zwar alle Beschränkungen eingehalten werden, die jedoch in

unerlaubte Zustände übergehen können. Wegen der begrenzten Stellgröße besteht nur ein beschränkter Einfluss auf den Prozess. Daher kann unter Umständen nicht mehr verhindert werden, dass wegen der Eigendynamik der Regelstrecke die kritischen Zustände in unerlaubte Zustände münden, welche die Begrenzungen verletzen. Der prädiktive Regler kann die kritischen Zustände voraussehen und daher auch vermeiden, so dass die Einhaltung aller Beschränkungen stets garantiert ist, wenn die Optimierungsaufgabe unter den gegebenen Nebenbedingungen lösbar ist. Dieser Vorteil wird besonders vor dem Hintergrund deutlich, dass die Vernachlässigung der in Realität vorhandenen Stellgrößenbeschränkungen bei der Reglerimplementierung zum Stabilitätsverlust führen kann. Besonders wenn Prozesse nahe an ihren zulässigen Grenzwerten betrieben werden sollen, muss die Einhaltung aller Beschränkungen garantiert sein [70]. Bei Verwendung von nicht-prädiktiven Reglern muss im Allgemeinen eine ad-hoc-Lösung gefunden werden, wie die Regelung auf einen gegebenen Prozess so abgestimmt werden kann, dass solche Stabilitätsprobleme nicht auftreten. Als Ergebnis erhält man meist eine sehr aufwändige, spezielle und damit auch teure Anpassung des Reglers, deren Kosten aufgrund der Speziallösung nicht auf viele verschiedene Problemstellungen verteilt werden können. Die Technik der Anti-Wind-up-Regelung ermöglicht durch Festhalten von Integralanteilen (Sättigung des Integrators) zwar eine Berücksichtigung von Stellgrößenbeschränkungen [87, 29], jedoch können andere Arten von Begrenzungen damit nicht eingehalten werden. Dagegen besteht beim prädiktiven Regler die Möglichkeit, auch Zustands- und Ausgangsgrößenbeschränkungen zu berücksichtigen. Dadurch können Prozesse besser im optimalen Bereich und enger an maximal zulässigen Grenzwerten betrieben werden, was häufig wirtschaftliche Verbesserungen mit sich bringt.

In der Praxis sind alle Prozesse physikalischen Beschränkungen unterworfen. Stellglieder können ihre maximale Ausgangsgröße nicht überschreiten. So kann ein Ventil beispielsweise maximal nur vollständig geöffnet sein. Eine weitere Steigerung des Durchflusses ist durch eine Einwirkung des Stellgliedes dann nicht mehr zu erzielen. Solche **Stellgrößenbeschränkungen** sind in allen technischen Stellgliedern zu finden. Weiterhin besteht die Möglichkeit, dass die Stellgeschwindigkeit begrenzt ist. Um beim Beispiel des Ventils zu bleiben, ist zu beachten, dass die Stellaktion nicht beliebig schnell verlaufen kann. Vielmehr dauert es eine gewisse Zeitspanne, bis ein geschlossenes Ventil vollständig geöffnet ist. Die begrenzte Geschwindigkeit der **Stellgrößenänderung** ist aber nicht in allen Anwendungen relevant. Gerade in hochdynamischen Antrieben kann ein Motormoment in sehr kurzer Zeit, im Vergleich zur Dynamik des mechanischen Systems, nahezu beliebige Änderungen durchführen.

Die Beschränkungen von Systemzuständen und Ausgangssignalen sind dagegen meist eine Folge wirtschaftlicher und verfahrenstechnischer Überlegungen. Die Forderungen nach maximaler Effizienz, Schonung der Anlage, gleichmäßiger Produktqualität, Einhaltung sicherheitstechnischer und umweltschonender Vorgaben zieht eine Reihe von Begrenzungen nach sich. Als konkrete Beispiele können Antriebswellen angeführt werden, deren Torsionswinkel unterhalb eines maximal zulässigen Wertes bleiben muss, um einen Bruch zu verhindern. Ebenso dürfen Drücke und Temperaturen in Rührkesselreaktoren vorgegebene Werte nicht überschreiten. Auch die Unterschreitung einer minimalen Temperatur kann unerwünscht sein, da dies bei chemischen Reaktionen ineffizienten Betrieb oder die Entstehung von schädlichen Nebenprodukten bedeuten kann. Die angesprochenen Begrenzungen können sich im Optimierungsproblem entweder als **Zustandsbeschränkungen** oder als

Ausgangsbeschränkungen auswirken.

Die Berücksichtigung von Beschränkungen wird als Ungleichungsnebenbedingung (UNB) oder als Gleichungsnebenbedingung (GNB) in die Optimierungsaufgabe eingebracht. Es werden im Folgenden Ungleichungsnebenbedingungen berücksichtigt.

$$\begin{aligned}
 u^{min} &\leq u(k+j) \leq u^{max} & 0 \leq j \leq N_u \\
 \Delta u^{min} &\leq \Delta u(k+j) \leq \Delta u^{max} & 0 \leq j \leq N_u \\
 \mathbf{x}^{min} &\leq \mathbf{x}(k+j) \leq \mathbf{x}^{max} & 1 \leq j \leq N_2 \\
 y^{min} &\leq y(k+j) \leq y^{max} & 1 \leq j \leq N_2
 \end{aligned} \tag{6.119}$$

Zur effizienten Lösung mittels quadratischer Programmierung [48, 86] muss das Gütefunktional leicht modifiziert werden. Es wird hier nur die prädiktive Regelung mit Integralanteil betrachtet, die Regelung ohne Integralanteil erhält man für $\alpha = 0$. Das Gütefunktional in Vektor-Matrix-Schreibweise lautet nach Gleichung (6.107):

$$\begin{aligned}
 J(\Delta \mathbf{u}) &= \Delta \mathbf{u}^T (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u) \Delta \mathbf{u} - 2 \mathbf{d}^T \mathbf{H} \Delta \mathbf{u} \\
 &+ 2 \alpha \mathbf{h}^T \mathbf{H}_u \Delta \mathbf{u} + \mathbf{d}^T \mathbf{d} + \alpha \mathbf{h}^T \mathbf{h}
 \end{aligned} \tag{6.120}$$

Für die Vektoren $\mathbf{d}$ und $\mathbf{h}$ gilt:

$$\mathbf{d} = \mathbf{r} - \mathbf{F} \mathbf{x}(k) - \mathbf{G} \mathbf{v} \tag{6.121}$$

$$\mathbf{h} = \mathbf{F}_s \mathbf{x}(k) + \mathbf{H}_w \mathbf{r}_2 + \mathbf{G}_s \mathbf{v} \tag{6.122}$$

Die Terme $\mathbf{d}^T \mathbf{d}$ und $\mathbf{h}^T \mathbf{h}$ sind unabhängig von der Stellgrößenänderung $\Delta \mathbf{u}$ und können deshalb bei der Bestimmung des Minimums der Gütefunktion entfallen. Es ist nicht der Wert des Minimums von Interesse, sondern das Argument der Gütefunktion im Minimum. Durch die beiden Terme verändert sich zwar die Steigung der Gütefunktion, dies hat aber keinen Einfluss auf den Ort des Minimums. Zusätzlich wird Gleichung (6.120) noch mit dem Faktor $1/2$ multipliziert, um sie in die Standardform der quadratischen Programmierung zu überführen. Auch dadurch verändert sich der Ort des Minimums nicht. Es wird äquivalent das Gütemaß J^* zu Berechnung der Stellgrößenfolge benutzt.

$$J^*(\Delta \mathbf{u}) = \frac{1}{2} \Delta \mathbf{u}^T \underbrace{(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u)}_{\mathbf{C}} \Delta \mathbf{u} - \underbrace{(\mathbf{d}^T \mathbf{H} - \alpha \mathbf{h}^T \mathbf{H}_u)}_{\mathbf{f}^T} \Delta \mathbf{u} \tag{6.123}$$

Gleichung (6.123) kann nun an Standardalgorithmen zur Minimierung übergeben werden. Für die praktische Umsetzung auf dem Echtzeitsystem xPC-Target wurde der Algorithmus aus [86] als C-Programm in Simulink eingebunden. Ein eindeutiges Minimum des quadratischen Programms existiert nur, wenn das Optimierungsproblem streng konvex ist; andernfalls ist die Existenz lokaler Minima nicht ausgeschlossen. Das quadratische Programm ist genau dann streng konvex, wenn die Matrix des quadratischen Terms, $\mathbf{C} = \mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u$, positiv definit und symmetrisch ist [24]. Dazu muss für die zugeordnete quadratische Form gelten:

$$\Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u} > 0 \quad \forall \Delta \mathbf{u} \neq \mathbf{0} \tag{6.124}$$

$$\Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u} = 0 \quad \text{nur für } \Delta \mathbf{u} = \mathbf{0} \tag{6.125}$$

Beweis von Symmetrie und positiver Definitheit:

1. Die Matrix $\mathbf{C}$ ist symmetrisch, wenn ihre Transponierte wieder sie selbst ergibt.

$$\begin{aligned}\mathbf{C}^T &= (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u)^T = (\mathbf{H}^T \mathbf{H})^T + \lambda \mathbf{E}^T + \alpha (\mathbf{H}_u^T \mathbf{H}_u)^T \\ &= \mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u = \mathbf{C}\end{aligned}\quad (6.126)$$

2. Die quadratische Form $J(\Delta \mathbf{u}) = \Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u}$ bzw. die symmetrische Matrix $\mathbf{C}$ heißt positiv definit, wenn aus $\Delta \mathbf{u} \neq 0$ stets $J(\Delta \mathbf{u}) > 0$ folgt. Zerlegt man den Term in seine Summanden, so ist leicht ersichtlich, dass die Teilmatrizen $\mathbf{H}^T \mathbf{H}$, $\lambda \mathbf{E}$ und $\alpha \mathbf{H}_u^T \mathbf{H}_u$ positiv semidefinit bzw. positiv definit sind. Der gesamte Term ergibt somit eine positiv definite Matrix.

$$\begin{aligned}\Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u} &= \Delta \mathbf{u}^T (\mathbf{H}^T \mathbf{H} + \lambda \mathbf{E} + \alpha \mathbf{H}_u^T \mathbf{H}_u) \Delta \mathbf{u} \\ &= \Delta \mathbf{u}^T (\mathbf{H}^T \mathbf{H}) \Delta \mathbf{u} + \Delta \mathbf{u}^T (\lambda \mathbf{E}) \Delta \mathbf{u} + \Delta \mathbf{u}^T (\alpha \mathbf{H}_u^T \mathbf{H}_u) \Delta \mathbf{u} \\ &= (\mathbf{H} \Delta \mathbf{u})^T \mathbf{H} \Delta \mathbf{u} + \lambda \Delta \mathbf{u}^T \mathbf{E} \Delta \mathbf{u} + \alpha (\mathbf{H}_u \Delta \mathbf{u})^T \mathbf{H}_u \Delta \mathbf{u} \\ &= \|\mathbf{H} \Delta \mathbf{u}\|^2 + \lambda \cdot \|\mathbf{E} \Delta \mathbf{u}\|^2 + \alpha \cdot \|\mathbf{H}_u \Delta \mathbf{u}\|^2 > 0 \quad \forall \Delta \mathbf{u} \neq 0\end{aligned}\quad (6.127)$$

Die Beträge der Matrix-Vektor-Produkte $\|\mathbf{H} \Delta \mathbf{u}\|$, $\|\mathbf{E} \Delta \mathbf{u}\|$ und $\|\mathbf{H}_u \Delta \mathbf{u}\|$ sind immer positiv. Für $\lambda, \alpha > 0$ ist demnach Gleichung (6.127) immer erfüllt.

Damit ist gezeigt, dass für die gewünschte Konvexitätseigenschaft des quadratischen Optimierungsproblems besteht, d.h. der Algorithmus findet immer das gesuchte globale Optimum.

Zusätzlich muss jedoch noch die Frage nach der Erreichbarkeit gestellt werden. Im Falle zu restriktiver Beschränkungen existiert keine Lösung $\Delta \mathbf{u}^*$, die unter strikter Einhaltung aller Nebenbedingungen (6.119) die Gütefunktion (6.123) minimiert. Dann kann der Optimierungsalgorithmus nur eine Lösung ermitteln, welche die Nebenbedingungen möglichst wenig verletzt. Die fehlende Erreichbarkeit ist in technischen Anwendungen eine Folge zu anspruchsvoller Forderungen an die Regelung bzw. falscher Dimensionierungen der Stellglieder. Beispielsweise kann die Lastmasse eines Zweimassensystems nicht beliebig stark beschleunigt werden, wenn das Torsionsmoment in der Übertragungswelle und das Antriebsmoment des Motors begrenzt sind. In solchen Fällen ist zu überdenken, ob der gewünschte Sollwertverlauf eine technisch sinnvolle Anforderung an den Prozess und das Stellglied darstellt, oder ob das Stellglied den hohen Ansprüchen angepasst werden muss.

6.5.1 Stellgrößenbeschränkungen

Stellgrößenbeschränkungen werden berücksichtigt, indem ein Satz von $N_u + 1$ Ungleichungsnebenbedingungen (UNB) aufgestellt wird. Jede dieser Ungleichungen ist für einen Zeitpunkt gültig und erzwingt dort die Einhaltung der Beschränkung. Im Stellhorizont $0 \leq j \leq N_u$ muss die Ungleichung

$$u^{min} \leq u(k+j) \leq u^{max} \quad (6.128)$$

eingehalten werden. Da die Entscheidungsvariable des Optimierungsproblems $\Delta \mathbf{u}$ ist, und nicht $\mathbf{u}$, müssen die UNB in der Variablen $\Delta \mathbf{u}$ formuliert werden.

$$\mathbf{N}_u \Delta \mathbf{u} \leq \mathbf{g}_u \quad (6.129)$$

Dazu werden, ausgehend von der letzten Stellgröße $u(k-1)$, die jeweiligen Änderungen aufsummiert.

$$u(k) = u(k-1) + \Delta u(k) \quad (6.130)$$

$$\vdots$$

$$u(k+j) = u(k-1) + \sum_{i=0}^j \Delta u(k+i) \quad (6.131)$$

Durch Einsetzen von $u(k+j)$ in Gleichung (6.128) erhält man die beiden UNB,

$$\sum_{i=0}^j \Delta u(k+i) \leq u^{max} - u(k-1) \quad \forall \quad 0 \leq j \leq N_u \quad (6.132)$$

$$-\sum_{i=0}^j \Delta u(k+i) \leq u(k-1) - u^{min} \quad \forall \quad 0 \leq j \leq N_u \quad (6.133)$$

die formal zu einer UNB zusammengefasst werden können. Man erhält damit die in Gleichung (6.129) geforderte Form.

$$\underbrace{\begin{bmatrix} \bar{\mathbf{N}}_u \\ -\bar{\mathbf{N}}_u \end{bmatrix}}_{\mathbf{N}_u} \Delta \mathbf{u} \leq \underbrace{\begin{bmatrix} \mathbf{g}_u^u \\ \mathbf{g}_u^l \end{bmatrix}}_{\mathbf{g}_u} \quad (6.134)$$

Die verwendeten Vektoren und Matrizen ergeben sich zu:

$$\bar{\mathbf{N}}_u = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 1 & 1 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{N_u+1 \times N_u+1} \quad (6.135)$$

$$\mathbf{g}_u^l = \begin{bmatrix} u(k-1) - u^{min} \\ \vdots \\ u(k-1) - u^{min} \end{bmatrix} \in \mathbb{R}^{N_u+1} \quad (6.136)$$

$$\mathbf{g}_u^u = \begin{bmatrix} u^{max} - u(k-1) \\ \vdots \\ u^{max} - u(k-1) \end{bmatrix} \in \mathbb{R}^{N_u+1} \quad (6.137)$$

Bei dieser Vorgehensweise wurden – ausgehend von der Stellgröße $u(k-1)$ – die einzelnen Stellgrößenänderungen betrachtet und der erhaltene Wert mit den maximal möglichen

Grenzen u^{min} und u^{max} verglichen. Zum Zeitpunkt der Berechnung der Stellgrößenänderung $\Delta u(k)$ liegt die Stellgröße $u(k-1)$ des vergangenen Zeitschrittes bereits am Prozesseingang an und ist somit bekannt. Wegen der Erweiterung des Prozessmodelles in Gleichung (6.4) um den Zustand $u(k-1)$ liegt diese Größe als N -tes Element des aktuellen Zustandsvektors $\mathbf{x}(k)$ vor.

6.5.2 Stellgrößenänderungsbeschränkungen

Die Verwendung von langsamen Stellgliedern kann die Berücksichtigung von Stellgrößenänderungsbeschränkungen erfordern. Bauartbedingt können zum Beispiel Schieber und Ventile mehrere Minuten benötigen, um sich von einer Position in eine andere zu bewegen. Daher sollte der Optimierungsalgorithmus im Regler diese Randbedingungen in die Berechnung der Stellgröße mit einbeziehen, so dass keine unrealistisch schnellen Veränderungen der Stellgröße als optimale Lösung berechnet werden.

Innerhalb des gesamten Stellhorizonts $0 \leq j \leq N_u$ muss Gleichung (6.138) eingehalten werden.

$$\Delta u^{min} \leq \Delta u(k+j) \leq \Delta u^{max} \quad (6.138)$$

Durch Einführung einer kompakten vektoriellen Schreibweise ergibt sich:

$$\underbrace{\begin{bmatrix} \bar{\mathbf{N}}_{\Delta \mathbf{u}} \\ -\bar{\mathbf{N}}_{\Delta \mathbf{u}} \end{bmatrix}}_{\mathbf{N}_{\Delta \mathbf{u}}} \Delta \mathbf{u} \leq \underbrace{\begin{bmatrix} \mathbf{g}_{\Delta \mathbf{u}}^u \\ \mathbf{g}_{\Delta \mathbf{u}}^l \end{bmatrix}}_{\mathbf{g}_{\Delta \mathbf{u}}} \quad (6.139)$$

Die Vektoren und Matrizen ergeben sich durch einen Vergleich mit (6.138) zu:

$$\bar{\mathbf{N}}_{\Delta \mathbf{u}} = \mathbf{E} \in \mathbb{R}^{N_u+1 \times N_u+1} \quad (6.140)$$

$$\mathbf{g}_{\Delta \mathbf{u}}^l = \begin{bmatrix} -\Delta u^{min} \\ \vdots \\ -\Delta u^{min} \end{bmatrix} \in \mathbb{R}^{N_u+1} \quad (6.141)$$

$$\mathbf{g}_{\Delta \mathbf{u}}^u = \begin{bmatrix} \Delta u^{max} \\ \vdots \\ \Delta u^{max} \end{bmatrix} \in \mathbb{R}^{N_u+1} \quad (6.142)$$

6.5.3 Zustandsgrößenbeschränkungen

Die Einführung von Zustandsgrößenbeschränkungen kann Qualitäts-, Sicherheits- oder Effizienzgründe haben. Es können natürlich nur Begrenzungen für prädizierte Zustandsgrößen eingehalten werden, da zukünftige Messwerte nicht zur Verfügung stehen. Wie gut die Begrenzungen der tatsächlichen Zustandsgrößen eingehalten werden, hängt maßgeblich von der Güte des verwendeten Prozessmodells ab.

$$\mathbf{x}^{min} \leq \tilde{\mathbf{x}}(k+j) \leq \mathbf{x}^{max} \quad \text{für} \quad 1 \leq j \leq N_2 \quad (6.143)$$

Die Vektoren $\mathbf{x}^{max}$ und $\mathbf{x}^{min}$ beschreiben die Grenzen der einzelnen Prozesszustände. Damit besteht die Freiheit, für jede einzelne Zustandsgröße einen eigenen Arbeitsbereich festzulegen. Da prädizierte Zustandsgrößen in Begrenzungen eingehen, ist die Vorhersage des Verlaufes des Zustandsvektors notwendig, die gemäß der j-Schritt-Prädiktion für den Zustandsvektor aus Kap. 6.1.2 erfolgen kann. Der seinerseits aus den prädizierten Zustandsvektoren zusammengesetzte Vektor

$$\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{\mathbf{x}}(k+1) \\ \vdots \\ \tilde{\mathbf{x}}(k+N_2) \end{bmatrix} \quad \text{mit} \quad \tilde{\mathbf{x}} \in \mathbb{R}^{N_2 \cdot N} \quad (6.144)$$

beschreibt den Verlauf des Zustandsvektors abhängig von den zukünftigen Stellgrößen und kann analog zur Berechnung des Prozessausganges in Kap. 6.1.2 mit einer einzigen Prädiktionsgleichung vorhergesagt werden:

$$\tilde{\mathbf{x}} = \mathbf{F}_z \cdot \mathbf{x}(k) + \mathbf{H}_z \cdot \Delta \mathbf{u} + \mathbf{G}_z \cdot \mathbf{v} \quad (6.145)$$

Der Aufbau der Matrizen $\mathbf{F}_z$, $\mathbf{H}_z$ und $\mathbf{G}_z$ unterscheidet sich von der Bildung der Prädiktionsmatrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ für das Ausgangssignal in den Gleichungen (6.36) bis (6.40). Nachdem der Verlauf des Zustandsvektors gesucht ist, entfällt die Multiplikation mit dem Auskoppelvektor $\mathbf{c}^T$ bei der Berechnung. Eine weitere Änderung betrifft ihre Dimension. Für die Einhaltung der Nebenbedingungen ist das Prozessverhalten ab dem Zeitpunkt $k+1$ zu betrachten. Dies wird erreicht, indem der untere Prädiktionshorizont in den Gleichungen (6.36) bis (6.40) für die Berechnung der Matrizen $\mathbf{F}_z$, $\mathbf{H}_z$ und $\mathbf{G}_z$ auf $N_1 = 1$ gesetzt wird.

$$\mathbf{F}_z \in \mathbb{R}^{N_2 \cdot N \times N} \quad \mathbf{H}_z \in \mathbb{R}^{N_2 \cdot N \times N_u + 1} \quad \mathbf{G}_z \in \mathbb{R}^{N_2 \cdot N \times N_2} \quad (6.146)$$

Die Beschränkung der Zustände lautet damit in vektorieller Notation

$$\underline{\mathbf{x}}^{min} \leq \tilde{\mathbf{x}} \leq \underline{\mathbf{x}}^{max} \quad (6.147)$$

und beinhaltet die Vorgabe der Gleichung (6.143) für alle Zeitschritte bis zum oberen Prädiktionshorizont. Dabei fassen die unterstrichenen Vektoren

$$\underline{\mathbf{x}}^{max} = \begin{bmatrix} \mathbf{x}^{max} \\ \vdots \\ \mathbf{x}^{max} \end{bmatrix} \quad \text{und} \quad \underline{\mathbf{x}}^{min} = \begin{bmatrix} \mathbf{x}^{min} \\ \vdots \\ \mathbf{x}^{min} \end{bmatrix} \quad \in \mathbb{R}^{N_2 \cdot N} \quad (6.148)$$

die Begrenzungen zusammen, die durch die beiden Vektoren

$$\mathbf{x}^{max} = \begin{bmatrix} x_1^{max} \\ \vdots \\ x_N^{max} \end{bmatrix} \quad \text{und} \quad \mathbf{x}^{min} = \begin{bmatrix} x_1^{min} \\ \vdots \\ x_N^{min} \end{bmatrix} \quad \in \mathbb{R}^N \quad (6.149)$$

vorgegeben werden. Aus den Gleichungen (6.147) und (6.145) ergibt sich daher die Formulierung der UNB zu:

$$\mathbf{H}_z \cdot \Delta \mathbf{u} \leq \underline{\mathbf{x}}^{max} - \mathbf{F}_z \cdot \mathbf{x}(k) - \mathbf{G}_z \cdot \mathbf{v} \quad (6.150)$$

$$-\mathbf{H}_z \cdot \Delta \mathbf{u} \leq -\underline{\mathbf{x}}^{min} + \mathbf{F}_z \cdot \mathbf{x}(k) + \mathbf{G}_z \cdot \mathbf{v} \quad (6.151)$$

Die übliche vektorielle Schreibweise lautet dann:

$$\underbrace{\begin{bmatrix} \bar{\mathbf{N}}_{\mathbf{x}} \\ -\bar{\mathbf{N}}_{\mathbf{x}} \end{bmatrix}}_{\mathbf{N}_{\mathbf{x}}} \Delta \mathbf{u} \leq \underbrace{\begin{bmatrix} \mathbf{g}_{\mathbf{x}}^u \\ \mathbf{g}_{\mathbf{x}}^l \end{bmatrix}}_{\mathbf{g}_{\mathbf{x}}} \quad (6.152)$$

mit den Vektoren und Matrizen

$$\bar{\mathbf{N}}_{\mathbf{x}} = \mathbf{H}_z \quad (6.153)$$

$$\mathbf{g}_{\mathbf{x}}^u = \underline{\mathbf{x}}^{max} - \mathbf{F}_z \mathbf{x}(k) - \mathbf{G}_z \mathbf{v} \quad (6.154)$$

$$\mathbf{g}_{\mathbf{x}}^l = -\underline{\mathbf{x}}^{min} + \mathbf{F}_z \mathbf{x}(k) + \mathbf{G}_z \mathbf{v} \quad (6.155)$$

In dieser Schreibweise können die Nebenbedingungen in ein quadratisches Programm integriert werden. Es ist zu beachten, dass die Vektoren $\mathbf{g}_{\mathbf{x}}^u$ und $\mathbf{g}_{\mathbf{x}}^l$ jeweils vom aktuellen Zustandsvektor abhängen und bei jedem Abtastschritt neu bestimmt werden müssen.

6.5.4 Ausgangsgrößenbeschränkungen

Der zulässige Bereich des Prozessausgangs wird durch die Werte y^{min} und y^{max} vorgegeben. Neben derartigen *harten* Begrenzungen, werden in [58, 84] *weiche* Nebenbedingungen diskutiert, die überschritten werden dürfen, dabei allerdings entsprechende Kosten verursachen. Dies führt allerdings zu einem komplizierteren Gütefunktional. Die hier benutzten Begrenzungen lauten

$$y^{min} \leq \tilde{y}(k+j) \leq y^{max} \quad \text{für} \quad 1 \leq j \leq N_2 \quad (6.156)$$

und müssen vom aktuellen Zeitpunkt bis zum Ende des Vorhersagezeitraums erfüllt sein. Es können nur Begrenzungen für prädizierte Ausgangssignale berücksichtigt werden, da zukünftige Messsignale nicht vorliegen. Um die entsprechende Vorhersage des Ausgangssignales zwischen den Zeitpunkten $k+1$ und $k+N_2$ zu erhalten, wird auf die Prädiktionsgleichung zurückgegriffen.

$$\tilde{\mathbf{y}} = \mathbf{F}_a \cdot \mathbf{x}(k) + \mathbf{H}_a \cdot \Delta \mathbf{u} + \mathbf{G}_a \cdot \mathbf{v} \quad (6.157)$$

Die Bildung der Matrizen $\mathbf{F}_a$, $\mathbf{H}_a$ und $\mathbf{G}_a$ erfolgt gemäß den Gleichungen (6.36) bis (6.40) mit einem unteren Prädiktionshorizont von $N_1 = 1$, damit die Nebenbedingungen im geforderten Zeitintervall erfüllt werden können. Wenn der Regler mit einem unteren Horizont arbeiten soll, der größer als eins ist, dann können die Prädiktionsmatrizen $\mathbf{F}$, $\mathbf{H}$ und $\mathbf{G}$ als Untermatrizen aus $\mathbf{F}_a$, $\mathbf{H}_a$ und $\mathbf{G}_a$ entnommen werden und brauchen nicht separat berechnet werden. Der Vektor

$$\tilde{\mathbf{y}} = [\tilde{y}(k+1) \quad \dots \quad \tilde{y}(k+N_2)]^T \in \mathbb{R}^{N_2} \quad (6.158)$$

enthält alle prädizierten Ausgangssignale, die Vektoren

$$\mathbf{y}^{max} = \begin{bmatrix} y^{max} \\ \vdots \\ y^{max} \end{bmatrix} \quad \text{und} \quad \mathbf{y}^{min} = \begin{bmatrix} y^{min} \\ \vdots \\ y^{min} \end{bmatrix} \in \mathbb{R}^{N_2} \quad (6.159)$$

legen die Beschränkungen fest. Aus der Prädiktionsgleichung (6.157) folgt zusammen mit Gleichung (6.156) die vektorielle Notation der UNB:

$$\mathbf{H}_a \cdot \Delta \mathbf{u} \leq \mathbf{y}^{max} - \mathbf{F}_a \cdot \mathbf{x}(k) - \mathbf{G}_a \cdot \mathbf{v} \quad (6.160)$$

$$-\mathbf{H}_a \cdot \Delta \mathbf{u} \leq -\mathbf{y}^{min} + \mathbf{F}_a \cdot \mathbf{x}(k) + \mathbf{G}_a \cdot \mathbf{v} \quad (6.161)$$

$$\underbrace{\begin{bmatrix} \bar{\mathbf{N}}_y \\ -\bar{\mathbf{N}}_y \end{bmatrix}}_{\mathbf{N}_y} \Delta \mathbf{u} \leq \underbrace{\begin{bmatrix} \mathbf{g}_y^u \\ \mathbf{g}_y^l \end{bmatrix}}_{\mathbf{g}_y} \quad (6.162)$$

Die verwendeten Vektoren und Matrizen ergeben sich zu:

$$\bar{\mathbf{N}}_y = \mathbf{H}_a \quad (6.163)$$

$$\mathbf{g}_y^u = \mathbf{y}^{max} - \mathbf{F}_a \mathbf{x}(k) - \mathbf{G}_a \mathbf{v} \quad (6.164)$$

$$\mathbf{g}_y^l = -\mathbf{y}^{min} + \mathbf{F}_a \mathbf{x}(k) + \mathbf{G}_a \mathbf{v} \quad (6.165)$$

Die Vektoren $\mathbf{g}_y^u$ und $\mathbf{g}_y^l$ hängen vom aktuellen Zustandsvektor ab und erfordern eine Neuberechnung bei jedem Abtastschritt.

6.5.5 Resultierendes Optimierungsproblem

Die verschiedenen Arten von Prozessbeschränkungen für u , Δu , $\mathbf{x}$ und y ergeben jeweils lineare Ungleichungsnebenbedingungen, die alle eine gemeinsame Struktur aufweisen. Aus diesem Grund können sie zu einer einzigen Ungleichungsnebenbedingung zusammengefasst werden.

$$\underbrace{\begin{bmatrix} \mathbf{N}_{\Delta \mathbf{u}} \\ \mathbf{N}_u \\ \mathbf{N}_x \\ \mathbf{N}_y \end{bmatrix}}_{\mathbf{N}} \Delta \mathbf{u} \leq \underbrace{\begin{bmatrix} \mathbf{g}_{\Delta \mathbf{u}} \\ \mathbf{g}_u \\ \mathbf{g}_x \\ \mathbf{g}_y \end{bmatrix}}_{\mathbf{g}} \quad (6.166)$$

Lösungsalgorithmen für quadratische Programme können die folgende Optimierungsaufgabe durch Übergabe der benötigten Matrizen und Vektoren lösen. Als Beispiel für einen solchen Algorithmus sei hier die Matlab-Funktion `quadprog` [48] oder das C-Programm `qld` [86] angeführt. Das quadratische Programm lautet:

$$\begin{aligned} \min_{\Delta \mathbf{u}} J^* &= \min_{\Delta \mathbf{u}} \frac{1}{2} \Delta \mathbf{u}^T \mathbf{C} \Delta \mathbf{u} - \mathbf{f}^T \Delta \mathbf{u} \\ \text{u.B.v. } \mathbf{N} \cdot \Delta \mathbf{u} &\leq \mathbf{g} \end{aligned} \quad (6.167)$$

Wenn die Nebenbedingungen keine leere Menge spezifizieren, das Optimierungsproblem damit lösbar ist, wird wegen der strengen Konvexität der Matrix $\mathbf{C}$ garantiert das globale Minimum bestimmt. Diese Tatsache bedeutet einen gewichtigen Vorteil gegenüber der Verwendung eines nicht-linearisierten Prädiktionsmodelles, das im Allgemeinen auf eine nichtkonvexe Optimierungsaufgabe führt und das Auffinden des globalen Minimums nicht ohne weiteres garantieren kann.

Somit ergibt sich unter Berücksichtigung von Nebenbedingungen (NB) für das prädiktive Regelgesetz die Struktur aus Bild 6.8. Für Prozesse mit nichtlinearer Abhängigkeit vom Zustandsvektor zeigt Bild 6.9 das Funktionsprinzip des Prädiktors (siehe Kap. 6.2.1). Bei Abhängigkeit vom Ausgangssignal kann die a-priori-Annahme entfallen.

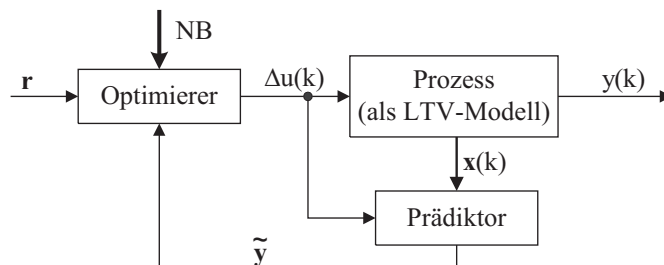


Bild 6.8: *Prädiktiver Regelkreis für nichtlineare Prozesse unter Berücksichtigung von Beschränkungen*

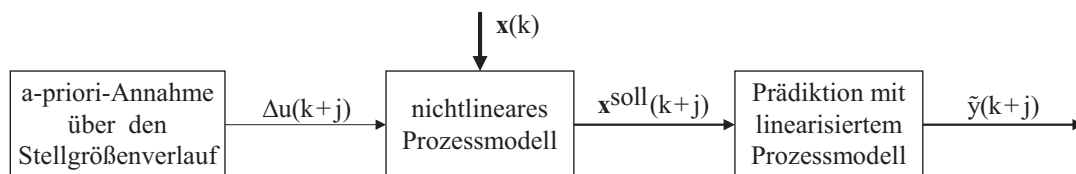


Bild 6.9: *Prinzip des Prädiktors bei Abhängigkeit der Nichtlinearität vom Zustandsvektor*

6.6 Erweiterung auf mehrere Nichtlinearitäten

Auch beim Auftreten mehrerer Nichtlinearitäten innerhalb eines SISO-Prozesses kann die Prädiktion und Optimierung der Stellgrößenfolge mit kleinen Modifikationen angewendet werden. Ohne Beschränkung der Allgemeinheit wird im Folgenden von einem Prozess mit zwei Nichtlinearitäten ausgegangen. Eine Erweiterung auf mehr als zwei Nichtlinearitäten ist möglich. Die beiden Nichtlinearitäten seien von den Zustandsgrößen x_1 und x_2 abhängig und greifen über die Einkoppelvektoren $\mathbf{k}_1$ und $\mathbf{k}_2$ in die Strecke ein. Der Signalflussplan in Bild 6.10 zeigt die Streckenstruktur mit Nichtlinearitäten. Daraus ergibt sich die Zustandsdarstellung in Gleichung (6.168).

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A} \mathbf{x}(k) + \mathbf{b} \Delta u(k) + \mathbf{k}_1 \mathcal{NL}_1(x_1(k)) + \mathbf{k}_2 \mathcal{NL}_2(x_2(k)) \\ y(k) &= \mathbf{c}^T \mathbf{x}(k) + d \Delta u(k) \end{aligned} \quad (6.168)$$

Beide Nichtlinearitäten werden um die jeweiligen Referenztrajektorien x_1^{soll} und x_2^{soll} linearisiert, die entweder durch physikalische Zusammenhänge vorab bekannt sind oder gemäß den Ausführungen in Kap. 6.2 bestimmt wurden. Die Linearisierung von $\mathcal{NL}_1$ und $\mathcal{NL}_2$

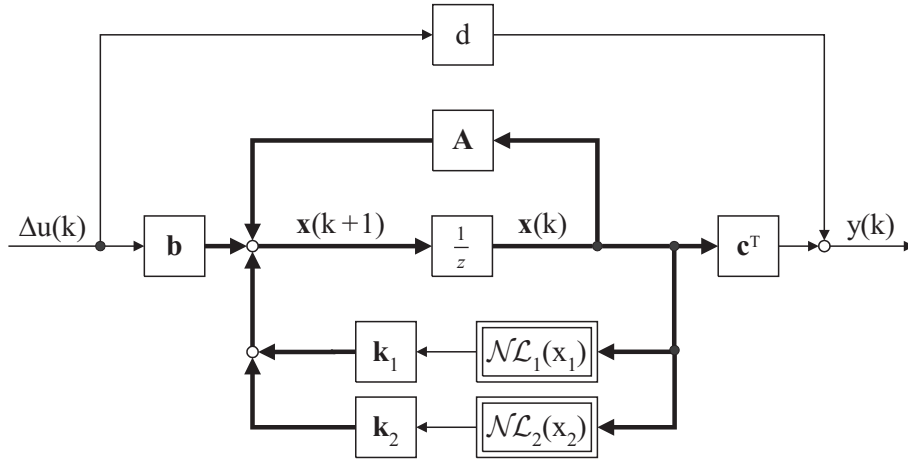


Bild 6.10: Prozess mit zwei Nichtlinearitäten

ergibt:

$$\mathcal{NL}_1(x_1(k)) \approx \mathcal{NL}_1(x_1^{soll}(k)) + \underbrace{\frac{d\mathcal{NL}_1}{dx_1} \Big|_{x_1=x_1^{soll}}}_{\mathcal{NL}_{1x1}} (x_1(k) - x_1^{soll}(k)) \quad (6.169)$$

$$= \mathcal{NL}_1(x_1^{soll}(k)) + \mathcal{NL}_{1x1}(x_1^{soll}(k)) \cdot ([1 \ 0 \ \dots \ 0] \cdot \mathbf{x}(k) - x_1^{soll}(k))$$

$$\mathcal{NL}_2(x_2(k)) \approx \mathcal{NL}_2(x_2^{soll}(k)) + \underbrace{\frac{d\mathcal{NL}_2}{dx_2} \Big|_{x_2=x_2^{soll}}}_{\mathcal{NL}_{2x2}} (x_2(k) - x_2^{soll}(k)) \quad (6.170)$$

$$= \mathcal{NL}_2(x_2^{soll}(k)) + \mathcal{NL}_{2x2}(x_2^{soll}(k)) \cdot ([0 \ 1 \ 0 \ \dots \ 0] \cdot \mathbf{x}(k) - x_2^{soll}(k))$$

Setzt man die Gleichungen (6.169) und (6.170) in die nichtlineare Systemgleichung (6.168) ein, so erhält man die linearisierte Zustandsdarstellung als lineares zeitvariantes System:

$$\mathbf{x}(k+1) = \mathbf{A}(k) \mathbf{x}(k) + \mathbf{b} \Delta u(k) + \mathbf{k}_1 v_1(k) + \mathbf{k}_2 v_2(k) \quad (6.171)$$

$$y(k) = \mathbf{c}^T \mathbf{x}(k) + d \Delta u(k)$$

Die neuen zeitvarianten Größen ergeben sich zu:

$$\begin{aligned} \mathbf{A}(k) &= \mathbf{A} + \mathbf{k}_1 \cdot [1 \ 0 \ \dots \ 0] \cdot \mathcal{NL}_{1x1}(x_1^{soll}(k)) \\ &\quad + \mathbf{k}_2 \cdot [0 \ 1 \ 0 \ \dots \ 0] \cdot \mathcal{NL}_{2x2}(x_2^{soll}(k)) \end{aligned} \quad (6.172)$$

$$v_1(k) = \mathcal{NL}_1(x_1^{soll}(k)) - x_1^{soll}(k) \cdot \mathcal{NL}_{1x1}(x_1^{soll}(k)) \quad (6.173)$$

$$v_2(k) = \mathcal{NL}_2(x_2^{soll}(k)) - x_2^{soll}(k) \cdot \mathcal{NL}_{2x2}(x_2^{soll}(k)) \quad (6.174)$$

Die Prädiktion der zukünftigen Ausgangssignale erfolgt in der selben Weise wie in Kap. 6.2. Als Prädiktionsgleichung erhält man:

$$\tilde{\mathbf{y}} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta \mathbf{u} + \mathbf{G}_1 \cdot \mathbf{v}_1 + \mathbf{G}_2 \cdot \mathbf{v}_2 \quad (6.175)$$

Die Vektoren $\mathbf{v}_1$ und $\mathbf{v}_2$ enthalten die Elemente $v_1(k+j)$ bzw. $v_2(k+j)$ für den gesamten Prädiktionshorizont zwischen $1 \leq j \leq N_2$. Die Matrizen $\mathbf{F}$ und $\mathbf{H}$ werden gemäß den Gleichungen (6.51) und (6.52) gebildet. Da zwei Nichtlinearitäten im System angreifen werden auch zwei $\mathbf{G}$ -Matrizen in der Prädiktionsgleichung benötigt. Sie werden gemäß Gleichung (6.54) gebildet, einmal mit dem Vektor $\mathbf{k}_1$ und einmal mit dem Vektor $\mathbf{k}_2$.

Die Bestimmung der optimalen Stellfolge $\Delta \mathbf{u}$ kann entweder analytisch (ohne Nebenbedingungen, Kap. 6.3) oder numerisch (mit Nebenbedingungen, Kap. 6.5) erfolgen. An den Berechnungen ändert sich grundsätzlich nichts, man muss lediglich in den entsprechenden Gleichungen den Term $\mathbf{G} \cdot \mathbf{v}$ durch den Term $\mathbf{G}_1 \cdot \mathbf{v}_1 + \mathbf{G}_2 \cdot \mathbf{v}_2$ ersetzen. Auf diese Weise können beliebig viele Nichtlinearitäten berücksichtigt werden. Es entstehen in der Prädiktionsgleichung weitere Terme $\mathbf{G}_i \cdot \mathbf{v}_i$ und in Gleichung (6.172) weitere Terme der Form $\mathbf{k}_i \cdot [0 \ 1 \ 0 \ \dots \ 0] \cdot \mathcal{N}_{ixi}(x_i^{sol}(k))$. Der Vektor $[0 \ 1 \ 0 \ \dots \ 0]$ besitzt an der Position i eine 1, wenn die Nichtlinearität den Zustand i als Eingangssignal besitzt. Entscheidend ist aber immer, dass die Eingangsgrößen aller Nichtlinearitäten x_i^{sol} als Solltrajektorien innerhalb des kompletten Prädiktionshorizonts bekannt sein müssen oder berechnet werden können. Das stellt in der Praxis das größte Problem dar.

7 Stabilität prädiktiver Regelungen

Die Hauptanforderung an technisch brauchbare Regelungen ist die Stabilität des geschlossenen Regelkreises. Während die Stabilität linearer Regelkreise durch bekannte algebraische oder geometrische Verfahren (Eigenwerte, Nyquist-Kriterium, Bode-Diagramm [25]) analysiert werden kann, muss zur Stabilitätsuntersuchung nichtlinearer Systeme auf andere Verfahren zurückgegriffen werden. Lyapunov veröffentlichte bereits 1893 eine Methode zum Stabilitätsnachweis nichtlinearer dynamischer Systeme [77]. Die Grundidee beruht auf der Tatsache, dass ein mechanisches System asymptotisch stabil sein muss, wenn seine Energie über der Zeit abnimmt. Schließlich abstrahierte er vom Energiegedanken und formulierte die folgende Stabilitätsbedingung (siehe Kap. 3.2.6.1):

Definition 7.1: Stabilität nach Lyapunov

Ein dynamisches System ist genau dann stabil, wenn eine beschränkte Auslenkung aus einem Ruhezustand zu einer beschränkten Systemantwort führt. Das System ist genau dann asymptotisch stabil, wenn aus einer beschränkten Auslenkung aus einer Ruhelage folgt, dass es für $t \rightarrow \infty$ wieder zur Ruhelage zurückkehrt.

Zur Überprüfung der Stabilität kann der Satz von Lyapunov herangezogen werden.

Satz 7.1: Satz von Lyapunov

Ein dynamisches System $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ besitze die Ruhelage $\mathbf{x} = 0$. Es existiere im gesamten Zustandsraum eine positiv definite Lyapunov-Funktion $V(\mathbf{x})$, die nur für den Nullzustand den Wert null annimmt. Wenn die zeitliche Ableitung $\dot{V}(\mathbf{x})$ kleiner null für alle $\mathbf{x} \neq \mathbf{0}$ ist, dann ist die Ruhelage asymptotisch stabil.

Aufbauend auf diesem Ansatz erfolgt die Stabilitätsuntersuchung für die Mehrzahl der nichtlinearen MPC-Verfahren. Nach [68] bestehen seit Ende der 80er Jahre Bemühungen, einen Stabilitätsbeweis für nichtlineare prädiktive Regelkreise zu finden. Nevistić [50] bezeichnet den theoretischen Hintergrund von Stabilitätsuntersuchungen als *proven to be a complicated and difficult issue* und nennt folgende Gründe für die Schwierigkeit:

- Prozessbeschränkungen führen zu einem Optimierungsproblem mit Nebenbedingungen. Daraus ergibt sich eine nichtlineare Regelung, auch wenn der Prozess und das Prädiktionsmodell lineare Dynamiken aufweisen.
- Es existiert keine explizite Formulierung des Regelgesetzes, wenn Beschränkungen berücksichtigt werden. Eine analytische Beschreibung des Regelgesetzes ist allerdings für die meisten Stabilitätsuntersuchungen notwendig.

Eine wichtige Problematik, die zur Instabilität führen kann, sind die endlichen Stell- und Prädiktionshorizonte. Prädiktive Regelungen mit einer endlichen Gütefunktion, die nur

einen Teil des in Wirklichkeit unendlichen Systemverhaltens enthält, erzeugen im Sinne dieses endlichen Horizonts optimale Stellgrößenverläufe. Über diesen endlichen Horizont hinaus kann keine Aussage über konvergentes oder divergentes Verhalten gemacht werden. Aus einer endlichen Gütefunktion heraus ist es daher nicht möglich, eine asymptotische Stabilitätsaussage zu treffen. Wegen des sich mitbewegenden Horizonts muss zudem bei jedem Abtastschritt ein neues Optimierungsproblem gelöst werden, das gegebenenfalls sogar unlösbar sein kann.

Aus den genannten Schwierigkeiten heraus, wurde die Stabilität nichtlinearer prädiktiver Regelkreise für industrielle Anwendungen bislang durch trial-and-error-Vorgehensweisen bei der Reglereinstellung erreicht. Die aktuelle Literatur zur prädiktiven Regelung widmet sich daher eingehend den theoretischen Stabilitätsuntersuchungen, welche für eine bestimmte Wahl der Reglerparameter die Stabilität der Regelung gewährleisten. Es muss an dieser Stelle aber auch betont werden, dass neben Stabilität gutes Führungs- und Störverhalten ebenso wichtige Anforderungen an eine Regelung darstellen. Bei Stabilitätsbeweisen sind daher Bedingungen gesucht, die instabiles Verhalten ausschließen und gleichzeitig noch genügend Freiheitsgrade zur Verfügung stellen, um weitere Anforderungen an die Regelung zu erfüllen.

Zur prinzipiellen Vorgehensweise wird ein modifiziertes Gütefunktional betrachtet, bevor in Kap. 7.4 die Stabilität der in dieser Arbeit entwickelten Regelung untersucht wird. Anstelle des Regelfehlers $r(k+j) - \tilde{y}(k+j)$ wird der gewichtete Betrag des Zustandsvektors für die Berechnung der Kosten aufsummiert. Zusätzlich wird der Vorhersagezeitraum bis in die unendlich entfernte Zukunft ausgedehnt. Die modifizierte Kostenfunktion lautet somit:

$$J^\infty(\Delta \mathbf{u}) = \sum_{j=1}^{\infty} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u(k+j) \quad (7.1)$$

Mit der positiv definiten Matrix $\mathbf{Q}$ ist eine Gewichtung der einzelnen Prozesszustände x_i möglich. Es wird die Stabilität des Ursprungs als Ruhelage untersucht. Nachdem jede Ruhelage durch eine Koordinatentranslation in den Ursprung verschoben werden kann, ist hierdurch keine Einschränkung auf spezielle Sollwerte gegeben. Der Prozess ist in allgemeiner nichtlinearer Zustandsform gegeben.

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), u(k)) \quad (7.2)$$

7.1 Stabilitätsbeweis durch monotone Abnahme der Kostenfunktion

Für eine prädiktive Regelung auf der Grundlage der Gütefunktion (7.1) wird nun die asymptotische Stabilität nach Lyapunov gezeigt. Dazu muss zum aktuellen Zeitpunkt Erreichbarkeit gegeben sein, d.h. es existiert eine Lösung des Optimierungsproblems unter (hier nicht explizit angegebenen) Nebenbedingungen. Unter den drei Voraussetzungen, dass

- der Prozess und das Prädiktionsmodell exakt übereinstimmen,
- der Zustandsvektor $\mathbf{x}(k)$ genau bekannt ist,

- keine Störungen wirken,

kann das Gütefunktional als Lyapunovfunktion verwendet werden. Das Minimum der Kostenfunktion ist mit J^* bezeichnet. Es wird nun gezeigt, dass die optimalen Kosten von einem zum nächsten Zeitschritt stets abnehmen und beschränkt sind. Eine Summe über unendlich viele positive Terme kann nur dann beschränkt sein, wenn die einzelnen Summanden gegen null konvergieren. Aufgrund der monotonen Kostenreduktion und der Beschränktheit der Gütefunktion muss der Zustandsvektor also gegen null streben, woraus asymptotische Stabilität des Ursprungs als Ruhelage folgt.

Zum Zeitpunkt k ergeben sich die minimalen Kosten zu $J^*(k)$. Wegen des zurückweichenden Horizontes muss zum Zeitpunkt $k + 1$ der Zustand $\tilde{\mathbf{x}}(k + 1)$ nicht mehr bei der Berechnung der Kosten $J^*(k + 1)$ berücksichtigt werden. Am Ende des Horizontes kann dagegen kein neuer Zustand in die Bewertung mit aufgenommen werden, da sich der Horizont bis in die unendlich entfernte Zukunft erstreckt und somit bereits alle zukünftigen Zustände gewichtet. Aus diesem Grund bewirkt der erste Summand der Gütefunktion (7.1) eine Kostenreduktion zwischen dem aktuellen Zeitpunkt k und dem darauf folgenden Zeitschritt $k + 1$ von $\tilde{\mathbf{x}}^T(k + 1)\mathbf{Q}\tilde{\mathbf{x}}(k + 1)$. Ebenso verursacht der zurückweichende Horizont eine Verringerung des zweiten Summanden um die gewichtete Stellgrößenänderung $\lambda\Delta u^2(k)$. Unter der Annahme, dass nach Ablauf des endlichen Stellhorizontes N_u , d.h. ab dem Zeitpunkt $k + N_u + 1$ keine Stellgrößenänderungen mehr auftreten, kommen für den Zeitschritt $k + 1$ keine zusätzlichen Kosten durch die neu erfasste Stellgrößenänderung mehr hinzu. Insgesamt ergibt sich so eine Kostenreduktion von

$$\Delta J = \tilde{\mathbf{x}}^T(k + 1)\mathbf{Q}\tilde{\mathbf{x}}(k + 1) + \lambda\Delta u^2(k) - \underbrace{\lambda\Delta u^2(k + N_u + 1)}_{=0} \quad (7.3)$$

zwischen zwei Zeitschritten. Für den Fall einer nicht verschwindenden Stellgrößenänderung zum Zeitpunkt $k + N_u + 1$, ist die Berücksichtigung der zusätzlichen Stelländerung $\Delta u(k + N_u + 1)$ erforderlich, die Kostenreduktion wird geringer. Allerdings kommt hierdurch ein neuer Freiheitsgrad im Optimierungsproblem hinzu, der zur weiteren Verkleinerung der Summe $\sum_{j=N_u+1}^{\infty} \tilde{\mathbf{x}}^T(k + j)\mathbf{Q}\tilde{\mathbf{x}}(k + j)$ verwendet werden kann. Zum Zeitpunkt $k + N_u + 1$ wird eine Stellgrößenänderung vom Optimierungsalgorithmus also nur dann zugelassen, wenn die Reduktion der Summe über die zukünftigen Zustandsvektoren die benötigte Bestrafung der zusätzlichen Stelländerung überwiegt. Andernfalls führt eine Stelländerung von $\Delta u(k + N_u + 1) = 0$ zu geringeren Gesamtkosten und wird vom Optimierer auch ausgewählt. Demnach ruft die Stelländerung $\Delta u(k + N_u + 1) \neq 0$ insgesamt eine stärkere Kostenabnahme

$$\Delta J \geq \tilde{\mathbf{x}}^T(k + 1)\mathbf{Q}\tilde{\mathbf{x}}(k + 1) + \lambda\Delta u^2(k) \quad (7.4)$$

hervor, wodurch sich die minimalen Kosten für den folgenden Zeitschritt $k + 1$ zu

$$J^*(k + 1) \leq J^*(k) - \tilde{\mathbf{x}}^T(k + 1)\mathbf{Q}\tilde{\mathbf{x}}(k + 1) - \lambda\Delta u^2(k) \quad (7.5)$$

abschätzen lassen. Zu jedem Zeitschritt gilt, dass die Kosten des nächsten Zeitschritts geringer sind als die aktuellen Kosten. Die quadratische Kostenfunktion ist positiv definit und weist nur für $\tilde{\mathbf{x}} = \mathbf{0}$ eine Nullstelle auf, die gleichzeitig auch das globale Minimum bildet. Durch die gezeigte stetige Abnahme der Kostenfunktion und mit der Tatsache, dass die

anfänglichen Kosten beschränkt sind ($J^* < \infty$), wenn der Prozess mit beschränkten Anfangszuständen startet, streben die Systemzustände in den Ursprung des Zustandsraumes und das asymptotisch stabile Verhalten des Regelkreises ist gezeigt.

Allerdings ist ein solcher Ansatz wegen des unendlichen Prädiktionshorizontes für die praktische Anwendung in realen Regelungsproblemen nicht einsetzbar. Es entstünden unendlich dimensionale Optimierungsprobleme, die praktisch nicht handhabbar sind.

7.2 Endzustandsbeschränkung

Um die Problematik eines unendlichen Prädiktionshorizontes zu umgehen, kann eine Endzustandsbeschränkung der Art

$$\tilde{\mathbf{x}}(k + N_2 + 1) = \mathbf{0} \quad (7.6)$$

eingeführt werden [85]. Unter der Voraussetzung, dass der Ursprung eine Gleichgewichtslage des Prozesses darstellt, müssen keine Stellgrößenänderungen mehr aufgebracht werden, um den Zustandsvektor für alle Zeiten im Ursprung zu halten. Es folgt aus der exakten Einhaltung einer solchen Nebenbedingung, dass ab dem Zeitpunkt $k + N_2$ keine Kosten mehr hinzukommen, da alle weiteren Summanden der Gütefunktion (7.1) dann gleich null sind und die Summation über die Zustände kann abgebrochen werden. Daher ist die Abschätzung der Kosten, die sich aus der unendlich langen Gütefunktion J^∞ ergeben, mit einer endlichen Summe möglich. Es entsteht eine äquivalente Kostenfunktion die auf einen endlichen Horizont N_2 reduziert ist:

$$J = \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (7.7)$$

Eine Verkürzung des unendlichen Prädiktionshorizontes auf einen endlichen Wert ist zulässig und begrenzt die Dimension des Optimierungsproblems. Die Endzustandsbeschränkung stellt eine Gleichungsnebenbedingung für das Optimierungsproblem dar, die eine Abschätzung des Wertes J^∞ der unendlich weit reichenden Kostenfunktion durch eine Kostenfunktion mit beschränkten Prädiktionshorizont ermöglicht.

$$J^\infty(k) = \sum_{j=1}^{\infty} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) = \quad (7.8)$$

$$= \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (7.9)$$

$$\text{u.B.v. } \tilde{\mathbf{x}}(k + N_2 + 1) = \mathbf{0}$$

Nachteilig an der Forderung nach der exakten Einhaltung der Gleichungsnebenbedingung (7.6) ist, dass dadurch erhebliche, wenn nicht gar unerfüllbare Ansprüche an die benötigte Rechengenauigkeit gestellt werden. Beides erfordert allgemein unendlich viele Iterationen des Optimierungsalgorithmus im Falle nichtlinearer Prozesse [63, 80]. Bei einer

näherungsweisen Einhaltung der Gleichungsnebenbedingung hält die obige Argumentation nicht, was zum Verlust der Stabilitätsgarantie führt. Deshalb ist eine Kostenfunktion mit einer Endzustandsbeschränkung als GNB für die praktische Anwendung unter Modellunsicherheiten nicht geeignet.

7.3 Quasi-Infinite Horizon Regelungskonzept

Eine Weiterentwicklung der Abschätzung der Kosten eines unendlichen Prädiktionshorizonts wurde von Chen und Allgöwer vorgestellt [38, 63]. Dabei wird nicht mehr die Einhaltung einer Gleichungsnebenbedingung gefordert. Vielmehr reicht es aus, wenn der Zustandsvektor in einem Endgebiet $\mathcal{W}$ für Zeiten größer als der obere Prädiktionshorizont zu liegen kommt. Es muss demnach nur eine Ungleichungsnebenbedingung eingehalten werden, was in der Praxis eine wesentlich schwächere Forderung als eine GNB darstellt. Die restlichen Kosten für $k + N_2 + 1$ bis ∞ werden durch eine Endzustandsgewichtung abgeschätzt. Dafür wird die Kostenfunktion in zwei Teile aufgespalten.

$$\begin{aligned} J^\infty(\Delta \mathbf{u}) &= \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \underbrace{\sum_{j=N_2+1}^{\infty} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j)}_{\tilde{\mathbf{x}}^T(k+N_2+1) \mathbf{P} \tilde{\mathbf{x}}(k+N_2+1)} + \lambda \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (7.10) \\ &\leq \tilde{\mathbf{x}}^T(k+N_2+1) \mathbf{P} \tilde{\mathbf{x}}(k+N_2+1) \end{aligned}$$

Wie durch die Klammerung angedeutet, werden die Kosten, die von den Zuständen $\tilde{\mathbf{x}}(k+j)$ mit $N_2 + 1 \leq j \leq \infty$ verursacht werden, durch eine quadratische Form $\tilde{\mathbf{x}}^T \mathbf{P} \tilde{\mathbf{x}}$ abgeschätzt. Zur Berechnung der positiv definiten Matrix $\mathbf{P}$ zur Endzustandsbewertung wird die Rückführverstärkung $\mathbf{K}$ eines fiktiven Zustandsreglers im Endbereich $\mathcal{W}$ verwendet. Die

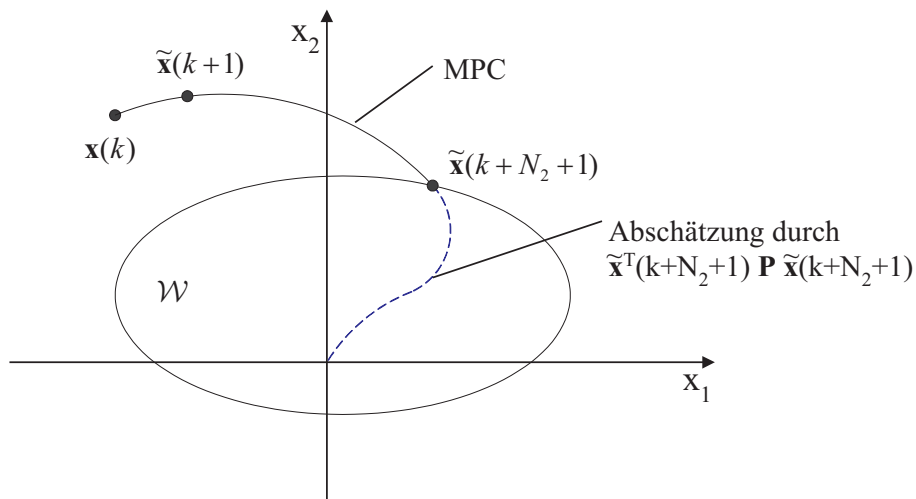


Bild 7.1: *Quasi-Infinite Horizon Regelungsprinzip in der Zustandsebene*

Abschätzung entspricht damit den Kosten, die bei Aufschaltung des Zustandsregelgesetzes

entstehen würden (siehe Bild 7.1). Die Abschätzung

$$\tilde{\mathbf{x}}^T(k + N_2 + 1)\mathbf{P}\tilde{\mathbf{x}}(k + N_2 + 1) \geq \sum_{j=N_2+1}^{\infty} \tilde{\mathbf{x}}^T(k + j)\mathbf{Q}\tilde{\mathbf{x}}(k + j) \quad (7.11)$$

erlaubt zusammen mit einer Zustandsbeschränkung in Form einer Ungleichungsnebenbedingung

$$\tilde{\mathbf{x}}(k + N_2 + 1) \in \mathcal{W} \quad (7.12)$$

wiederum den Abbruch der Summe. Der Wert der unendlich langen Gütefunktion liegt durch die gewählte Abschätzung sicher niedriger als der Wert der folgenden Gütefunktion:

$$\begin{aligned} J(\Delta \mathbf{u}) &= \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k + j)\mathbf{Q}\tilde{\mathbf{x}}(k + j) + \sum_{j=0}^{N_u} \Delta u^2(k + j) + \\ &\quad + \tilde{\mathbf{x}}^T(k + N_2 + 1)\mathbf{P}\tilde{\mathbf{x}}(k + N_2 + 1) \end{aligned} \quad (7.13)$$

$$\text{mit } u = \mathbf{K}\mathbf{x} \quad \text{für } \tilde{\mathbf{x}}(k + N_2 + 1) \in \mathcal{W} \quad (7.14)$$

Damit stellt die verwendete Gütefunktion eine maximale Obergrenze der real anfallenden Kosten dar, indem der nicht durch die Summation erfasste Zeitraum durch eine Abschätzung berücksichtigt wird:

$$J^\infty(\Delta \mathbf{u}) = \sum_{j=1}^{\infty} \tilde{\mathbf{x}}^T(k + j)\mathbf{Q}\tilde{\mathbf{x}}(k + j) + \lambda \sum_{j=0}^{N_u} \Delta u^2(k + j) \leq J(\Delta \mathbf{u}) \quad (7.15)$$

Weil mit einem endlichen Horizont die Kosten der unendlichen Gütefunktion nach oben abgeschätzt werden, wird dieses Verfahren als *Quasi-Infinite Horizon* Regelung bezeichnet. Dabei wird zu keinem Zeitpunkt das lineare Zustandsregelgesetz implementiert, denn der Umschaltzeitpunkt $k + N_2 + 1$ wird wegen des zurückweichenden Horizonts nie erreicht. Die einzige Anforderung an den linearen Zustandsregler ist die Stabilisierung der Regelstrecke in $\mathcal{W}$, da nur dann eine positiv definite Matrix $\mathbf{P}$ zur Abschätzung existiert. Der Nachweis der Stabilität erfolgt auf der Grundlage der stetigen Abnahme der Kostenfunktion $J(\Delta \mathbf{u})$. Details zur Berechnung der Matrix $\mathbf{P}$ und des Endbereiches $\mathcal{W}$ sind [38, 63] zu entnehmen.

Im Vergleich zu prädiktiven Regelungen, die auf Bedingungen des Endzustands in Gleichungsform basieren, sind die Endbedingungen in Ungleichungsform durch numerische Optimierungsverfahren sehr viel einfacher zu erfüllen. Ferner muss bei diesem Verfahren nicht zu jedem Abtastschritt eine optimale Lösung gefunden werden, es genügt, die Abnahme der Kosten sicherzustellen. Dadurch erlangt dieses Verfahren große Relevanz im praktischen Einsatz. Neben der reinen Stabilitätsbetrachtung wurden in den letzten Jahren auch Verfahren zur robusten prädiktiven Regelung von Prozessen mit unsicheren Parametern entwickelt [43].

7.4 Stabilität der entwickelten Regelung

Aufbauend auf den Überlegungen der Kap. 7.1 bis 7.3 wird im Folgenden ein Stabilitätsnachweis des geschlossenen Regelkreises für die prädiktive Regelung mit linearisierten Zustandsraummodellen ohne Berücksichtigung von Prozessbeschränkungen geführt.

Betrachtet wird die Regelung eines ungestörten, nicht sprungfähigen SISO-Prozesses, dessen isolierte Nichtlinearität von der Ausgangsgröße abhängig ist. Der Prozess sei vollständig bekannt und durch das gewählte Prozessmodell ausreichend genau beschrieben. Das Prädiktionsmodell bildet daher das reale zukünftige Prozessverhalten ideal ab. Das um die Referenztrajektorie linearisierte System wird beschrieben durch:

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}(k) \mathbf{x}(k) + \mathbf{b}(k) u(k) + \mathbf{k} v(k) \\ y(k) &= \mathbf{c}^T \mathbf{x}(k)\end{aligned}\quad (7.16)$$

Man beachte, dass das System in dieser Darstellung nicht um den Zustand $u(k-1)$ erweitert wurde, sondern $u(k)$ die Eingangsgröße darstellt, da die Erweiterung für den Stabilitätsnachweis nicht benötigt wird. Es wird die Stabilität einer Ruhelage betrachtet, die durch zeitlich konstante Zustandsgrößen und eine konstante Referenztrajektorie beschrieben wird.

$$r(k+j) = r = \text{const} \quad (7.17)$$

Da die Zeitvarianz der Systemgrößen $\mathbf{A}(k)$, $\mathbf{b}(k)$ und $v(k)$ allein durch eine zeitvariante Referenztrajektorie bedingt ist, sind diese Größen für eine konstante Referenztrajektorie ebenfalls konstant.

$$\mathbf{A}(k) = \mathbf{A} = \text{const} \quad \mathbf{b}(k) = \mathbf{b} = \text{const} \quad v(k) = v = \text{const} \quad (7.18)$$

Gleichung (7.16) stellt demnach ein LTI-System mit konstanter Störgröße dar. Eine konstante Störgröße hat keinen Einfluss auf die Stabilität eines LTI-Systems und kann bei den weiteren Untersuchungen folglich entfallen. Auch ist es unerheblich, welcher stationäre Arbeitspunkt betrachtet wird. Ohne Beschränkung der Allgemeinheit wird daher der stationäre Arbeitspunkt $\mathbf{x}_0 = 0$ mit $r(k+j) = r = 0$ gewählt. Nach diesen Vorüberlegungen muss nun die Stabilität des prädiktiv geregelten Systems in Gleichung (7.19) untersucht werden.

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A} \mathbf{x}(k) + \mathbf{b} u(k) \\ y(k) &= \mathbf{c}^T \mathbf{x}(k) \quad \text{mit } r(k+j) = 0\end{aligned}\quad (7.19)$$

Zum Stabilitätsbeweis wird ein Gütefunktional bis $j = \infty$ angesetzt, das anschließend durch ein endliches Gütefunktional und einer Endzustandsbewertung abgeschätzt wird.

$$\begin{aligned}J &= \sum_{j=1}^{\infty} \tilde{y}^2(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \\ &= \sum_{j=1}^{\infty} \underbrace{\mathbf{c}^T \tilde{\mathbf{x}}(k+j) \cdot \mathbf{c}^T \tilde{\mathbf{x}}(k+j)}_{\tilde{\mathbf{x}}^T \mathbf{c} \mathbf{c}^T \tilde{\mathbf{x}}} + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j)\end{aligned}\quad (7.20)$$

Die Gütefunktion (7.20) wird nun zum Stabilitätsnachweis als Lyapunov-Funktion verwendet. Um den Beweis gemäß Satz 7.1 [17] führen zu können, muss J die folgenden beiden Bedingungen erfüllen:

- J muss für alle $\mathbf{x} \neq 0$ positiv sein.
- $J(k+1) - J(k)$ muss für alle $\mathbf{x} \neq 0$ negativ sein. Dies stellt die zeitdiskrete Lyapunov-Bedingung für asymptotische Stabilität dar.

Wenn $[\mathbf{A}, \mathbf{c}]$ beobachtbar ist, dann können zwar einzelne Terme $\tilde{\mathbf{x}}^T \mathbf{c}$ in Gleichung (7.20) null sein (wenn $\tilde{\mathbf{x}}$ und $\mathbf{c}$ senkrecht zueinander stehen), niemals jedoch die gesamte Summe [15], da sich spätestens nach N Zeitschritten ein $\tilde{\mathbf{x}} \neq 0$ in einem $\tilde{y} \neq 0$ auswirken würde, wenn N die Systemordnung ist. Die Funktion (7.20) kann nur dann null sein, wenn $\tilde{\mathbf{x}}(k+j) = 0$ für $1 \leq j \leq \infty$ gilt und das System sich in der Ruhelage befindet. Dazu muss auch die Stellgrößenänderung $\Delta u(k+j)$ gleich null sein. Für alle anderen Zustände ist J positiv, da nur quadrierte Terme auftreten.

Um die Abnahme von J zwischen zwei beliebigen Zeitschritten zu zeigen, und das Gütefunktional durch eine endliche Summe mit Endzustandsgewichtung auszudrücken, ist die Einführung der zeitdiskreten Lyapunov-Gleichung nötig [1, 58].

Zeitdiskrete Lyapunovgleichung: Betrachtet wird ein lineares zeitdiskretes System der Form

$$\mathbf{x}(k+1) = \mathbf{A} \mathbf{x}(k) \quad (7.21)$$

mit der Anfangsbedingung $\mathbf{x}(k) \in \mathbb{R}^N$. Die konstante Matrix $\mathbf{A}$ besitze nur Eigenwerte innerhalb des Einheitskreises, d.h. das System (7.21) ist asymptotisch stabil. Dem dynamischen System wird das Gütemaß (7.22) mit der positiv definiten Matrix $\mathbf{Q}$ zugeordnet.

$$V(\mathbf{x}(k)) = \sum_{m=k}^{\infty} \mathbf{x}^T(m) \mathbf{Q} \mathbf{x}(m) = \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k) \quad \forall \mathbf{x}(k) \in \mathbb{R}^N \quad (7.22)$$

Für stabile Systeme kann die unendliche Summe durch eine einzelne Gewichtung des Anfangszustands $\mathbf{x}(k)$ mit einer positiv definiten Matrix $\mathbf{P}$ bestimmt werden. Aus der Differenz

$$\begin{aligned} V(\mathbf{x}(k)) - V(\mathbf{x}(k+1)) &= \mathbf{x}^T(k) \mathbf{Q} \mathbf{x}(k) \\ &= \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k) - \mathbf{x}^T(k+1) \mathbf{P} \mathbf{x}(k+1) \\ &= \mathbf{x}^T(k) \mathbf{P} \mathbf{x}(k) - \mathbf{x}^T(k) \mathbf{A}^T \mathbf{P} \mathbf{A} \mathbf{x}(k) \\ &= \mathbf{x}^T(k) (\mathbf{P} - \mathbf{A}^T \mathbf{P} \mathbf{A}) \mathbf{x}(k) \end{aligned} \quad (7.23)$$

folgt die **zeitdiskrete Lyapunovgleichung**

$$\mathbf{Q} = \mathbf{P} - \mathbf{A}^T \mathbf{P} \mathbf{A} \quad (7.24)$$

Ist durch die Matrix $\mathbf{A}$ ein stabiles System beschrieben, und ist $\mathbf{Q}$ eine symmetrische, positiv definite Matrix, so existiert eine eindeutig bestimmte positiv definite, symmetrische

Lösung $\mathbf{P}$ der Lyapunovgleichung (7.24). Diese Aussage ist auch umgekehrt gültig. Ein System ist stabil, wenn es positiv definite Matrizen $\mathbf{P}$ und $\mathbf{Q}$ gibt, die Gleichung (7.24) erfüllen.

Der erste Term des Gütefunktional (7.20) wird nun in zwei Anteile aufgeteilt und mit der positiv semidefiniten Matrix $\mathbf{Q} = \mathbf{c} \mathbf{c}^T$ umformuliert.

$$J = \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \sum_{j=N_2+1}^{\infty} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (7.25)$$

Wegen der Beobachtbarkeit von $[\mathbf{A}, \mathbf{c}]$ kann in Gleichung (7.20) die Summe bis ∞ niemals null werden. Da die symmetrische Matrix $\mathbf{Q} = \mathbf{c} \mathbf{c}^T$ aus den Auskoppelvektoren des Prozesses aufgebaut ist, reicht für die Gültigkeit der zeitdiskreten Lyapunovgleichung eine positiv semidefinite Matrix $\mathbf{Q}$ aus. Das Gütefunktional wird nun mit Hilfe der Lyapunovgleichung vereinfacht.

$$J = \sum_{j=1}^{N_2} \tilde{\mathbf{x}}^T(k+j) \mathbf{Q} \tilde{\mathbf{x}}(k+j) + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) + \tilde{\mathbf{x}}^T(k+N_2+1) \mathbf{P} \tilde{\mathbf{x}}(k+N_2+1) \quad (7.26)$$

Für die prädiktive Regelung wird nun von folgender Vorstellung ausgegangen: Innerhalb des Prädiktionshorizonts N_2 wird der Prozess mit dem aus dem Optimierer der prädiktiven Regelung bestimmten Stellsignal beaufschlagt. Außerhalb wird eine Zustandsrückführung $u(k) = -\mathbf{k}_{reg}^T \cdot \mathbf{x}(k)$ angewendet. Der Rückführvektor $\mathbf{k}_{reg}$ wird so ausgelegt, dass die Zustandsmatrix $\mathbf{A}_{reg} = \mathbf{A} - \mathbf{b} \mathbf{k}_{reg}^T$ des geschlossenen Regelkreises Eigenwerte innerhalb des Einheitskreises besitzt. Diese Wahl ist sicher immer dann möglich, wenn $[\mathbf{A}, \mathbf{b}]$ vollständig steuerbar ist [14]. Es kann demnach eine beliebige Rückführung $\mathbf{k}_{reg}$ ausgewählt werden, die das System stabilisiert.¹⁾ In Gleichung (7.26) ist die positiv definite Matrix $\mathbf{P}$ die Lösung der Lyapunovgleichung

$$\mathbf{Q} = \mathbf{P} - \mathbf{A}_{reg}^T \mathbf{P} \mathbf{A}_{reg} \quad (7.27)$$

Wenn zum Zeitpunkt k die komplette Stellgrößenfolge $\Delta \mathbf{u}(k)$ auf den Prozess geschaltet würde, so ergibt sich für die Differenz zweier aufeinanderfolgender Kostenfunktionen:

$$J(k+1) - J(k) = -\tilde{\mathbf{x}}^T(k+1) \mathbf{c} \mathbf{c}^T \tilde{\mathbf{x}}(k+1) - \lambda \Delta u^2(k) + \lambda \Delta u^2(k+N_u+1) \quad (7.28)$$

Unter der Annahme, die neu hinzugekommene Stellgrößenänderung $\Delta u(k+N_u+1)$ sei null, ergibt sich in Gleichung (7.28) für $\lambda > 0$ eine negative Differenz, also eine Abnahme der Gütefunktion von einem Zeitschritt zum nächsten. Das bedeutet, bereits eine Stelländerung von null zum Zeitpunkt $k+N_u+1$ ergibt eine Verringerung der Gütefunktion. Eine Stellgrößenänderung $\Delta u(k+N_u+1)$ ungleich null kann demnach durch den Optimierer nur dann berechnet werden, wenn durch diesen zusätzlichen Freiheitsgrad eine größere Verringerung des Gütefunktional $J(k+1)$ erreicht wird, als durch den Term $\lambda \Delta u^2(k+N_u+1)$ hinzukommt. Andernfalls führt $\lambda \Delta u^2(k+N_u+1) = 0$ zu geringeren Kosten und würde vom Optimierer auch ausgewählt. Daher lässt sich die Kostenabnahme abschätzen zu

$$J(k+1) - J(k) \leq -\tilde{\mathbf{x}}^T(k+1) \mathbf{c} \mathbf{c}^T \tilde{\mathbf{x}}(k+1) - \lambda \Delta u^2(k) < 0 \quad (7.29)$$

¹⁾ $\mathbf{k}_{reg}$ kann auch zu null gewählt werden, wenn alle Eigenwerte von $\mathbf{A}$ bereits innerhalb des Einheitskreises liegen.

Zu jedem Zeitschritt ist hiermit eine Kostenabnahme gezeigt. Zusammen mit der Tatsache, dass J immer positiv mit der einzigen Nullstelle $\mathbf{x} = 0$ ist, folgt daraus die asymptotische Stabilität der prädiktiven Regelung nach Lyapunov.

Aus diesen Überlegungen folgen folgende Voraussetzungen für die Stabilität der entwickelten prädiktiven Regelung:

- $[\mathbf{A}(k), \mathbf{c}]$ muss für alle Zeitpunkte beobachtbar sein.
- $[\mathbf{A}(k), \mathbf{b}(k)]$ muss für alle Zeitpunkte vollständig steuerbar sein.
- Die Gewichtung λ muss größer null sein.

Für einen beliebigen stationären Arbeitspunkt $\mathbf{x}_0$ wird als Gütefunktional die folgende Gleichung verwendet:

$$J = \sum_{j=1}^{N_2} [\tilde{y}(k+j) - r]^2 + \lambda \cdot \sum_{j=0}^{N_u} \Delta u^2(k+j) \quad (7.30)$$

$$+ (\tilde{\mathbf{x}}(k+N_2+1) - \mathbf{x}_0)^T \mathbf{P} (\tilde{\mathbf{x}}(k+N_2+1) - \mathbf{x}_0)$$

Die Matrix $\mathbf{P}$ zur Endzustandsbewertung ergibt sich aus der Lösung der Lyapunovgleichung (7.27) mit $\mathbf{Q} = \mathbf{c} \mathbf{c}^T$. Die Matrix $\mathbf{A}_{reg}$ errechnet sich aus $\mathbf{A}$ durch eine lineare Zustandsrückführung, so dass $\mathbf{A}_{reg}$ Eigenwerte innerhalb des Einheitskreises besitzt. Der Vektor $\mathbf{x}_0$ kann durch Auswertung von Gleichung (7.16) im stationären Zustand bestimmt werden. $\mathbf{A}_{reg}$ und $\mathbf{x}_0$ müssen für jeden Arbeitspunkt neu berechnet werden. Unter diesen Voraussetzungen nehmen die Kosten zwischen zwei beliebigen Zeitschritten ab und die asymptotische Stabilität nach Lyapunov ist gewährleistet. Nur für den stationären Zustand $\mathbf{x}_0$ sind die Kosten gleich null. Es ist zu beachten, dass der gedachte lineare Zustandsregler mit der Rückführung $\mathbf{k}_{reg}$ nur zu Analysezwecken benötigt wurde und zu keinem Zeitpunkt tatsächlich auf den Prozess wirkt. Durch den zurückweichenden Horizont, wird der obere Prädiktionshorizont niemals erreicht.

Die durchgeführte Argumentation zur Stabilitätsuntersuchung ändert sich nicht, wenn Begrenzungen des Prozesses eingehalten werden müssen. Es muss aber zu jedem Abtastzeitpunkt die Lösbarkeit des Optimierungsproblems gewährleistet sein, d.h. die Nebenbedingungen dürfen nicht zu restriktiv sein. Um garantierte Lösbarkeit zu gewährleisten, ist die Einführung *weicher* Begrenzungen möglich [58], die bei Überschreitung entsprechende Kosten verursachen. Damit können aber bestimmte unerlaubte Betriebsbereiche nicht mehr ausgeschlossen werden.

In der Praxis kann auf die Endzustandsbeschränkung in Gleichung (7.30) meist verzichtet werden. Es genügt, den oberen Prädiktionshorizont N_2 so groß zu wählen, dass die wesentliche Prozessantwort dadurch berücksichtigt wird. Einen Richtwert stellt die größte Zeitkonstante des Prozesses dar. Dies hat zur Folge, dass die Kosten innerhalb des Prädiktionshorizonts diejenigen außerhalb bei weitem überwiegen und dadurch ebenfalls eine Abnahme zwischen zwei Zeitschritten gewährleistet ist. Mit dieser Methode wurden auch alle praktischen Ergebnisse in den Kap. 8 und 9.2 erzielt.

8 Experimentelle Validierung der prädiktiven Regelung

Zur Validierung der theoretischen Ergebnisse zur prädiktiven Regelung erfolgt in diesem Kapitel eine Implementierung auf dem in Anhang A beschriebenen Versuchsstand. Es handelt sich um ein Zweimassensystem (ZMS) mit isolierter Nichtlinearität und Reibung. Die Reibkennlinie wurde zwar in Anhang A.3 experimentell bestimmt, sie wird aber nicht im Prozessmodell berücksichtigt, sondern wirkt als Störung. Die isolierte Nichtlinearität wirkt auf die Lastmaschine und lautet:

$$\mathcal{N}(\Omega_2) = 6.4 [Nm] \cdot \arctan(10 [s/rad] \cdot \Omega_2) \quad (8.1)$$

Die Zustandsdarstellung der Regelstrecke ist in Gleichung (A.8) gegeben. Den zugehörigen Signalflussplan zeigt Bild A.5.

Im Folgenden wird die prädiktive Regelung am Zweimassensystem implementiert. Es werden zunächst Ergebnisse ohne Integralanteil dargestellt, die die Notwendigkeit der Erweiterung um einen LQI-Regler herausstellen. Schließlich werden auch Messungen mit verschiedenen Begrenzungen durchgeführt und die Auswirkungen der Einstellparameter der Regelung untersucht. Ein Vergleich mit einer konventionellen PI-Regelung zeigt die Verbesserungen durch den prädiktiven Regler.

8.1 Prädiktive Regelung ohne Begrenzungen

Zunächst werden alle prädiktiven Regelverfahren ohne die Berücksichtigung von Beschränkungen implementiert. Die prädiktive Regelung mit Beschränkungen ist in Kap. 8.2 ausgeführt.

8.1.1 MPC ohne Integralanteil

In den folgenden Untersuchungen wird eine prädiktive Drehzahlregelung des Zweimassensystems ohne Integralanteil (LQI-Regler) durchgeführt. Das ZMS wird mit einem Sprung, einer Rampe und einem Sinussignal als Führungsgröße angeregt. Der Sprung wird nicht direkt als Solltrajektorie verwendet, da das Zweimassensystem nie exakt folgen kann. Der Sprung wird durch ein PT₂-Glied mit zwei identischen Zeitkonstanten von 25 ms geglättet. Für alle drei Anregungsarten werden folgende Einstellungen vorgenommen:

$$\begin{array}{lll} T_{ab} = 3 \text{ ms} & z = 1.5 \text{ Nm (Störgröße)} & \\ N_1 = 3 & N_2 = 15 & N_u = 3 \end{array}$$

Für die Sprunganregung wird der Gewichtungsfaktor $\lambda = 0.05$ gesetzt. Für alle Anregungsarten werden kleine Amplituden verwendet, da in diesem Bereich der nichtlineare Charak-

ter der isolierten Nichtlinearität in Gleichung (8.1) besonders ausgeprägt ist. Für große Amplituden geht die Nichtlinearität in Sättigung und wirkt wie eine konstante Störgröße. Bild 8.1 zeigt die Sprungantworten des prädiktiv geregelten Zweimassensystems in einer Simulation (oben) und als Messung (unten). In den beiden linken Bildern sind die Stell-

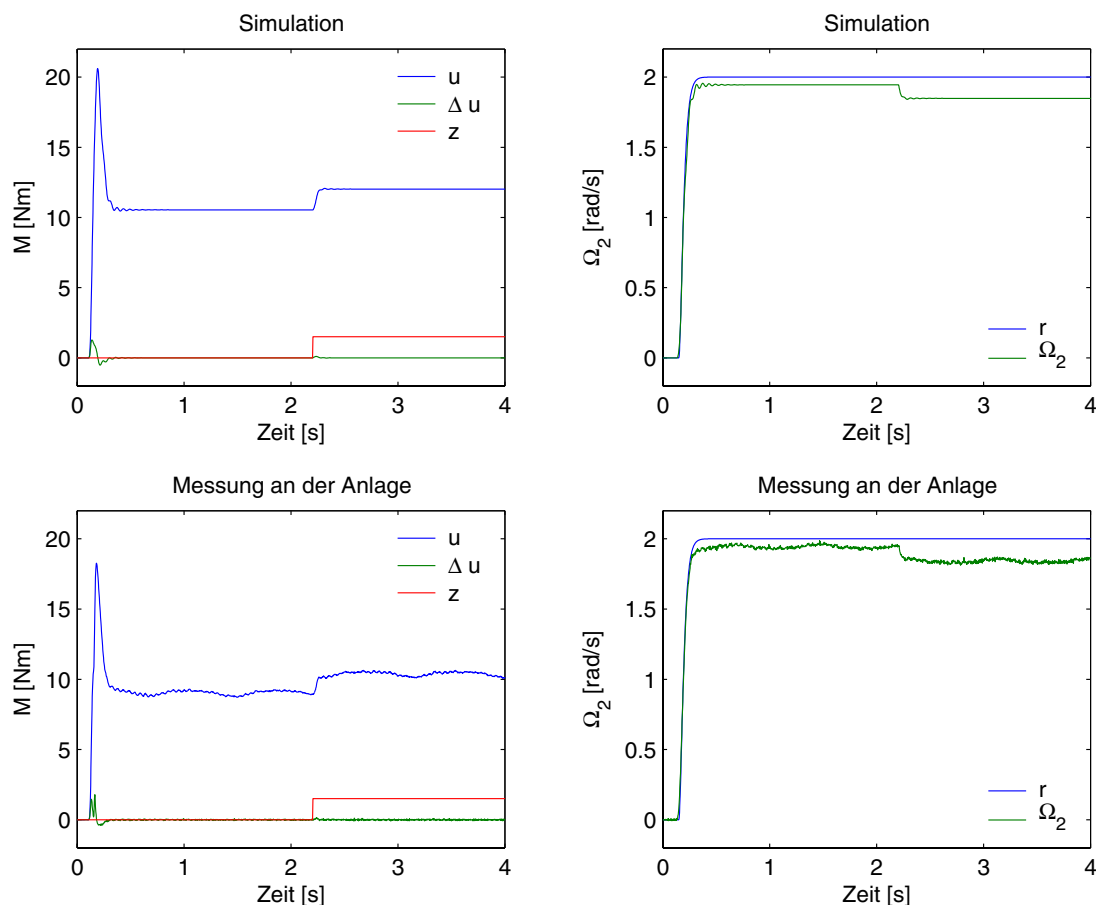


Bild 8.1: Sprungantwort des prädiktiv geregelten ZMS

größe u , die Stellgrößenänderung Δu und die Störgröße z dargestellt. Die beiden rechten Bilder zeigen die Solltrajektorie r und den Streckenausgang Ω_2 . Es fällt auf, dass selbst ohne Störgröße eine kleine stationäre Abweichung bestehen bleibt. Das liegt daran, dass der prädiktive Regler wie ein proportionaler Zustandsregler wirkt und unbekannte Störungen nicht unterdrücken kann. Als Störung ist jedoch immer ein Reibmoment vorhanden. Bei zugeschalteter Störgröße vergrößert sich diese stationäre Abweichung. Zwischen Simulation und Messung ist sehr gute Übereinstimmung festzustellen.

Für die rampenförmige Führungsgröße wird der Gewichtungsfaktor $\lambda = 0.05$ gewählt. Bild 8.2 zeigt die Stellgrößen (links) und das Führungsverhalten (rechts) bei Rampenanregung. In den beiden linken Teilbildern sind die Stell- und Störgrößen dargestellt, rechts ist das Folgeverhalten für Simulation und Messung gezeigt. Es ist wieder gutes Folgeverhalten erkennbar, jedoch mit zunehmender Abweichung nach Auftreten der Störgröße. Dieser Schleppfehler entsteht durch die Störgröße z und das natürlich vorhandene Reibmoment der Lagerung.

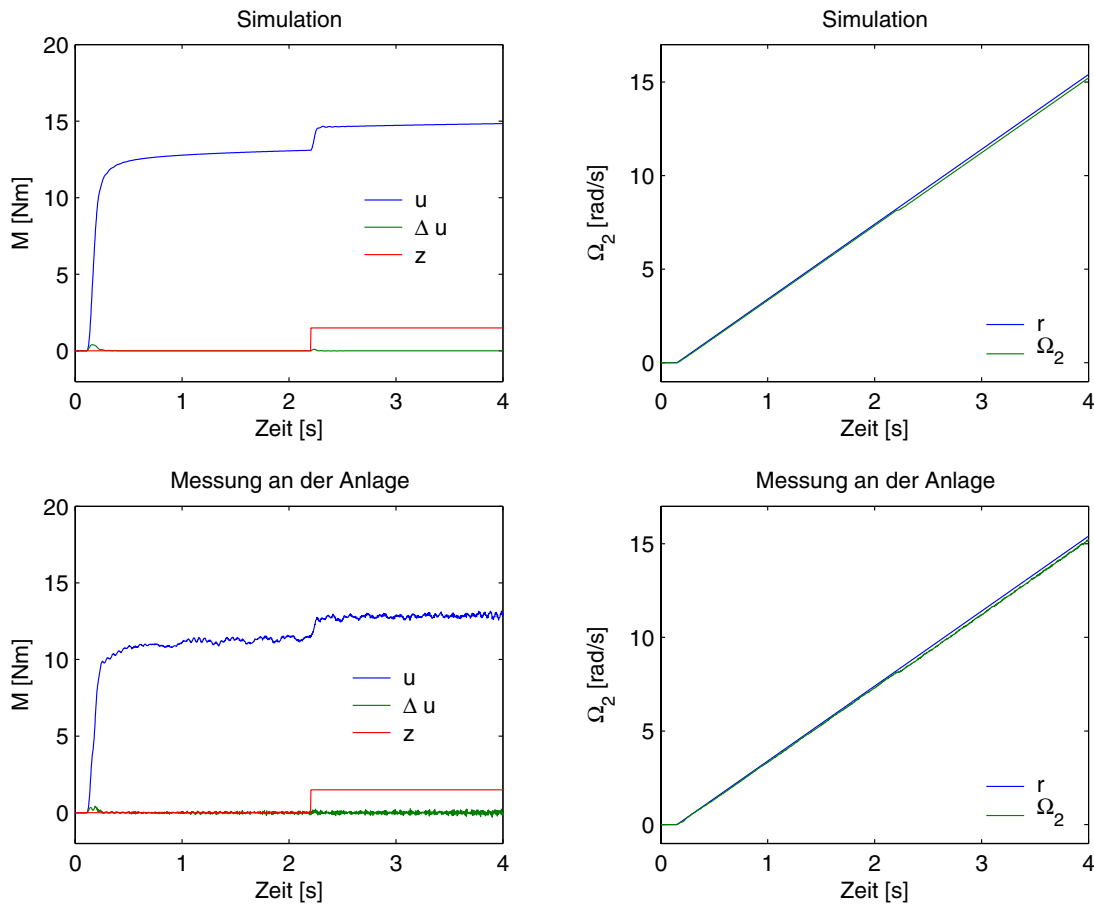


Bild 8.2: Folgeverhalten des ZMS bei rampenförmiger Führungsgröße

Bei sinusförmiger Führungsgröße wird wiederum $\lambda = 0.05$ gesetzt. Bild 8.3 zeigt die Stell- und Störgrößen (links) und das Folgeverhalten bei Sinusanregung (rechts) für Simulation und Messung. Auch hier zeigt sich gutes Folgeverhalten, der negative Einfluss der nicht ausgeglichenen Störgröße ist beim Sinusverlauf nur schwach sichtbar. Vergleicht man die Simulationsergebnisse mit den Messergebnissen der Anlage, so kann sehr gute Übereinstimmung festgestellt werden. Die geringen Unterschiede zwischen den Ergebnissen aus der Simulation und den gemessenen Daten lassen sich auf Messungenauigkeiten und Rauscheinflüsse sowie nicht ideales Umrichterverhalten (im Umrichter integrierte Momentenregelung!) zurückführen. Dies deutet auf eine ausreichende Genauigkeit bei der Modellierung bzw. Identifikation hin. Deshalb wird im Folgenden auf die Gegenüberstellung von Simulations- und Messergebnissen verzichtet und es wird nur noch das jeweilige Messergebnis des Prüfstands vorgestellt.

Da bei einer Regelung stationäre Genauigkeit auch bei unbekannten Störgrößen ein primäres Regelziel ist, wird nun der prädiktive Regler um einen LQI-Regler erweitert.

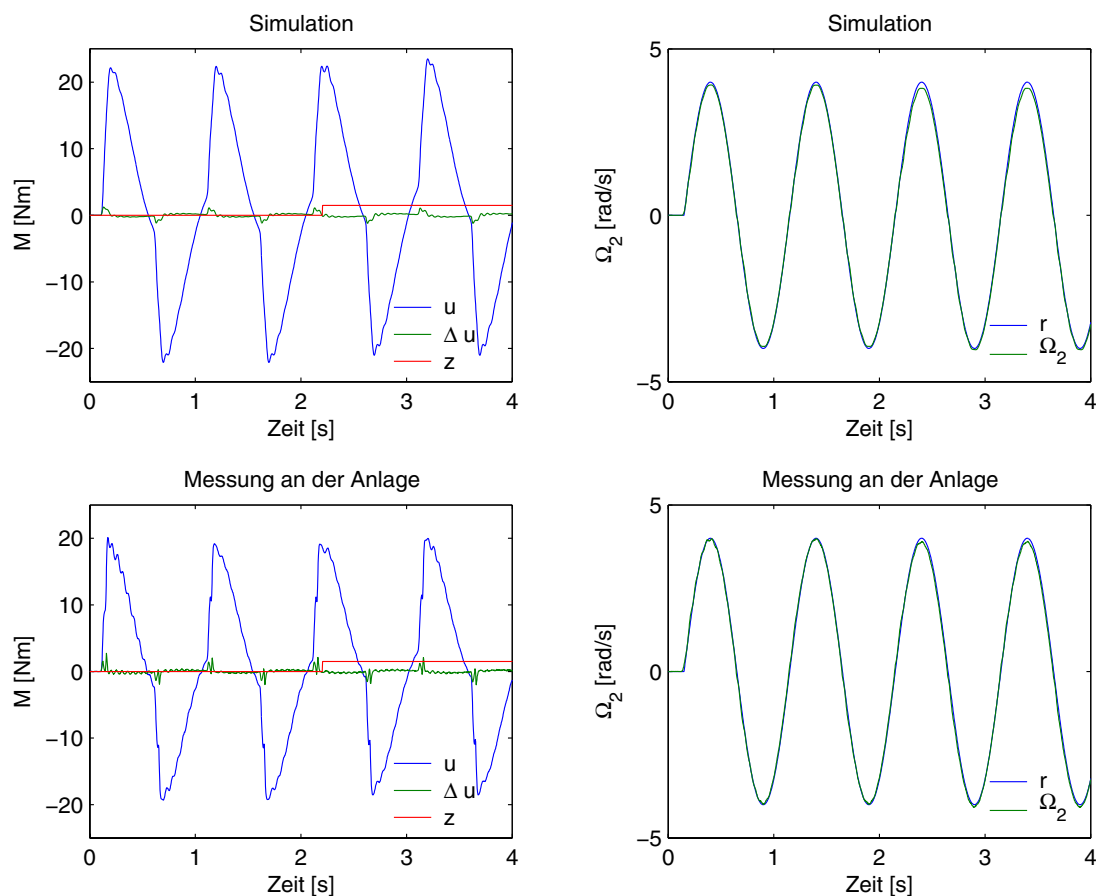


Bild 8.3: Folgeverhalten des ZMS bei sinusförmiger Führungsgröße

8.1.2 MPC mit Integralanteil

Die Erweiterung um einen Integralanteil erfolgt gemäß Kap. 6.4. Es werden die gleichen Formen der Führungsgrößen wie in Kap. 8.1.1 verwendet. Die Sprungantwort des prädiktiv geregelten Zweimassensystems ist in Bild 8.4 dargestellt. Es wurden die Gewichtungsfaktoren $\lambda = 0.05$ und $\alpha = 0.01$ gewählt. Die Störgröße z besitzt eine Amplitude von 3 Nm . Es kann sehr gutes Führungsverhalten festgestellt werden. Auch die Störgröße wird nach einer kurzen Ausregelzeit ohne stationären Fehler ausgeregelt. Dazu steigt die Stellgröße u entsprechend an.

Bild 8.5 zeigt den Stell- und Ausgangsgrößenverlauf bei rampenförmiger Führungsgröße. Für die Gewichtungsfaktoren wurde $\lambda = 0.05$ und $\alpha = 0.075$ gesetzt. Sowohl vor als auch nach Auftreten der Störung ist nahezu exakte Übereinstimmung zwischen Soll- und Istwert der Regelgröße festzustellen. Lediglich zur Ausregelung der Störung entsteht eine kurzzeitige Regeldifferenz. Die Stellgröße u ist nahezu konstant, was einer konstanten Beschleunigung des ZMS entspricht. Dies liegt daran, dass die Nichtlinearität ab einer Drehzahl von etwa 3 rad/s in Sättigung ist und wie eine konstante Störung wirkt. Aus diesem Grund ist bei ansteigender Drehzahl keine ansteigende Stellgröße nötig.

Bei sinusförmiger Referenztrajektorie werden die Faktoren $\lambda = 0.05$ und $\alpha = 0.01$ gewählt.

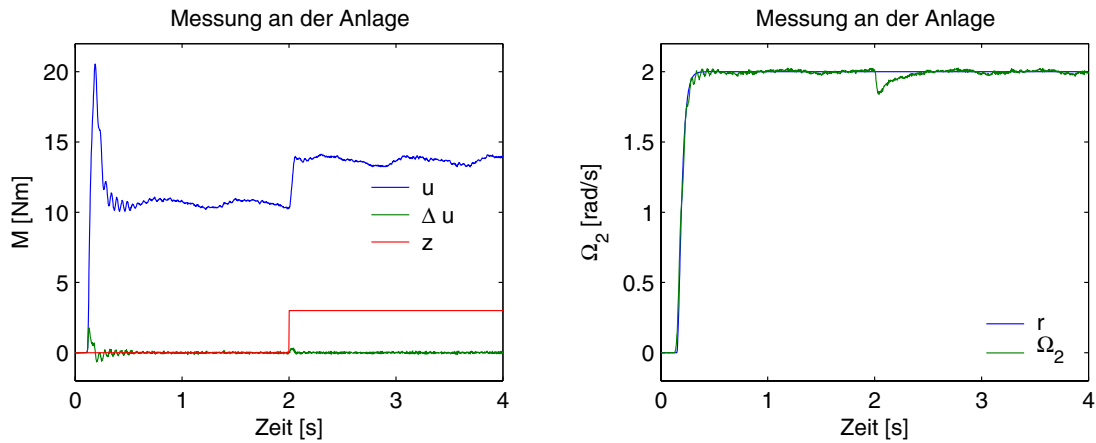


Bild 8.4: Sprungantwort des prädiktiv geregelten ZMS mit Integralanteil

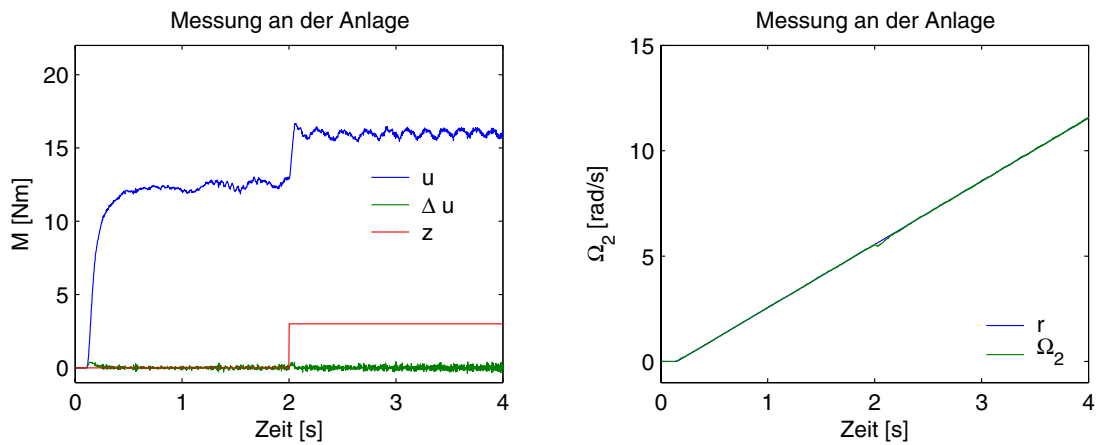


Bild 8.5: Folgeverhalten des ZMS bei rampenförmiger Führungsgröße und I-Anteil

Der Stell- und Regelgrößenverlauf ist in Bild 8.6 dargestellt. Durch Verwendung des prädiktiven

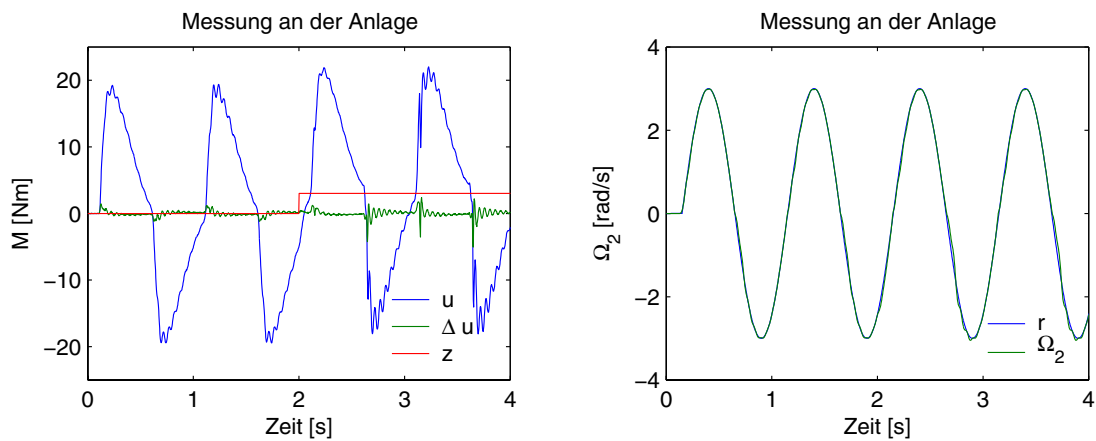


Bild 8.6: Folgeverhalten des ZMS bei sinusförmiger Führungsgröße und I-Anteil

Reglers mit Integralanteil sind in diesem Fall weder vor noch nach Auftreten der Störgröße Abweichungen vom Sollverlauf zu erkennen. Es ist auch kein Ausregelvorgang nach Auftreten der Störung sichtbar. Es wird demnach sowohl die Störgröße, als auch die natürlich vorhandene Lagerreibung ausgeregelt.

Bei sprungförmiger Anregung kann die Regelung mit Integralanteil beim Übergang zu Schwingungen neigen. Die Wahl des Gewichtungsfaktors α sollte so gewählt werden, dass auf der einen Seite ein möglichst geringes Schwingen bei Übergängen verursacht wird und auf der anderen Seite der stationäre Regelfehler so schnell wie möglich ausgeregelt wird. Zwischen der Sollwerttrajektorie und dem Ausgangssignal ergeben sich folgende mittlere relative Regelfehler: Sprunganregung 1.4 %, Rampenanregung 0.479 % und Sinusanregung 1.13 %. Diese geringen Abweichungen unterstreichen das sehr gute Führungsverhalten der prädiktiven Regelung.

8.2 Prädiktive Regelung mit Begrenzungen

Beim verwendeten Prüfstand ist eine natürliche Begrenzung der Stellgröße (Motormoment) u auf $\pm 22 \text{ Nm}$ gegeben. Diese muss eingehalten werden, damit eine thermische Überlastung der Motoren vermieden wird. Stellgrößenänderungsbeschränkungen Δu werden bei Schiebern, Ventilen etc., die bauartbedingt nur mit einer maximalen Beschleunigung betrieben werden dürfen, eingesetzt. Am ZMS ist die Begrenzung der Stellgrößenänderung Δu aufgrund der guten Dynamik der Momentenregelung der Umrichter nicht nötig. Um ein gutes Regelergebnis bei Begrenzungen zu erhalten, bzw. um Sicherheitsabstände von gefährlichen Betriebspunkten einzuhalten, werden in Kap. 8.2.2 und 8.2.3 Begrenzungen eines inneren Zustandssignals und der Regelgröße berücksichtigt.

8.2.1 Stellgrößenbegrenzungen

Die Einhaltung der Stellgrößenbegrenzungen soll anhand der Drehzahlregelung des ZMS gezeigt werden. Die Stellgröße ist auf $\pm 22 \text{ Nm}$ begrenzt. Es wird ein prädiktiver Regler mit Integralanteil zur Sicherstellung stationärer Genauigkeit eingesetzt. Der prädiktive Regler wird mit den folgenden Einstellungen betrieben:

$$\begin{array}{lll} T_{ab} = 3 \text{ ms} & z = 3 \text{ Nm} & \\ N_1 = 3 & N_2 = 15 & N_u = 3 \end{array}$$

In den Bildern 8.7 bis 8.9 ist das Regelergebnis für sprung-, rampen- und sinusförmige Referenztrajektorien dargestellt. Bei allen drei Anregungsarten ist deutlich zu erkennen, dass die Stellgrößenbeschränkung $u = 22 \text{ Nm}$ eingehalten werden. Nach $t = 2 \text{ s}$ wird ein Störsignal $z = 3 \text{ Nm}$ auf den Motor 2 des ZMS aufgeschaltet. Befindet sich der Antriebsmotor in der Stellgrößenbeschränkung kann das Ausgangssignal Ω_2 der Sollwerttrajektorie r nur verzögert folgen und es entsteht ein stationärer Fehler. Bei sprungförmiger Anregung wird die Stellbegrenzung nur beim Anfahren aktiv, da zu diesem Zeitpunkt der Momentenbedarf am größten ist. Auffällig in Bild 8.7 ist, dass die Stell- und Regelgröße vor der ansteigenden Referenztrajektorie ins Negative wechselt. Eine anschauliche

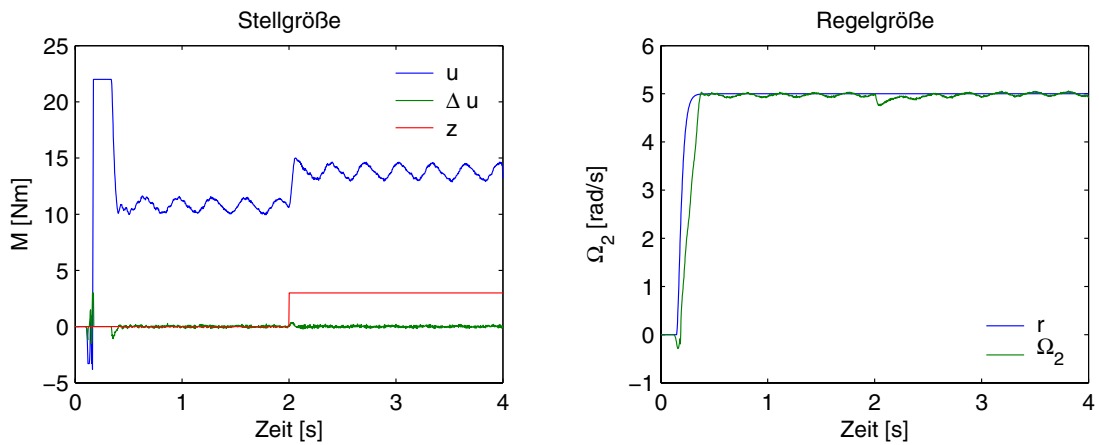


Bild 8.7: Regelergebnis am ZMS bei Stellgrößenbeschränkung (Sprung)

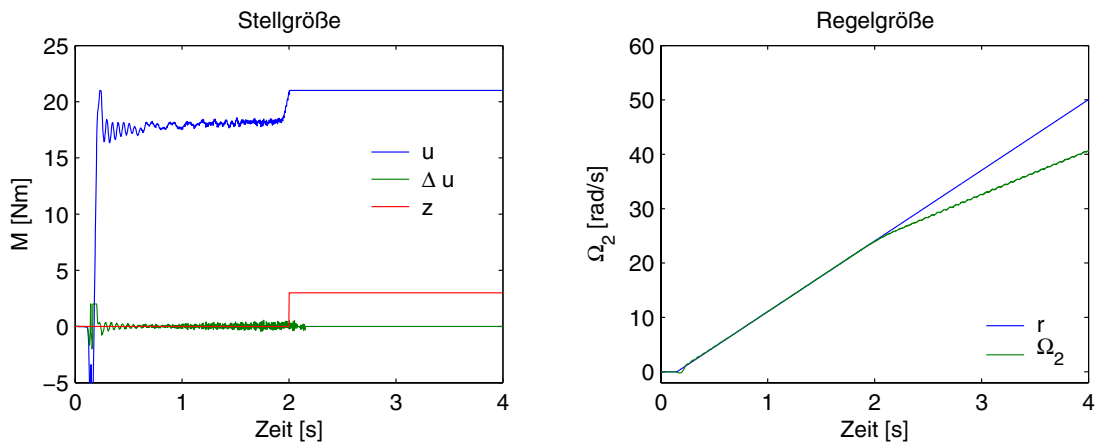


Bild 8.8: Regelergebnis am ZMS bei Stellgrößenbeschränkung (Rampe)

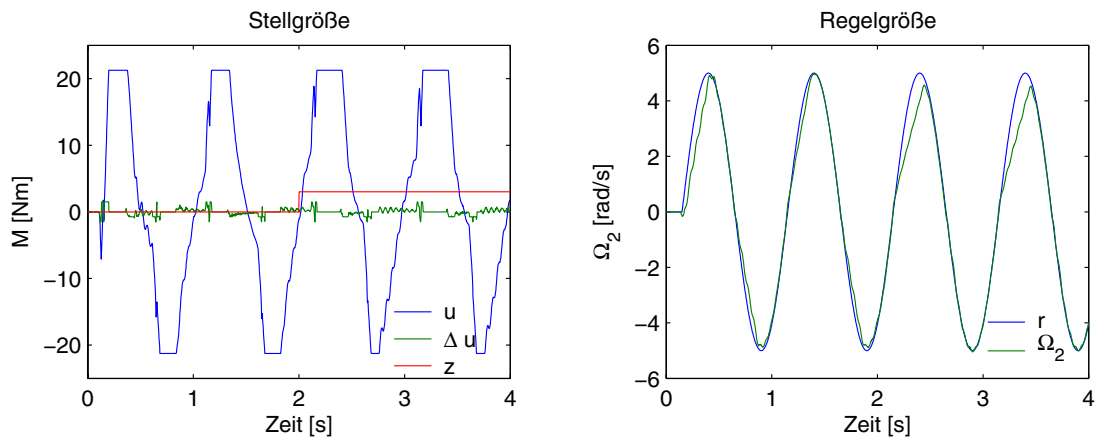


Bild 8.9: Regelergebnis am ZMS bei Stellgrößenbeschränkung (Sinus)

Erklärung erhält man, wenn man bedenkt, dass der prädiktive Regler versucht, zukünftige Regeldifferenzen, unter Berücksichtigung von Beschränkungen, zu vermeiden. Bereits vor dem Sollwertübergang stellt der Regler fest, dass der Übergang nicht ohne Stellbegrenzung möglich ist. Um dennoch einen möglichst steilen Übergang zu erzielen, wird zunächst die elastische Verbindung in die Gegenrichtung aufgezo-gen. Dadurch wird Energie im System gespeichert. Diese wird dann beim Übergang zusätzlich zur Stellenergie für einen möglichst steilen Anstieg frei. Der entstehende Regelfehler vor dem Übergang ist kleiner als der Fehler, der während eines Übergangs ohne vorheriges Unterschwingen entstehen würde. Dies stellt der Optimierer im prädiktiven Regler sicher. Dieses Verhalten ist typisch für das prädiktiv geregelte ZMS (siehe auch Bild 8.8). Bei rampenförmiger Referenztrajektorie wird die Begrenzung erst nach Zuschalten der Störgröße aktiv, da die Steigung der Rampe zum Antriebsmoment direkt proportional und kleiner als die Stellbegrenzung ist. Bei aktivem Störsignal ist der Regelkreis geöffnet und ein Folgen des Referenzsignals ist nicht mehr möglich. Die Stellgröße u erreicht dauerhaft ihren Grenzwert, da die geforderte Beschleunigung vom Stellglied bei aktiver Störgröße nicht mehr aufgebracht werden kann. Die Ursache für den Regelfehler ist daher nicht bei den Reglereinstellungen, sondern bei der Dimensionierung des Stellglieds zu suchen. Bei sinusförmiger Anregung ist pro Periode zweimal die Begrenzung aktiv, dementsprechend verschlechtert sich das Folgeverhalten in den entsprechenden Bereichen.

8.2.2 Zustandsbegrenzungen

Mit Hilfe von Zustandsgrößenbeschränkungen können interne Prozesssignale so eingegrenzt werden, dass sie zulässige Betriebsbereiche nicht verlassen. Beim Zweimassensystem sind die beiden Motoren mit einem Torsionsstab gekoppelt. Je nach Auslegung der Maschinen und der Welle besteht die Gefahr, dass das Torsionsmoment M_T zu groß wird und dies zum Bruch der Welle führt. Mit der Begrenzung des Zustandsvektors kann direkt die Verdrehung $\Delta\phi$ des Torsionsstabes begrenzt werden, um die Welle vor einer zu starken Beanspruchung bzw. vor ihrem Bruch zu schützen. Es wird eine Grenze $\Delta\phi = \pm 0.015 \text{ rad}$ für die Torsion festgelegt. In Bild 8.10 wird der Zustand $x_2 = \Delta\phi$ von $\Delta\phi = 0.02 \text{ rad}$ ($\approx 1.14^\circ$) auf $\Delta\phi = 0.015 \text{ rad}$ ($\approx 0.86^\circ$) reduziert, d.h. die Nebenbedingung wird eingehalten. Bild 8.10 oben zeigt Stell-, Regelgröße und Torsionswinkel ohne die Berücksichtigung von Begrenzungen. Bild 8.10 unten zeigt das Verhalten bei aktiver Begrenzung für $\Delta\phi$. Das Maximum der Stellgröße reduziert sich aufgrund der Zustandsbegrenzung kaum, lediglich ihre zeitliche Verteilung verändert sich. Dadurch kann das System nur verzögert der Sollwerttrajektorie r folgen. Die Einhaltung der Zustandsbeschränkung für $\Delta\phi$ wird garantiert, aber die Verläufe der Größen u , Δu und Ω_2 erfahren während des Übergangs deutliche Veränderungen. Dabei ist besonders der Drehzahlverlauf Ω_2 auffällig, der zuerst in die negative Richtung ausschlägt, bevor er der Trajektorie r folgt. Der Torsionsstab wird auch hier in die negative Richtung aufgezo-gen (Energie wird gespeichert), um dann einen schnelleren, aber leicht verspäteten Übergang auf den Sollwert mit Hilfe der gespeicherten Energie zu ermöglichen. Hier wird besonders deutlich, dass die Einführung von Beschränkungen zu nicht erwartetem Systemverhalten führen kann.

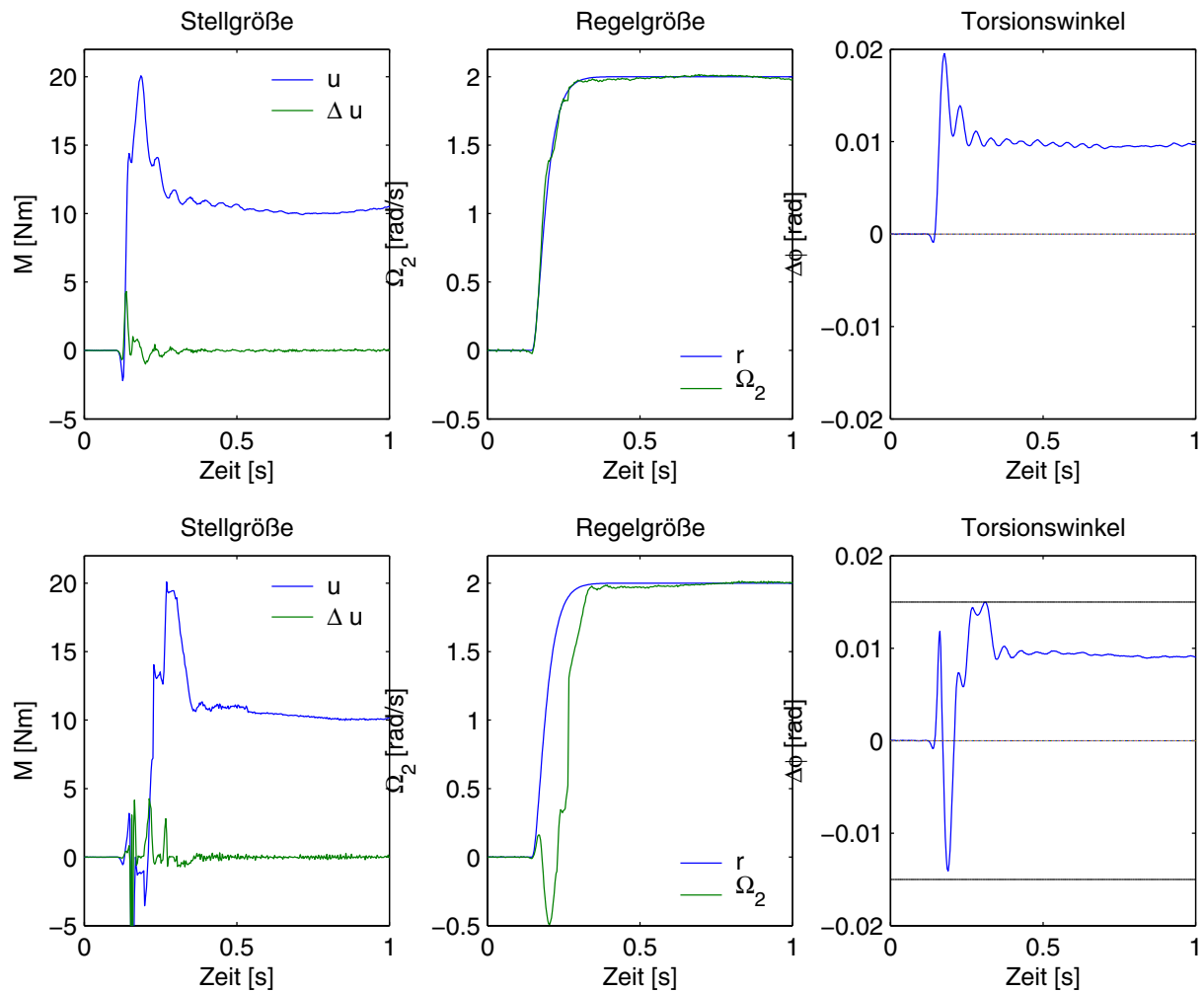


Bild 8.10: *Regelergebnis am ZMS ohne (oben) und mit (unten) Begrenzung des Torsionswinkels $\Delta\phi$*

8.2.3 Regelgrößenbegrenzungen

Beim Übergang der Drehzahl des Zweimassensystems von einem auf einen anderen stationären Wert neigt die prädiktive Regelung bei manchen Einstellungen von Begrenzungen und Reglerparametern zum Unterschwingen. Dieser Effekt ist besonders störend, wenn sich die Maschine an einem mechanischen Anschlag befindet. Das System stößt zuerst gegen die (mechanische) Begrenzung, bevor der Übergang erfolgt. Dies kann zur Zerstörung bzw. Beschädigung der Anlage führen. Um diesen Nebeneffekt zu vermeiden, wird eine Ausgangsbegrenzung $y \geq 0$ eingesetzt. Aufgrund der Ungleichungsnebenbedingung (UNB) schwingt das System nicht mehr unter die Drehzahl null. Bei der Wahl der Beschränkungen muss darauf geachtet werden, dass das Optimierungsproblem lösbar bleibt. Insbesondere dürfen keine sich gegenseitig ausschließenden Nebenbedingungen gefordert werden.

In Bild 8.11 wird die Einhaltung der Ausgangsbegrenkung am ZMS gezeigt. In den oberen Teilbildern ist die Sprungantwort des Zweimassensystems dargestellt, wobei die Ausgangsbegrenkung $y \geq 0 \text{ rad/s}$ noch nicht wirksam ist. Bei der Ausgangsgröße Ω_2 ist

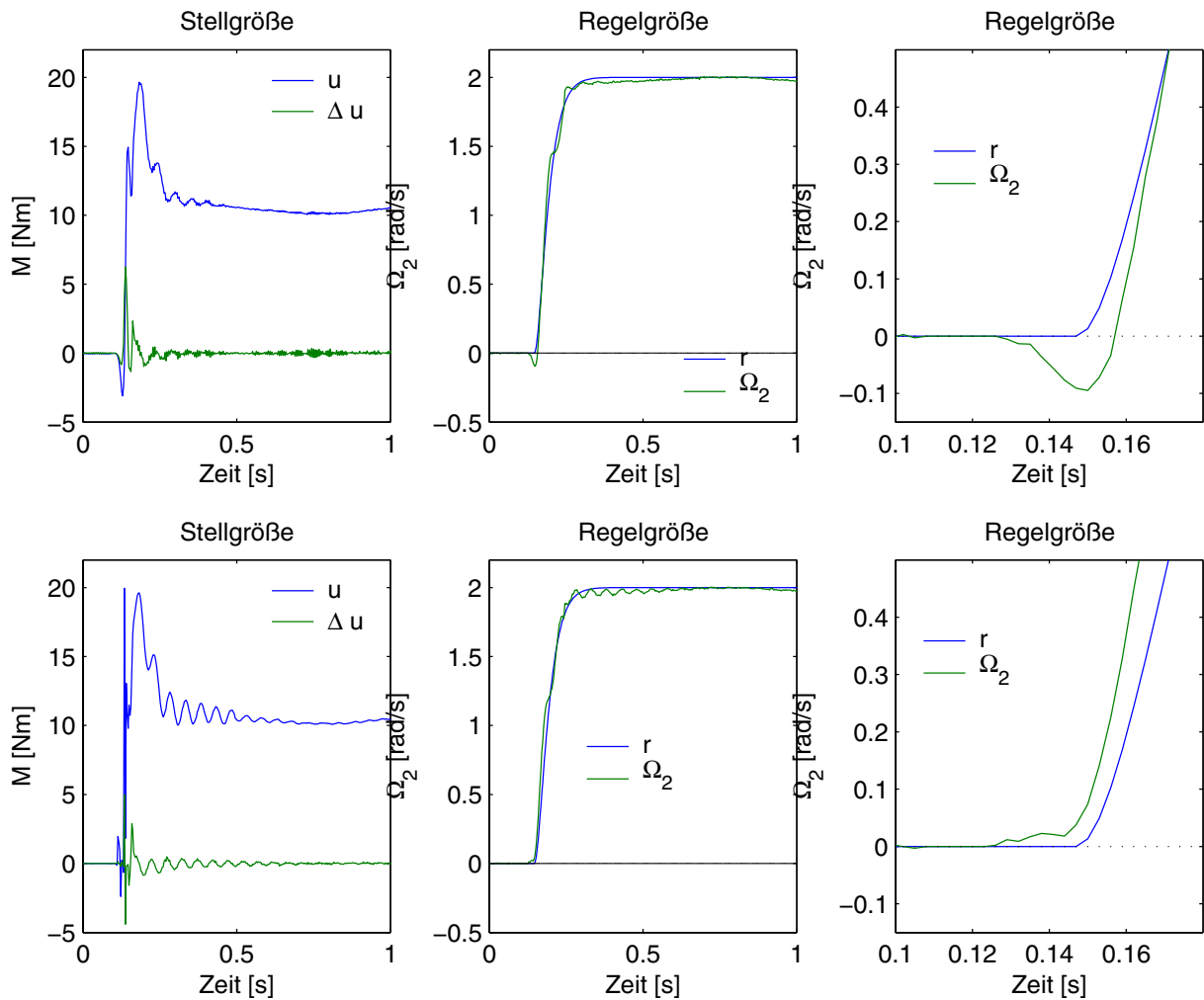


Bild 8.11: Regelergebnis am ZMS ohne (oben) und mit (unten) Begrenzung der Ausgangsdrehzahl auf $\Omega_2 \geq 0$

das Unterschwingen vor der Zustandsänderung deutlich sichtbar. In den unteren Bildern ist die Ausgangsbeschränkung aktiv. Des Systems hält die Ungleichungsnebenbedingung ein, und die Drehzahl Ω_2 wird nicht mehr negativ. Dies ist besonders in der Vergrößerung der beiden rechten Teilbilder erkennbar. Die Nebenbedingung bewirkt auch einen veränderten Verlauf der Stellgröße u , der Stellgrößenänderung Δu und der Regelgröße Ω_2 . Statt des Unterschwingens der Regelgröße ist nun ein verfrühter Anstieg erkennbar. Dadurch wird die Steilheit des Übergangs abgeflacht.

Alle gezeigten Begrenzung für Stellgrößen, Zustandsgrößen und Regelgrößen können gleichzeitig aktiv sein und in des Optimierungsproblem integriert werden. Dazu muss bei einer hohen Zahl von Begrenzungen beachtet werden, dass die Lösbarkeit der Optimierungsaufgabe erhalten bleibt. Dazu kann keine einfache Regel angegeben werden, dies stellt vielmehr noch ein Gebiet intensiver Forschung dar [82]. Außerdem steigt mit jeder Nebenbedingung der Online-Rechenaufwand. Es gilt daher, den Nutzen der Einhaltung einer Beschränkung gegen den zusätzlichen Entwurfs- und Rechenaufwand abzuwägen.

8.3 Untersuchung der Einstellparameter

Um die Auswirkungen der Einstellparameter N_1 , N_2 , N_u , λ und α besser einschätzen zu können, wird deren Einfluss auf den prädiktiven Regler durch Parametervariationen untersucht. Dies geschieht am ZMS mit isolierter Nichtlinearität und Führungsintegrator, aber ohne Einflussnahme der Beschränkungen. Da die Veränderung einzelner Parameter teilweise nur sehr schwache Auswirkungen zeigen, die bisherigen Ergebnisse am Versuchsaufbau mit der Simulation sehr gut übereinstimmen und zusätzliche Störeinflüsse wie Messrauschen nicht in das Ergebnis mit einfließen sollen, werden die Auswirkungen der Parameteränderungen simulativ untersucht. In den nachfolgenden Simulationen wird jeweils eine Einstellgröße in die positive und negative Richtung variiert und die anderen Parameter werden konstant gehalten.

Bei der Simulationsdurchführung wird das System mit einem geglätteten Sprung als Referenztrajektorie beaufschlagt. In den Bildern 8.12 bis 8.16 wird in den linken Teilbildern die Stellgröße u und in den rechten Teilbildern der Ausgangsgrößenverlauf Ω_2 mit der Referenztrajektorie r abgebildet. Ausgehend von den Einstellungen $N_1 = 3$, $N_2 = 15$, $N_u = 3$, $\lambda = 0.0025$ und $\alpha = 0.005$ erfolgen die Parametervariationen.

Einfluss von N_1 :

Die Variation des unteren Prädiktionshorizontes N_1 bewirkt nur eine geringe Veränderung im Regelkreis, siehe Bild 8.12. Für totzeitbehaftete Systeme sollte der untere Prädiktions-

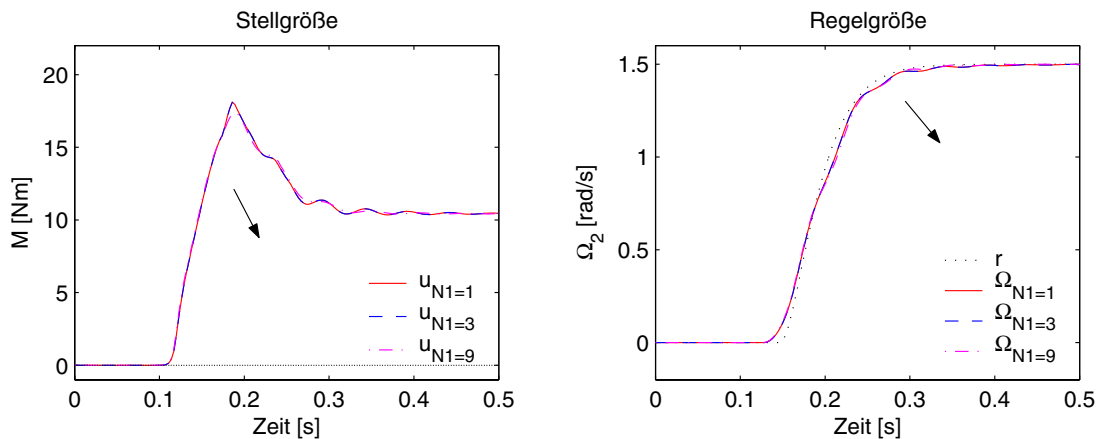


Bild 8.12: Variation des unteren Horizontes N_1

horizont ungefähr der Verzögerungszeit entsprechen, da eine gewisse Zeitspanne gewartet werden muss, bis der Stelleingriff sich auf die Prozessänderung auswirkt. In der Kostenfunktion wird der Zeitraum zwischen unterem und oberem Prädiktionshorizont bewertet, wird N_1 zu groß gewählt werden relevante Zeitschritte nicht mehr gewichtet, so dass eine Verschlechterung des Regelungsergebnis eintritt. Bei trägen Prozessen ist es vorteilhaft den Horizont $N_1 > 1$ zu setzen, da dem Regler N_1 Abtastschritte Zeit bleiben, um den Prozessausgang an die Sollwerttrajektorie anzugleichen. Damit wird vermieden, dass der Regler versucht, einem unrealistisch schnellen Sollwertverlauf zu folgen. Das System arbeitet dann ruhiger und das Ausgangssignal verläuft glatter. Das System wird mit einem

steigenden N_1 träger, die Stellgröße u vermindert ihre Amplituden und der Ausgangsverlauf Ω_2 flacht am Übergang ab. Aufgrund der schwachen Änderungen ist die Richtung von steigendem N_1 mit einem Pfeil gekennzeichnet.

Einfluss von N_2 :

Der obere Prädiktionshorizont ist ein entscheidender Einstellparameter der prädiktiven Regelung. Je größer der obere Horizont N_2 gewählt wird, desto weniger neigt das Ausgangssignal Ω_2 zum Überschwingen. Zu lange Vorhersagezeiträume führen aber zu keiner weiteren Verbesserung der Regelungseigenschaften und steigern den Rechenaufwand erheblich. Der obere Horizont bestimmt auch, wie lange vor der Sollwertänderung bereits mit Stellaktionen begonnen wird. Da bei der prädiktiven Regelung ein unerwünschtes Unterschwingen auftreten kann, sollte darauf geachtet werden, dass die Stellaktionen nicht zu früh beginnen. Ein klein gewählter Wert für N_2 erhöht die Dynamik, verringert aber gleichzeitig die Dämpfung des Gesamtsystems. In Bild 8.13 ist bereits die Welligkeit beim Ausgangssignal Ω_2 und bei der Stellgröße u deutlich erkennbar. Bei einer weiteren Absen-

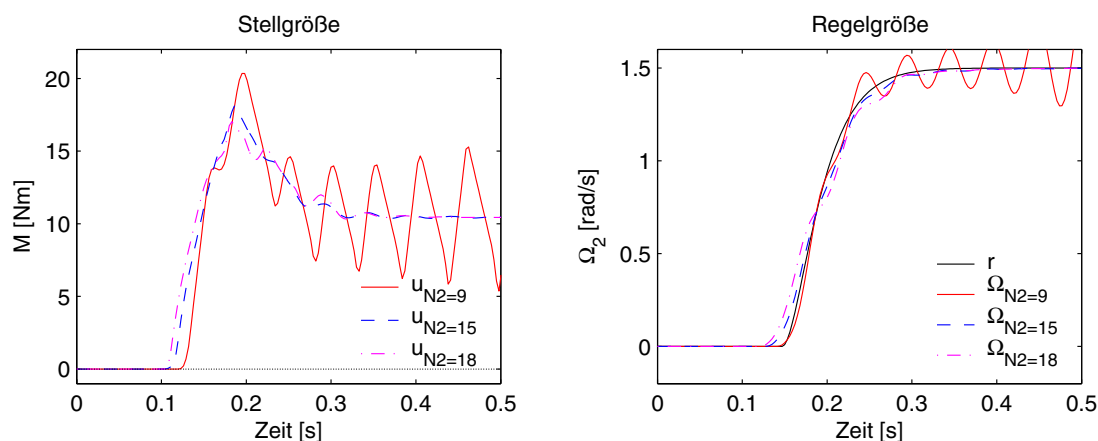


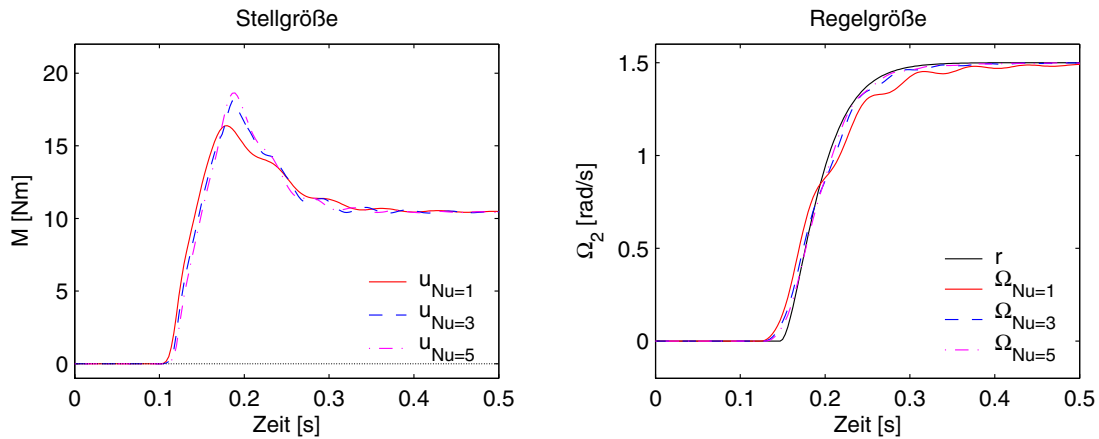
Bild 8.13: Variation des oberen Horizontes N_2

kung des oberen Prädiktionshorizontes wird die Prozessantwort nicht mehr lange genug im Gütefunktional bewertet, da wesentliche Abschnitte des zukünftigen Prozessverhaltens vernachlässigt werden. Dies führt zur Instabilität des Systems. In [64] wird die Empfehlung gegeben, dass der Wert von $N_2 \cdot T_{ab}$ in der Nähe der größten Systemzeitkonstanten liegen sollte.

Einfluss von N_u :

Die Begrenzung des Stellgrößenvektors $\Delta \mathbf{u}$ auf N_u Elemente führt zur Reduzierung der Freiheitsgrade des Optimierungsproblems, woraus eine Verminderung des Rechenaufwandes resultiert. Bei zu geringem N_u sind zu wenige Freiheitsgrade im Optimierungsproblem vorhanden und es kann nur eine suboptimale Lösung gefunden werden. Dadurch kommt es zu Abweichungen zwischen der Ausgangsgröße Ω_2 und der Sollwerttrajektorie r .

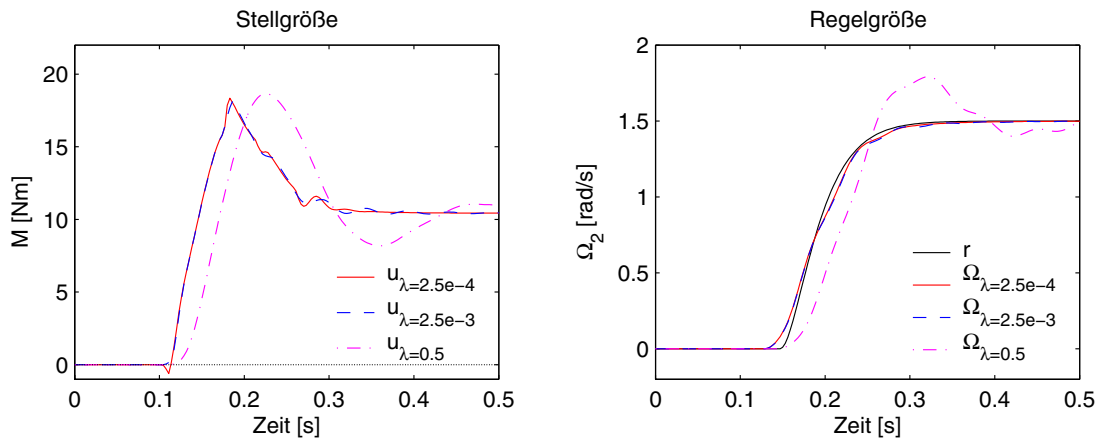
In Bild 8.14 ist ersichtlich, dass bei der Einstellung $N_u = 3$ bereits eine sehr gute Übereinstimmung des Ausgangssignals mit der Referenztrajektorie erzielt wird. Eine weitere Vergrößerung der Stellgröße N_u erhöht den Rechenaufwand, ohne das Folgeverhalten der

Bild 8.14: Variation des Stellhorizontes N_u

Drehzahl Ω_2 noch gravierend zu verbessern. In der Praxis werden bereits sehr gute Ergebnisse für Stellhorizonte zwischen $N_u = 2$ und $N_u = 4$ erzielt.

Einfluss von λ :

Der Faktor λ bewertet die Gewichtung der Stellgrößenänderung Δu in der Kostenfunktion. Ein großer Wert für λ beeinflusst die Beweglichkeit des Systems, da sich die Stellgrößenaktivität Δu verringert. Das Gesamtsystem wird träger und es dauert länger, bis Schwingungen abgeklungen sind. Mit dem Faktor λ kann die Einschwingzeit der Regelung entscheidend beeinflusst werden.

Bild 8.15: Variation der Gewichtung λ

Einfluss von α :

Mit dem Gewichtungsfaktor α wird die Dynamik der Ausregelung des stationären Regelfehlers eingestellt. Bei einem sehr klein gewählten α , wird die aufsummierte Regeldifferenz e_s sehr schwach gewichtet und es dauert länger bis der stationäre Regelfehler den Wert null erreicht. Auf der anderen Seite neigt bei einem großen Wert von α das System bei Übergangsvorgängen zum Schwingen, was zur Instabilität des Prozesses führen kann. In

Bild 8.16 ist das Schwingen bei einem Wert von $\alpha = 0.15$ schon deutlich erkennbar. Es muss daher ein Kompromiss zwischen Dynamik der Fehlerausregelung und Schwingneigung gesucht werden.

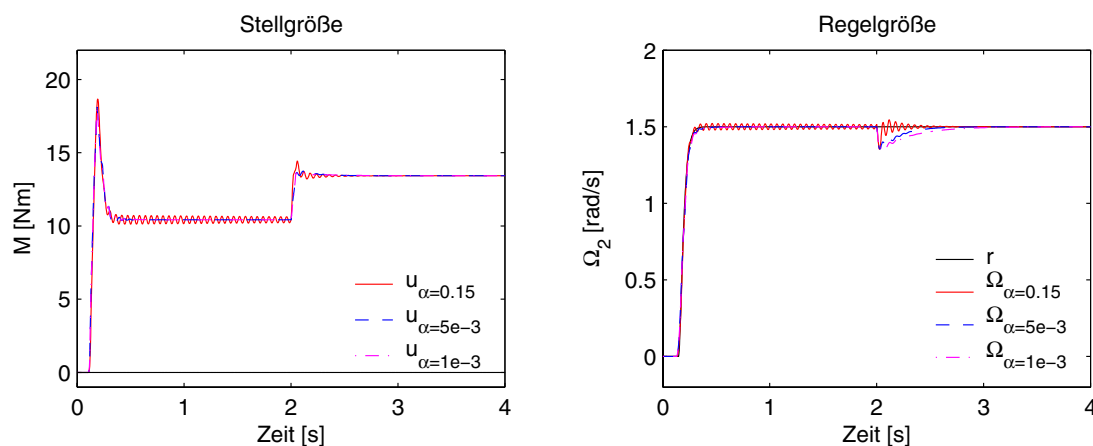


Bild 8.16: Variation des Parameters α

8.4 Robustheitsuntersuchung

Bei der prädiktiven Regelung wird die jeweils aktuelle Stellgröße so von einem Optimierer berechnet, dass zukünftige Regelfehler minimiert werden. Die zukünftigen Regelfehler werden auf Basis eines Prozessmodells geschätzt. Für die Qualität der Regelung ist neben einer sinnvollen Einstellung der Horizonte N_1 , N_2 , N_u und der Gewichtungen λ und α ein möglichst gut mit dem Prozess übereinstimmendes Prädiktionsmodell von entscheidender Bedeutung. Da das Prozessmodell eine so wichtige Grundlage darstellt, wurde in den Kap. 2 und 3 so ausführlich auf die Modellbildung und Identifikation eingegangen.

Wenn sich aufgrund unzureichender Modellbildung oder wegen veränderlicher Parameter Abweichungen zwischen Modell und Prozess ergeben, stellt sich die Frage nach der Robustheit der entworfenen Regelung. Dazu werden im Folgenden die physikalischen Parameter des Zweimassensystems im Prädiktionsmodell nach oben und unten variiert und die Auswirkung auf das Regelergebnis untersucht.

Ausgangspunkt ist das nach Kap. 8.1.2 geregelte ZMS (Daten siehe Anhang A). Es wird die prädiktive Regelung mit Integralanteil untersucht, da diese Variante stationär genau arbeitet und auch in der Praxis vorwiegend zum Einsatz kommen wird. Auch bei Parameterunsicherheiten wird sich demnach stationär genaues Verhalten ergeben, lediglich in der Dynamik sind Unterschiede zu erwarten. Zur Beurteilung von Parameterunsicherheiten wird die Auswirkung auf die Sprungantwort untersucht.

Bild 8.17 links zeigt das Regelergebnis, wenn im Prädiktionsmodell das Trägheitsmoment J_1 um 20% zu niedrig und um 40% zu hoch angenommen wird. Bei zu hoch angenommenem Trägheitsmoment ist die Eigenfrequenz des Modells niedriger als die des realen ZMS. Der Regler versucht diese durch moderate Stellgrößen nicht anzuregen, was letztlich

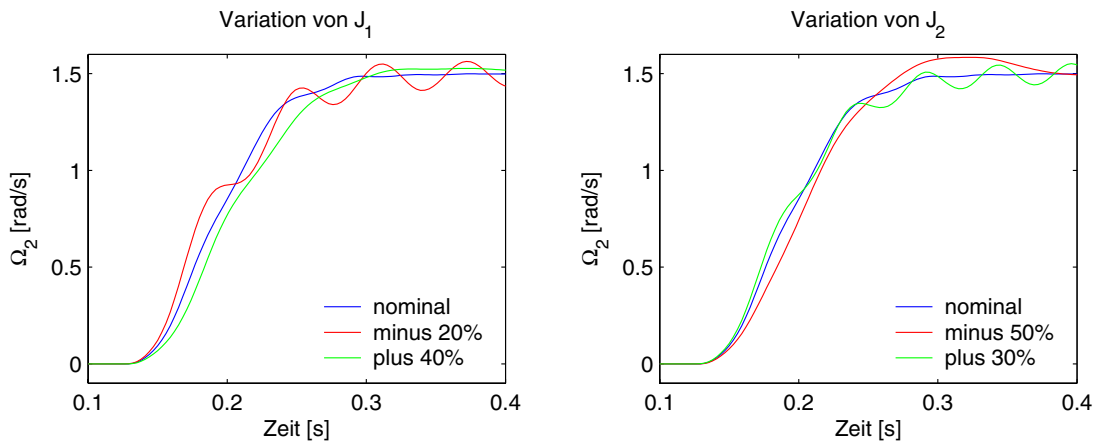


Bild 8.17: Sprungantworten bei Parameterunsicherheiten von J_1 und J_2

in einem langsameren Anstieg resultiert. Bei zu klein angenommenem J_1 drehen sich die Verhältnisse um, wodurch das reale ZMS zu Schwingungen angeregt wird. Bild 8.17 rechts zeigt die Auswirkungen bei falsch angenommenem Trägheitsmoment J_2 . Bei zu großem Trägheitsmoment J_2 im Modell wird vom prädiktiven Regler eine zu große Stellgröße erzeugt, was durch den Integralanteil im Regler korrigiert wird, aber zu Schwingungen führt. Ein zu kleines Trägheitsmoment im Modell bewirkt zunächst eine zu kleine Stellgröße, was zu einer langsameren Systemantwort führt. Der Integralanteil korrigiert dies, erzeugt dabei aber ein Überschwingen.

Eine Variation der Dämpfungskonstanten d um nahezu 100% führt zu keinen sichtbaren Abweichungen im Regelergebnis, weshalb auf diese Darstellung verzichtet wird. Bild 8.18 links zeigt die Auswirkungen einer zu hoch und zu niedrig angesetzten Federsteifigkeit c . In

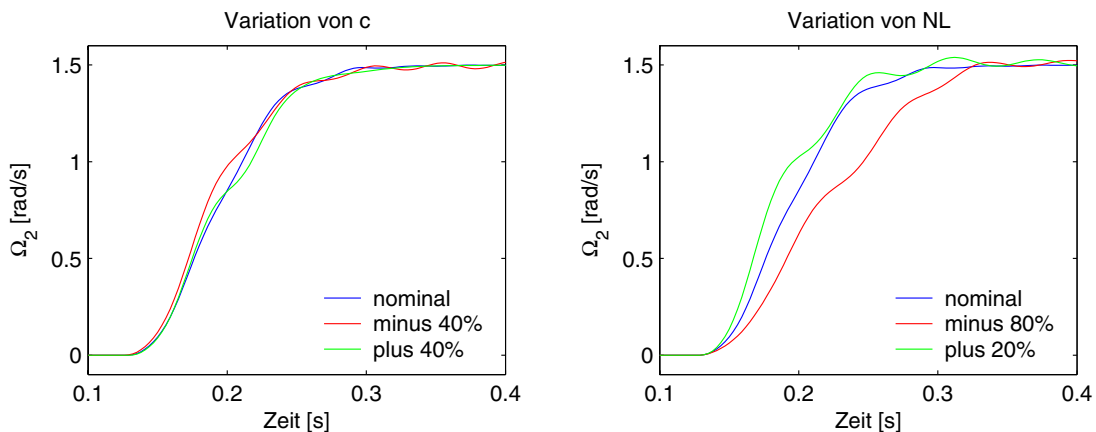


Bild 8.18: Sprungantworten bei Parameterunsicherheiten von c und NL

beiden Fällen ergeben sich selbst bei einer Abweichung von $\pm 40\%$ nur geringe Verschlechterungen. Das liegt daran, dass die Federsteifigkeit nur das Schwingungsverhalten des ZMS beeinflusst, nicht aber das grundsätzlich benötigte Moment für einen Beschleunigungsvorgang (im Gegensatz zu den Trägheitsmomenten). Bei noch größeren Abweichungen kann

es aber trotzdem zur Instabilität des Regelkreises kommen. Kleine Abweichungen hingegen können vom prädiktiven Regler ausgeglichen werden.

Eine wichtige Größe des Prädiktionsmodells stellt schließlich die isolierte Nichtlinearität $\mathcal{NL}$ dar. Bild 8.18 rechts zeigt Variationen von $\mathcal{NL}$ um minus 80% und plus 20%. Dabei wurde ihre Form unverändert belassen, sie wurde lediglich mit einem Faktor multipliziert. Bei zu groß angesetzter Nichtlinearität erzeugt der prädiktive Regler eine zu hohe Stellgröße, was zu einem zu schnellen Anstieg und zu Überspringen führt. Bei zu kleiner Nichtlinearität im Modell hingegen ist die berechnete Stellgröße zu klein. Dies führt zu einem trägeren Anstieg, jedoch ohne Schwingungsneigung.

Bei allen Parametervariationen wurde die Variationsbreite so gewählt, dass zwar Abweichungen sichtbar sind, jedoch noch ein sinnvolles Regelergebnis besteht. Bei allen Parametern sind Abweichungen von mindestens 20% zulässig, ohne dass Instabilität auftritt. Dies ist wichtig für den praktischen Einsatz, wenn einzelne Parameter nicht genau bekannt sind. Es wird aber auch deutlich, dass für hochwertige Regelergebnisse möglichst exakte Prozessmodelle benötigt werden. Starke Abweichungen ergeben sich etwa bei der Nichtlinearität, was die Notwendigkeit einer Identifikation unterstreicht. Die in Kap. 3.3 dargestellte Identifikation einer isolierten Nichtlinearität ist offline und online möglich. Dies eröffnet die Möglichkeit auf veränderliche Nichtlinearitäten während des Betriebs durch eine Online-Identifikation zu reagieren und so immer ein optimales Regelergebnis zu erzielen. Die Möglichkeit der Online-Identifikation wird in Kap. 8.6 dargestellt. Auch bei Variationen der linearen Parameter ergeben sich nur suboptimale Regelergebnisse. Aus diesem Grund ist auch eine lineare Parameteridentifikation gemäß Kap. 2 für ein gutes Regelergebnis äußerst wichtig und unterstreicht die Notwendigkeit einer sorgfältigen Modellbildung. Der Entwurf einer prädiktiven Regelung mit expliziter Berücksichtigung von Parameterunsicherheiten wird in [43] vorgestellt.

8.5 Reglervergleich

Das Lösen der quadratischen Optimierungsaufgabe innerhalb eines Abtastschrittes setzt bei der prädiktiven Regelung unter Berücksichtigung von Nichtlinearitäten und eines Führungsintegrators einen leistungsfähigen Rechner voraus, da durch das zeitvariante Prozessmodell die Prädiktionsmatrizen bei jedem Zeitschritt neu erstellt werden müssen. Es stellt sich die Frage, ob der Mehraufwand eines prädiktiven Reglers gegenüber einem PI- oder Zustandsregler zu einer qualitativen Verbesserung des Regelungsverhaltens führt. Bei der Beurteilung des Rechenaufwands müssen jedoch auch die zusätzlichen Möglichkeiten berücksichtigt werden, die ein prädiktiver Regler durch Einbeziehung von Nebenbedingungen im Regelgesetz bietet.

PI-Regler: Das Zweimassensystem mit zusätzlicher Nichtlinearität an der Lastmaschine kann auch mit einem gewöhnlichen PI-Regler betrieben werden. Wegen des nichtlinearen Einflusses stehen zur Parametereinstellung keine Standardeinstellregeln zur Verfügung. Die Parameter des PI-Reglers werden so ausgelegt, dass stationäre Genauigkeit besteht, der Betrag des Überspringens der Regelgröße Ω_2 möglichst gering ist und der Regler stabil

arbeitet. In praktischen Versuchen wurden folgende Einstellungen als am besten geeignet ermittelt:

$$R(s) = K_p \cdot \frac{1 + s \cdot T_n}{s \cdot T_n} \quad \text{mit} \quad K_p = 1 \quad T_n = 100 \text{ ms} \quad (8.2)$$

Zustandsregler: Auch bei Verwendung eines Zustandsreglers wird stationäre Genauigkeit gefordert. Deshalb wird ein Zustandsregler mit Führungsintegrator nach [29] implementiert. Der zusätzliche Integrator erweitert das System um einen Zustand. Bei der Bestimmung der Rückführkoeffizienten stehen wegen der isolierten Nichtlinearität keine einfachen Einstellregeln zur Verfügung. Es könnte zwar auch das Verfahren der Ein-Ausgangslinearisation angewendet werden [33], was aber wegen der ebenfalls erhöhten Komplexität nicht durchgeführt wird. Die Einstellung des linearen Zustandsreglers erfolgt nach dem Kriterium des Dämpfungsoptimums [29], wobei die Nichtlinearität beim Entwurf außer Acht gelassen wird. Gutes Verhalten des geschlossenen Regelkreises wurde experimentell ermittelt. Dabei musste ein Kompromiss zwischen Schnelligkeit und Schwingungsneigung gefunden werden. Das beste Verhalten wurde durch die Wahl des einzigen freien Reglerparameters zu $T_{ers} = 100 \text{ ms}$ erzielt.

Nun wird der prädiktive Regler mit dem PI- und dem Zustandsregler für die Referenztrajektorien Sprung, Rampe und Sinus verglichen. Der prädiktive Regler wird mit folgenden Parametern eingestellt:

$$N_1 = 3 \quad N_2 = 15 \quad N_u = 3 \quad \lambda = 0.01 \quad \alpha = 0.05 \quad (8.3)$$

Nach der Zeit $t = 4 \text{ s}$ wird dem ZMS ein Störmoment von $z = 4 \text{ Nm}$ aufgeschaltet. In den folgenden Bildern wird jeweils die Regelgröße Ω_2 und die Referenztrajektorie r dargestellt.

Bei der Sprunganregung in Bild 8.19 ist deutlich ersichtlich, dass der prädiktive Regler aufgrund seiner Vorausschau frühzeitig eine Stellgröße u erzeugt, um so der Referenztrajektorie eng folgen zu können. Im Gegensatz hierzu werden die beiden konventionellen Regler erst aktiv, wenn eine Differenz zwischen Soll- und Istwert auftritt. Deshalb haben PI- und Zustandsregler ein trägeres Folgeverhalten. Die Vorausberechnung ermöglicht der prädiktiven Regelung, die Stellgröße u rechtzeitig wieder zu verkleinern, um ein Überspringen zu vermeiden. Dagegen kann ein gewöhnlicher Regler, aufgrund der im System gespeicherten Energie die Regelgröße nicht schnell genug auf den Sollwert führen.

In Bild 8.20 sind die Systemantworten auf eine Rampenanregung dargestellt. Zwischen dem Ausgangssignal des prädiktiven Reglers und der Trajektorie ist so gut wie keine Abweichung festzustellen. Auch die Ausgangsgröße des zustandsgeregelten Systems folgt mit kleiner Verzögerung dem Sollverlauf. Der PI-Regler neigt beim Start der Rampe und beim Eingriff der Störgröße stark zum Schwingen.

Auch bei der Sinus-Solltrajektorie in Bild 8.21 liegt das Ausgangssignal Ω_2 des prädiktiven Reglers direkt auf der Sollwerttrajektorie. Der Zustandsregler folgt dem vorgegebenen Verlauf verzögert und erreicht eine deutlich höhere Amplitude. Bild 8.21 zeigt auch die schlechte Dynamik des PI-Reglers, die es ihm nicht ermöglicht, dem Sinusverlauf wenigstens annähernd zu folgen.

Zusammenfassend zeigt der prädiktive Regler mit Abstand das beste Regelverhalten. Bei dieser Untersuchung wurden Begrenzungen nicht wirksam. Diese können aber sehr ein-

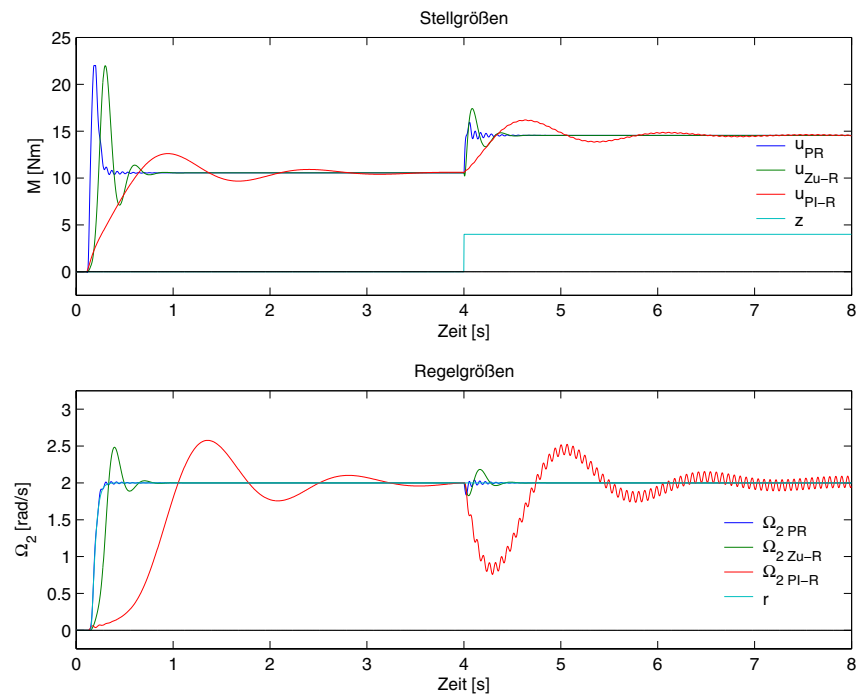


Bild 8.19: Sprungantwort für PI-, Zustands- und prädiktiven Regler

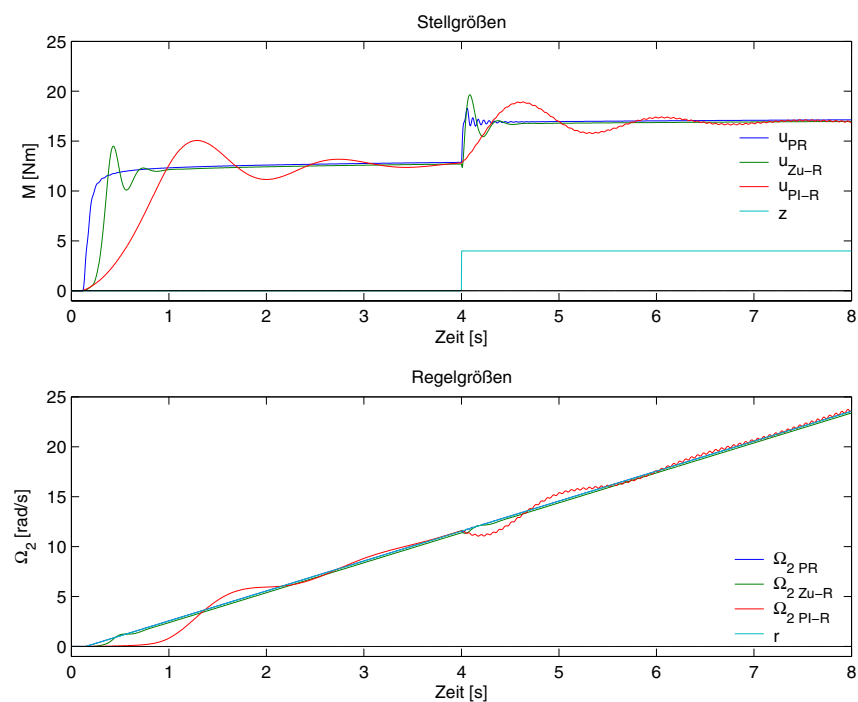


Bild 8.20: Rampenanregung mit PI-, Zustands- und prädiktiver Regelung

fach in den prädiktiven Regler implementiert werden und stellen zudem einen wesentlichen Vorteil der prädiktiven Regelung gegenüber den meisten anderen Verfahren dar. Stellbegrenzungen werden bei Zustands- und PI-Reglern üblicherweise durch Anhalten der Inte-

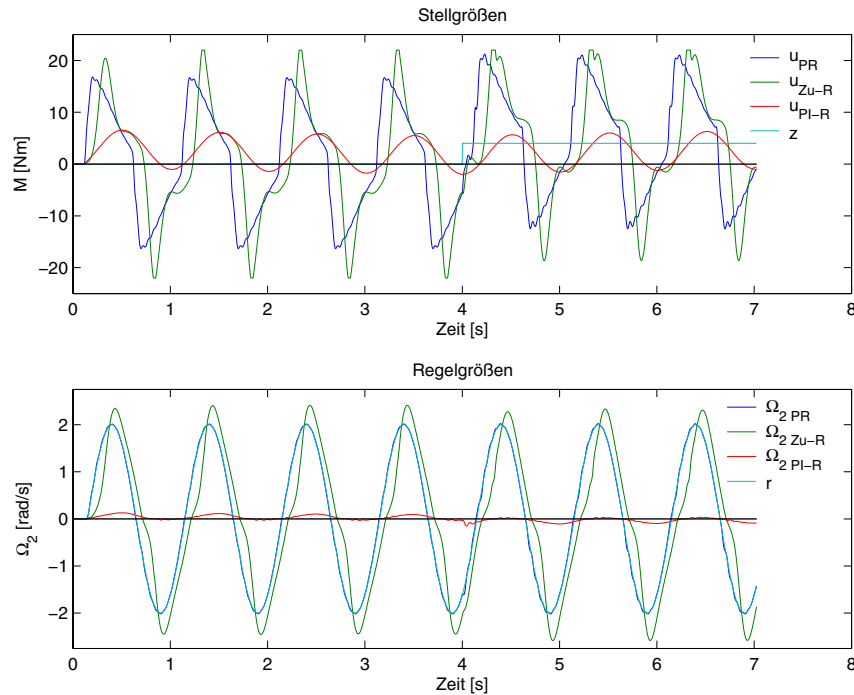


Bild 8.21: Sinusanregung mit PI-, Zustands- und prädiktiver Regelung

gratoren realisiert. Je höher die Anforderungen an die Regelung und an die Einhaltung von Begrenzungen, desto deutlicher kommen die Vorteile des prädiktiven Reglers zum Tragen.

8.6 Prädiktive Regelung mit lernfähigem Beobachter

In den bisherigen Betrachtungen und praktischen Versuchen wurde stets davon ausgegangen, dass der zu regelnde Prozess bekannt ist und dass alle Zustandsgrößen messbar sind. Diese Anforderung ist in vielen Anwendungen nicht gegeben. Selbst wenn nichtlineare Effekte einmalig identifiziert wurden, kann eine langsame Zeitvarianz durch Alterung oder Temperaturänderungen bestehen. Die Relevanz eines möglichst genauen Prozessmodells wurde in Kap. 8.4 besonders deutlich. Bei zu großen Abweichungen vom realen Prozess kommt es zu einem verschlechterten Regelergebnis. Dies ist ausschließlich auf das abweichende Prozessmodell zurückzuführen und kann auch durch veränderte Reglereinstellungen nicht verbessert werden. Aus diesem Grund ist die vorgestellte prädiktive Regelung mit lernfähigem Zustandsraummodell ein Schritt hin zu einem stets optimal eingestellten Prozessmodell. Bei Veränderungen am realen Prozess wird das Modell online adaptiert und die in Kap. 8.4 festgestellten negativen Auswirkungen werden vermieden. Ebenso muss für die Bestimmung der aktuellen Zustandsgrößen eine praxistaugliche Lösung gefunden werden. Die hohen Kosten für Sensoren lassen ihren großzügigen Einsatz kritisch erscheinen. Es besteht auch die Möglichkeit, dass sich manche Zustandsgrößen als nicht messbar erweisen, weil sie im Prozess nicht messtechnisch zugänglich sind. Aus diesen Gründen muss eine Methode gewählt werden, die nicht messbaren Zustandsgrößen auf andere Weise zu

bestimmen. Dazu soll der in Kap. 3.3 eingeführte lernfähige Beobachter online eingesetzt werden. Er liefert als zusätzliche Prozessinformation vorhandene nichtlineare Effekte als Kennlinien und Schätzwerte aller nicht messbaren Zustandsgrößen.

8.6.1 Besonderheit des lernfähigen Beobachters mit prädiktivem Regler

Aus der Herleitung der Prädiktionsgleichung in Kap. 6.1.2 geht hervor, dass für die Linearisierung erster Ordnung entlang der Referenztrajektorie die Kenntnis der nichtlinearen Funktion und deren Ableitung benötigt wird. Da der lernfähige Beobachter nur die Nichtlinearität selbst liefert, sind einige Modifikationen nötig. Dazu wird das GRNN des lernfähigen Beobachters in ein *Identifikations-GRNN* und ein *Approximations-GRNN* aufgeteilt. Das Identifikations-GRNN ist in den Beobachter integriert und liefert nach einer Lernphase die geschätzte Nichtlinearität zurück. Diese Information wird dem Approximations-GRNN in Form der Netzwerkgewichte $\hat{\Theta}$ übergeben (Bild 8.22). Daher kann das Lern-

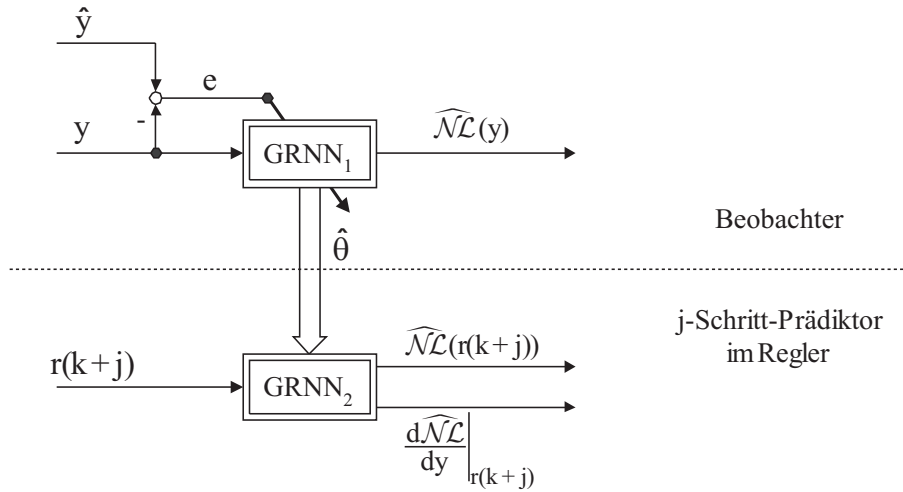


Bild 8.22: Zusammenspiel des GRNN im Beobachter mit dem GRNN im Regler

gesetz im Approximations-GRNN entfallen. Das GRNN des Reglers lernt damit nicht selbst und dient nur zur Funktionsauswertung für die Prädiktion. Das Identifikations-GRNN im Beobachter wird durch das Lerngesetz in Gleichung (3.22) oder (3.58) vom Beobachterfehler e adaptiert, was in Bild 8.22 durch den Pfeil im GRNN angedeutet ist. Die Stützwerte werden an das Approximations-GRNN bei jedem Zeitschritt im Vektor $\hat{\Theta}$ neu übergeben. Im Gegensatz zum Identifikations-GRNN im Beobachter muss das Approximations-GRNN zusätzlich zum Funktionswert $\mathcal{N}(y)$ auch die Ableitung der Nichtlinearität berechnen. Dies geschieht durch Differentiation der Approximationsgleichung (3.72) des GRNN. Es ergibt sich

$$\frac{d\hat{\mathcal{N}}(y)}{dy} = \hat{\Theta}^T \cdot \frac{d\mathcal{A}(y)}{dy} \quad (8.4)$$

woraus ersichtlich ist, dass lediglich die Ableitung der normierten Aktivierung berechnet werden muss. Es können die selben Stützwerte wie zur Funktionsauswertung verwendet

werden. Der Aktivierungsvektor wird mit dem Abstandsquadrat nach Gleichung (3.12) in Zähler und Nenner aufgespalten:

$$\mathcal{A}(y) = \frac{\mathcal{A}'(y)}{\sum_{k=1}^p \mathcal{A}'_k(y)} \quad \text{mit} \quad \mathcal{A}' = \exp\left(-\frac{\mathcal{C}}{2\sigma^2}\right) \quad (8.5)$$

Dies führt zum Ausdruck

$$\frac{d\mathcal{A}(y)}{dy} = \frac{\sum_{k=1}^p \mathcal{A}'_k(y) \cdot \frac{d\mathcal{A}'_k(y)}{dy} - \sum_{k=1}^p \frac{d\mathcal{A}'_k(y)}{dy} \cdot \mathcal{A}'(y)}{\left(\sum_{k=1}^p \mathcal{A}'_k(y)\right)^2} \quad (8.6)$$

mit dem die Ableitung der normierten Aktivierung bestimmt werden kann. Der noch unbekannte Vektor $d\mathcal{A}'(y)/dy$ kann durch Differentiation der Gleichung (3.11) erhalten werden:

$$\frac{d\mathcal{A}'_i(y)}{dy} = -\frac{y - \chi_i}{\sigma^2} \exp\left(-\frac{\mathcal{C}_i}{2\sigma^2}\right) \quad (8.7)$$

$$\Rightarrow \frac{d\mathcal{A}'(y)}{dy} = \left[\frac{d\mathcal{A}'_1(y)}{dy}, \frac{d\mathcal{A}'_2(y)}{dy}, \dots, \frac{d\mathcal{A}'_p(y)}{dy} \right]^T \quad (8.8)$$

Aus Gleichung (8.6) und (8.8) lässt sich schließlich die differenzierte Aktivierung berechnen, die mit Gleichung (8.4) eine Schätzung der abgeleiteten Nichtlinearität ermöglicht.

Die Gesamtstruktur der prädiktiven Regelung mit lernfähigem Beobachter ist in Bild 8.23 gezeigt. Der lernfähige Beobachter mit GRNN hat die Aufgabe, einen Schätzwert für den Zustandsvektor $\mathbf{x}$ zu liefern und die im Prädiktor benötigte isolierte Nichtlinearität und deren Ableitung zu schätzen. Diese geschätzten Größen werden abgetastet und im Prädiktor anstelle der bisher als bekannt angenommenen Größen verwendet. Die Abtastung ist nötig, weil das prädiktive Regelgesetz zeitdiskret arbeitet.

8.6.2 Ergebnisse

Die prädiktive Regelung wurde um den lernfähigen Zustandsbeobachter erweitert. Getestet wurde das Zusammenspiel von prädiktiver Regelung und lernfähigem Beobachter am Zweimassensystem aus Anhang A. Dabei wurden zwei Lernvorgänge untersucht, die sich durch die Parametrierung des neuronalen Netzes im Beobachter unterscheiden. Zunächst wurde eine Einstellung gewählt, die eine gute Identifikation der Nichtlinearität ermöglicht. In einem zweiten Lernvorgang wurde die Anzahl der Stützstellen reduziert und der Glättungsfaktor σ erhöht. Dadurch kann der steile Übergang der Arcus-Tangensfunktion nicht befriedigend nachgebildet werden. In den Bildern 8.24 und 8.25 sind die Identifikations- und Regelergebnisse dargestellt. Die beiden oberen Diagramme zeigen den Verlauf von Soll- und Istwert der Regelung für die ersten bzw. letzten 20 Sekunden des Lernvorganges. Darunter ist jeweils die reale und gelernte Nichtlinearität dargestellt, nach 20 bzw. nach 200

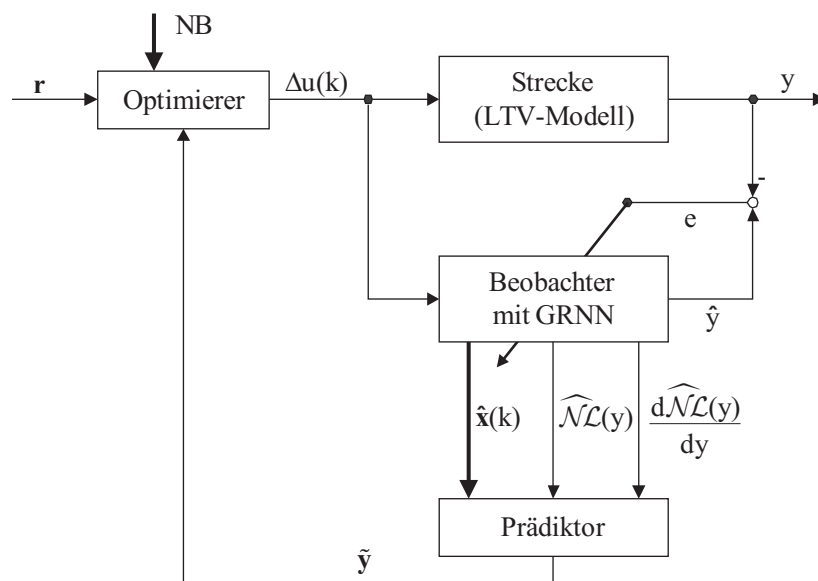


Bild 8.23: Struktur der prädiktiven Regelung mit Beobachter

Sekunden. In beiden Fällen ist zu erkennen, dass zu Beginn des Lernvorganges die unberücksichtigte Nichtlinearität noch zu einer verminderten Lastdrehzahl führt. Der maximale Sollwert im Scheitelpunkt der Sinusanregung wird nicht erreicht, da die Prädiktion der Ausgangsgröße Ω_2 zunächst ohne Berücksichtigung der bremsenden Wirkung der Nichtlinearität erfolgt. Beim Wechsel der Drehrichtung neigt die Drehzahl Ω_2 zum *Hängenbleiben*. Nach 180 Sekunden ist dagegen ein sehr gutes Folgeverhalten zu erkennen. Die identifizierte Kennlinie stimmt am Ende des Lernvorganges gut mit der realen Nichtlinearität überein. Lediglich die Steigung im Ursprung kann nicht exakt erreicht werden. Im zweiten Lernvorgang ist der Glättungsfaktor σ zu groß gewählt, so dass die Kennlinie nicht ausreichend genau identifiziert wird. Allein durch Verlängerung der Lerndauer ist bei diesen Einstellungen keine Verbesserung des Lernergebnisses erzielbar. Die zu große Glättung verhindert einen steilen Anstieg um den Ursprung, weshalb die Stick-Slip-Bewegung erhalten bleibt. Die Ergebnisse aus Bild 8.24 belegen, dass die Kombination aus modellbasierter prädiktiver Regelung mit einem lernfähigen Beobachter gute Regelergebnisse liefert. Dabei erzeugt der Beobachter eine Schätzung für den aktuellen Zustandsvektor, den Wert der Nichtlinearität und deren Ableitung. Da es sich hier um ein adaptives System handelt, muss die nichtlineare Funktion nicht beim Reglerentwurf bekannt sein, sondern wird während einer Lernphase im laufenden Betrieb identifiziert. Dadurch ist die vorgestellte prädiktive Regelung auch auf Prozesse mit unbekannter Nichtlinearität und nicht vollständig messbarem Zustandsvektor anwendbar.

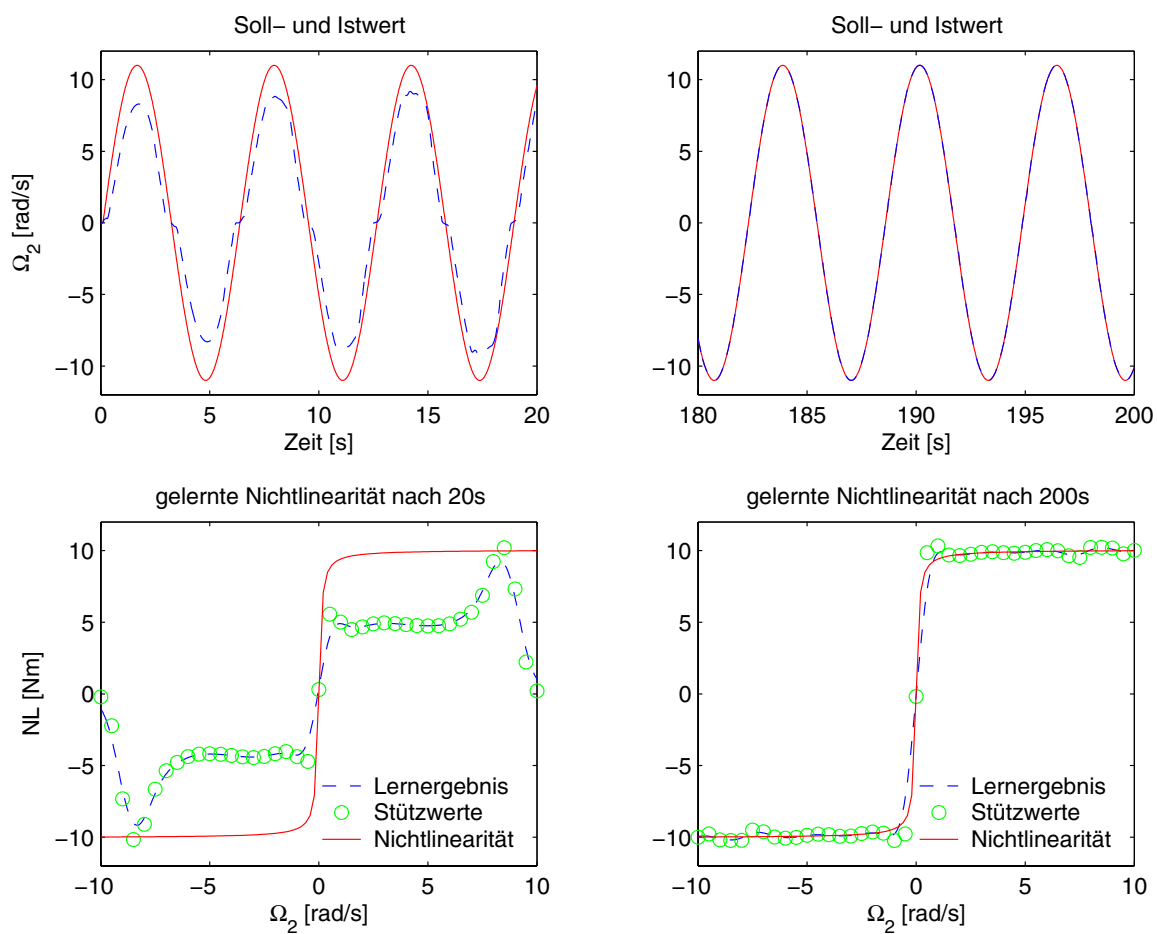


Bild 8.24: Lernvorgang am Zweimassensystem mit geeigneten Einstellungen

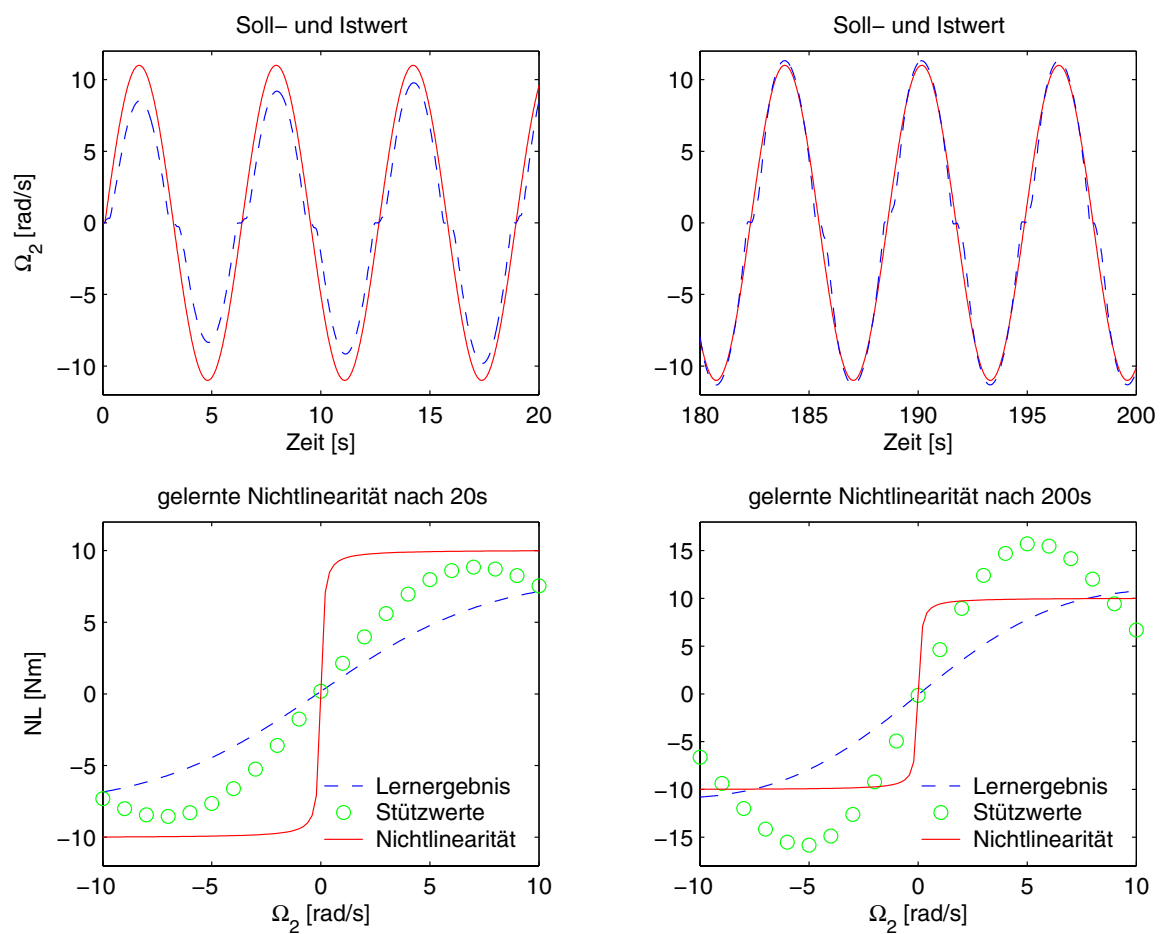


Bild 8.25: Lernvorgang am Zweimassensystem mit ungeeigneten Einstellungen

9 Mehrgrößenprozesse

Prädiktive Regelalgorithmen eignen sich ideal zur Anwendung auf Mehrgrößenprozesse, da mehrere Regelziele im Gütefunktional gleichzeitig spezifiziert werden können. Der Ursprung der prädiktiven Regelung liegt in der chemischen, Erdöl und verfahrenstechnischen Industrie. Sie kam vorwiegend zur Regelung linearer Mehrgrößensysteme mit einer großen Zahl von Eingangs- und Regelgrößen unter Einbeziehung von Begrenzungen zum Einsatz. Die Einbeziehung nichtlinearer Effekte in die Modellierung von MIMO-Prozessen gewinnt sowohl durch höhere Anforderungen an die Regelung, als auch durch die fortschreitende Mikroprozessortechnik an Bedeutung. Auch im Bereich der Antriebstechnik größerer Anlagen (z.B. kontinuierliche Fertigung, Druckmaschinen, oder Teleskope in Kap. 9.2) hat man es meist mit mehreren Stell- und Regelgrößen zu tun. Deshalb ist für dieses Anwendungsfeld eine MIMO-Erweiterung sinnvoll. Die hier gewählte Prozessmodellierung als Zustandsraummodell macht eine Erweiterung von SISO-Prozessen auf MIMO-Prozesse besonders einfach, da diese Darstellung grundsätzlich keinen Unterschied zwischen Ein- und Mehrgrößenprozessen macht.

9.1 Erweiterung auf Mehrgrößenprozesse

Ausgehend von den in Kap. 6 hergeleiteten Zustandsgleichungen des SISO-Systems, vergrößert sich beim Mehrgrößenprozess die Stellgrößenänderung $\Delta u(k)$ zum Stellgrößenänderungsvektor $\Delta \mathbf{u}(k)$, der Einkoppelvektor $\mathbf{b}$ geht in eine Einkoppelmatrix $\mathbf{B}$ über, der Durchgriff d vergrößert sich zu einer Durchgriffsmatrix $\mathbf{D}$ und der Auskoppelvektor $\mathbf{c}$ wird zur Auskoppelmatrix $\mathbf{C}$. Die Ausgangsgrößen werden zum Ausgangsvektor $\mathbf{y}(k)$ zusammengefasst. Daraus ergibt sich die Zustandsdarstellung des MIMO-Systems N -ter Ordnung mit m Ausgängen und o Eingängen zu:

$$\mathbf{x}(k+1) = \mathbf{A} \cdot \mathbf{x}(k) + \mathbf{B} \cdot \Delta \mathbf{u}(k) + \mathbf{k} \cdot \mathcal{NL}(y_i) \quad (9.1)$$

$$\mathbf{y}(k) = \mathbf{C} \cdot \mathbf{x}(k) + \mathbf{D} \cdot \Delta \mathbf{u}(k)$$

Hier wurde bereits die um die Zustände $u_i(k-1)$ erweiterte Prozessbeschreibung nach Gleichung (6.4) zugrunde gelegt. Außerdem wird von *einer* Nichtlinearität mit der Abhängigkeit von einem beliebigen Ausgangssignal y_i angenommen. Die Einbeziehung mehrerer Nichtlinearitäten kann gemäß Kap. 6.6 erfolgen. Mit der Taylorreihe lässt sich die Nichtlinearität $\mathcal{NL}$ in erster Ordnung annähern. Die linearisierte Zustandsdarstellung ergibt sich dann als zeitvariantes lineares System.

$$\mathbf{x}(k+1) = \mathbf{A}(k) \cdot \mathbf{x}(k) + \mathbf{B}(k) \cdot \Delta \mathbf{u}(k) + \mathbf{k} \cdot v(k) \quad (9.2)$$

$$\mathbf{y}(k) = \mathbf{C} \cdot \mathbf{x}(k) + \mathbf{D} \cdot \Delta \mathbf{u}(k) \quad (9.3)$$

$\mathbf{A}(k)$, $\mathbf{B}(k)$ und $v(k)$ sind gemäß den Gleichungen (6.32) bis (6.34) definiert. Im Hinblick auf eine übersichtliche Darstellung, erfolgt die Herleitung der j -Schritt Prädiktion des MIMO-Systems ohne Berücksichtigung des LQI-Reglers aus Kap. 6.4. Bei der j -Schritt-Vorhersage

für ein Mehrgrößensystem berechnet sich der Auskoppelvektor $\tilde{\mathbf{y}}(k+j)$ für jeden Zeitschritt zu:

$$\begin{aligned}\tilde{\mathbf{y}}(k+j) &= \mathbf{C} \left[\prod_{n=j-1}^0 \mathbf{A}(k+n) \right] \mathbf{x}(k) + \\ &+ \sum_{i=0}^{j-2} \mathbf{C} \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \mathbf{B}(k+i) \Delta \mathbf{u}(k+i) + \\ &+ \mathbf{C} \mathbf{B}(k+j-1) \Delta \mathbf{u}(k+j-1) + \mathbf{D} \Delta \mathbf{u}(k+j) + \\ &+ \sum_{i=0}^{j-2} \mathbf{C} \left[\prod_{n=j-1}^{i+1} \mathbf{A}(k+n) \right] \mathbf{k} v(k+i) + \mathbf{C} \mathbf{k} v(k+j-1)\end{aligned}\quad (9.4)$$

Die im Prädiktionsvektor $\tilde{\mathbf{y}}$ zusammengefassten Ausgangssignale lassen sich wieder kompakt in Vektor-Matrixschreibweise darstellen.

$$\tilde{\mathbf{y}} = \mathbf{F}^\diamond \mathbf{x}(k) + \mathbf{H}^\diamond \Delta \mathbf{u} + \mathbf{G}^\diamond \mathbf{v} \quad (9.5)$$

Die $\mathbf{F}^\diamond$ -Matrix beschreibt die freie Systemantwort ohne äußere Anregung. Die Stellgrößenänderungen $\Delta \mathbf{u}$ werden durch die $\mathbf{H}^\diamond$ -Matrix berücksichtigt und mit der $\mathbf{G}^\diamond$ -Matrix wird die Auswirkung der Nichtlinearität auf die Ausgänge beschrieben. Die Erweiterung auf ein Mehrgrößensystem verändert lediglich die Dimension der einzelnen Matrizen.

$$\mathbf{F}^\diamond = \begin{bmatrix} \mathbf{C} \prod_{n=N_1-1}^0 \mathbf{A}(k+n) \\ \mathbf{C} \prod_{n=N_1}^0 \mathbf{A}(k+n) \\ \vdots \\ \mathbf{C} \prod_{n=N_2-1}^0 \mathbf{A}(k+n) \end{bmatrix} \in \mathbb{R}^{(N_2-N_1+1) \cdot m \times N} \quad (9.6)$$

$$\mathbf{H}^\diamond = \begin{bmatrix} \mathbf{H}_{N_1-1, N_1-1} & \cdots & \mathbf{H}_{N_1-1, N_1-N_u-1} \\ \vdots & & \vdots \\ \mathbf{H}_{N_2-1, N_2-1} & \cdots & \mathbf{H}_{N_2-1, N_2-N_u-1} \end{bmatrix} \in \mathbb{R}^{(N_2-N_1+1) \cdot m \times (N_u+1) \cdot o} \quad (9.7)$$

mit

$$\mathbf{H}_{p,q} = \begin{cases} \mathbf{C} \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{B}(k+p-q) & \text{für } q > 0 \\ \mathbf{C} \mathbf{B}(k+p-q) & \text{für } q = 0 \\ \mathbf{D} & \text{für } q = -1 \\ 0 & \text{für } q < -1 \end{cases} \quad (9.8)$$

$$\mathbf{G}^\diamond = \begin{bmatrix} \mathbf{G}_{N_1-1, N_1-1} & \cdots & \mathbf{G}_{N_1-1, N_1-N_2} \\ \vdots & & \vdots \\ \mathbf{G}_{N_2-1, N_2-1} & \cdots & \mathbf{G}_{N_2-1, N_2-N_2} \end{bmatrix} \in \mathbb{R}^{(N_2-N_1+1) \cdot m \times N_2} \quad (9.9)$$

mit

$$\mathbf{G}_{p,q} = \begin{cases} \mathbf{C} \left[\prod_{n=p}^{p-q+1} \mathbf{A}(k+n) \right] \mathbf{k} & \text{für } q > 0 \\ \mathbf{C} \mathbf{k} & \text{für } q = 0 \\ 0 & \text{für } q < 0 \end{cases} \quad (9.10)$$

Die Vektoren der Ausgangssignale und Stellgrößenänderungen werden hintereinander gesetzt, so dass sich jeweils ein verlängerter Vektor ergibt.

$$\tilde{\mathbf{y}} = [\tilde{\mathbf{y}}^T(k+N_1), \tilde{\mathbf{y}}^T(k+N_1+1), \dots, \tilde{\mathbf{y}}^T(k+N_2)]^T \quad (9.11)$$

$$\Delta \mathbf{u} = [\Delta \mathbf{u}^T(k), \Delta \mathbf{u}^T(k+1), \dots, \Delta \mathbf{u}^T(k+N_u)]^T \quad (9.12)$$

Die m Sollwerte der zugehörigen Ausgangssignale werden im Sollwertvektor $\mathbf{r}(k)$ zusammengefasst.

$$\mathbf{r}(k) = [r_1(k), r_2(k), \dots, r_m(k)]^T \quad (9.13)$$

Als Kostenfunktion für ein Mehrgrößensystem eignet sich

$$\begin{aligned} J(\Delta \mathbf{u}) &= \sum_{j=N_1}^{N_2} (\mathbf{r}(k+j) - \tilde{\mathbf{y}}(k+j))^T \mathbf{\Gamma} (\mathbf{r}(k+j) - \tilde{\mathbf{y}}(k+j)) \\ &+ \sum_{j=0}^{N_u} \Delta \mathbf{u}^T(k+j) \mathbf{\Lambda} \Delta \mathbf{u}(k+j) \end{aligned} \quad (9.14)$$

die sich auch im DMC-Verfahren wiederfindet [66]. In diesem Zusammenhang ist

- $\mathbf{\Gamma}$ die Wichtungsmatrix der m Regelgrößenabweichungen mit

$$\mathbf{\Gamma} = \text{diag}(\gamma_1, \dots, \gamma_m) \quad \gamma_i \geq 0 \quad (9.15)$$

- $\mathbf{\Lambda}$ die Wichtungsmatrix der Stellgrößenänderung der o Stellgrößen mit

$$\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_o) \quad \lambda_i \geq 0 \quad (9.16)$$

Mit der Diagonalmatrix $\mathbf{\Gamma}$ können einzelne Abweichungen in den Regelgrößen priorisiert werden, indem diese Regelabweichungen im Verhältnis zu den anderen Regelabweichungen stärker gewichtet werden. Analog können mit der $\mathbf{\Lambda}$ -Matrix einzelne Stellgrößenänderungen vorrangig bestraft werden.

Mit Gleichung (9.5) und (9.14) lässt sich die Kostenfunktion in vektorieller Form darstellen.

$$J(\Delta \mathbf{u}) = \Delta \mathbf{u}^T (\mathbf{H}^{\diamond T} \mathbf{\Gamma} \mathbf{H}^{\diamond} + \mathbf{\Lambda}) \Delta \mathbf{u} - 2 \mathbf{d}^T \mathbf{\Gamma} \mathbf{H}^{\diamond} \Delta \mathbf{u} + \mathbf{d}^T \mathbf{\Gamma} \mathbf{d} \quad (9.17)$$

mit

$$\mathbf{d} = \mathbf{r} - \mathbf{F}^{\diamond} \mathbf{x}(k) - \mathbf{G}^{\diamond} \mathbf{v}$$

Der Vektor $\mathbf{r}$ enthält die Sollwerte der m Regelgrößen innerhalb des gesamten Prädiktionshorizonts, der Vektor $\mathbf{v}$ beinhaltet die Einflüsse der Nichtlinearität bis zum oberen Prädiktionshorizont N_2 .

$$\mathbf{r} = [\mathbf{r}^T(k+N_1), \mathbf{r}^T(k+N_1+1), \dots, \mathbf{r}^T(k+N_2)]^T \quad (9.18)$$

$$\mathbf{v} = [v(k), \dots, v(k+N_2-1)]^T \quad (9.19)$$

Die Berechnung des Optimums ohne Nebenbedingungen erfolgt durch Nullsetzen des Gradienten von Gleichung (9.17) nach $\Delta \mathbf{u}$. Als optimale Stellgrößenfolge ergibt sich

$$\Delta \mathbf{u}^* = (\mathbf{H}^{\diamond T} \mathbf{\Gamma} \mathbf{H}^{\diamond} + \mathbf{\Lambda})^{-1} \mathbf{H}^{\diamond T} \mathbf{\Gamma} \mathbf{d} \quad (9.20)$$

Im Sinne des Prinzips des zurückweichenden Horizonts wird nur jeweils das erste Element jeder der o Stellgrößenänderungen auf den Prozess geführt, alle anderen Elemente werden verworfen. Durch die Anordnung innerhalb des $\Delta \mathbf{u}$ -Vektors in Gleichung (9.12) bedeutet dies, dass die ersten o Elemente von $\Delta \mathbf{u}$ dem Prozess zugeführt werden.

Zur numerischen Bestimmung des Optimums unter Berücksichtigung von Nebenbedingungen nach Kap. 6.5, kann der von $\Delta \mathbf{u}$ unabhängige Term $\mathbf{d}^T \mathbf{\Gamma} \mathbf{d}$ im Gütefunktional (9.17) ohne Beeinflussung des Minimums entfallen. Um die Standardform eines quadratischen Programms herzustellen, wird Gleichung (9.17) noch mit dem Faktor $1/2$ multipliziert, um schließlich zum äquivalenten Gütefunktional (9.21) zu gelangen.

$$J^*(\Delta \mathbf{u}) = \frac{1}{2} \Delta \mathbf{u}^T (\mathbf{H}^{\diamond T} \mathbf{\Gamma} \mathbf{H}^{\diamond} + \mathbf{\Lambda}) \Delta \mathbf{u} - \mathbf{d}^T \mathbf{\Gamma} \mathbf{H}^{\diamond} \Delta \mathbf{u} \quad (9.21)$$

Dieses Gütefunktional kann direkt an numerische Algorithmen zur Lösung von quadratischen Programmen übergeben werden [86, 48]. Die Nebenbedingungen werden gemäß Kap. 6.5 spezifiziert und in Vektor-Matrix-Ungleichungen umformuliert. Bei den simulativen Untersuchungen in Kap. 9.2 wurde der Algorithmus `qld` von Prof. Schittkowski verwendet [86].

9.2 Anwendung der prädiktiven Regelung auf ein Radioteleskop

Im Rahmen eines Industrieprojekts wurde in [57] am 100 m Radioteleskop in Effelsberg die drehzahlabhängige Reibungskennlinie zwischen Rädern und Schiene mittels lernfähigem Beobachter identifiziert. Mittels einfacher Störkompensation wurde anschließend der Haftreibungseinfluss (Stick-Slip-Effekt) beim Geschwindigkeitsnulldurchgang der Räder weitestgehend kompensiert.

In diesem Kapitel wird die Reibkennlinie als bekannt angenommen und das Regelergebnis am Radioteleskop soll weiter verbessert werden. Dazu erfolgt zunächst eine detaillierte Beschreibung und Modellierung des Systems.

9.2.1 Beschreibung des Systems und Modellierung

Sehr große Radioteleskope können wegen des hohen Gewichts und der damit verbundenen Belastungen nicht mehr mit Drehgelenken gelagert werden, sondern müssen mit einem Rad-Schiene-System ausgerüstet werden. Dabei befinden sich an den Ecken eines Grundrahmes jeweils eine bestimmte Anzahl von massiven Rädern, die teilweise angetrieben sind und auf einer kreisförmigen Schiene umlaufen. Dabei ruht das gesamte Gewicht des Teleskops auf diesen Rädern. Eine Lagerung im Mittelpunkt des Schienenkreises sorgt nur noch für die Zentrität der Kreisbewegung. Durch das hohe Gewicht derartiger Anordnungen

(in Effelsberg etwa 3200 Tonnen) ist die Flächenpressung zwischen Rädern und Schiene so hoch, dass massive Reibungseinflüsse entstehen. Die beim Geschwindigkeits-Nulldurchgang auftretende Stick-Slip-Bewegung kann dabei bei der extrem großen Entfernung der beobachteten Objekte im Weltall von bis zu 8 Milliarden Lichtjahren (1 Lichtjahr $\approx 9,5 \cdot 10^{12}$ Kilometer) erhebliche Abweichungen vom Zielgebiet verursachen. Das Radioteleskop in Effelsberg ist trotz seiner Betriebszeit von mittlerweile über 30 Jahren immer noch die größte vollbewegliche Antenne mit Parabolreflektor der Welt ist. Die technischen Daten des Radioteleskops in Effelsberg sind in Tabelle 9.1 zusammengefasst.

Reflektordurchmesser:	100 m
Geometrische Antennenfläche:	7850 m ²
Brennweite:	30 m
max. Abweichung von der idealen Parabolform:	$\leq 0,5$ mm
Auflösung bei 3,5 mm Wellenlänge (86 GHz):	10 Bogensekunden
Drehbereich in Azimutrichtung:	480°
größte Drehgeschwindigkeit in Azimutrichtung:	32°/min
Drehbereich in Elevationsrichtung:	7° bis 94°
größte Drehgeschwindigkeit in Elevationsrichtung:	16°/min
empfangbarer Wellenlängenbereich:	0,35 mm bis 90 cm
Durchmesser des Schienenkreises:	64 m
Durchmesser eines Rades:	0,98 m
Anzahl der Räder:	32
Anzahl der angetriebenen Räder:	16
Leistung der 16 Antriebsmotoren:	je 10,2 kW
Leistung der 4 Kippmotoren:	je 17,5 kW
Getriebeübersetzung der Radantriebe:	423,4
Gesamtgewicht:	3200 t
Masse eines Rades:	1000 kg
niedrigste Eigenfrequenz:	ca. 1 Hz

Tabelle 9.1: Technische Daten des Radioteleskops in Effelsberg

In Bild 9.1 ist eine Aufnahme des Radioteleskops zu sehen, wie es sich kurz vor Ende der Bauphase darstellte.

Um ein möglichst einfaches Modell niedriger Ordnung für das Radioteleskop zu erhalten, wurde die Anlage als Mehrmassenmodell beschrieben. Zur Modellierung wurde nur die Bewegung in Azimutrichtung, also eine Drehung um die vertikale Achse berücksichtigt. Der Reflektor steht während der Bewegung still und befindet sich in der 90° Position. Das Radioteleskop wird vereinfacht als Fünfmassensystem modelliert. Die Massen werden vom Reflektor, der Azimut Plattform, dem Grundrahmen und zwei Gruppen von Motoren gebildet. Die Motoren werden in zwei Gruppen verschaltet, um durch gegenseitige Verspannung den Loseinfluss der Getriebe zu unterdrücken. Jeweils eine Gruppe wirkt antreibend, die andere bremsend. Dadurch wird der Losebereich nie durchfahren und ist daher auch nicht wirksam. Jeweils eine Gruppe erhält einen gemeinsamen Drehmoment-sollwert. Das Schemabild des Radioteleskops ist in Bild 9.2 dargestellt. Das Radioteleskop

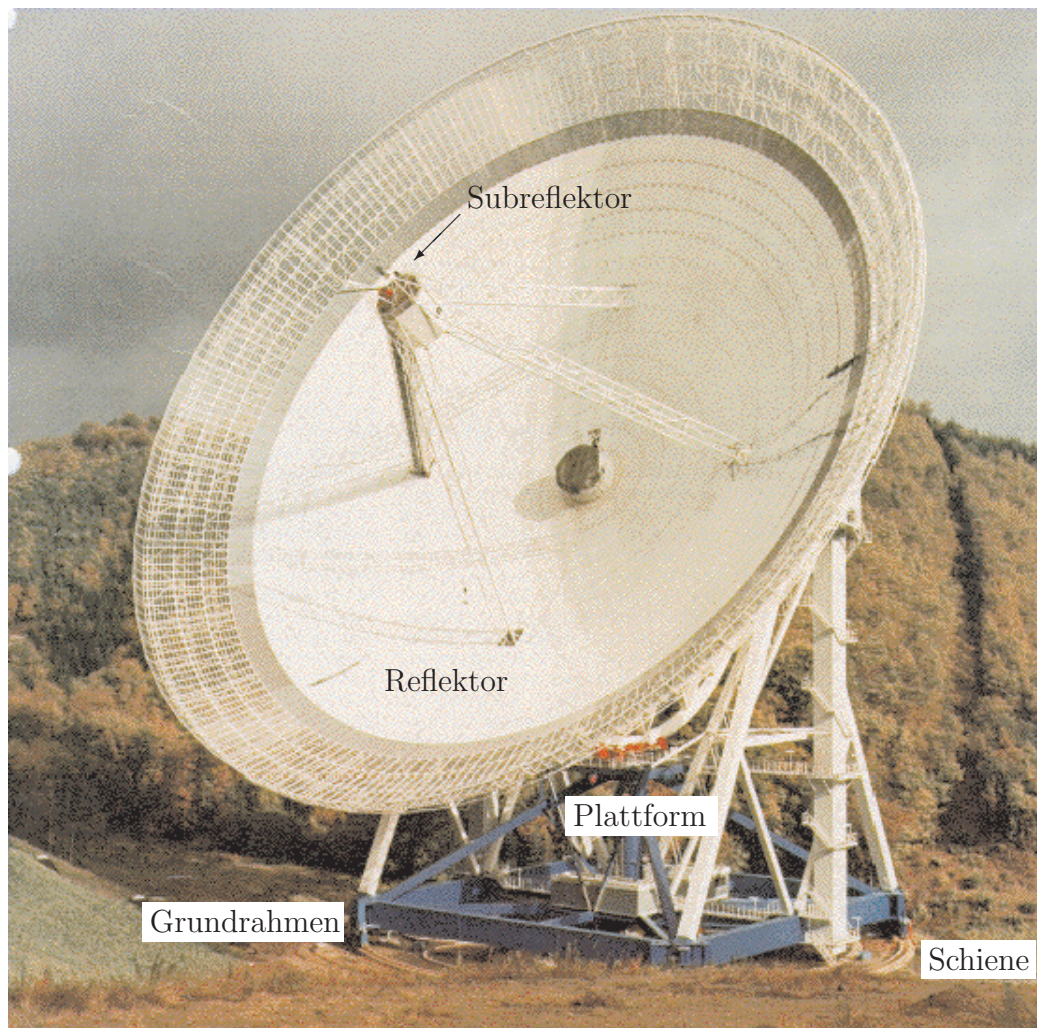


Bild 9.1: Radioteleskop in Effelsberg

wird mit den beiden Motorgruppen 1 und 2 angetrieben. Die nichtlineare Reibung wirkt auf das Trägheitsmoment J_3 des Grundrahmens. Die Azimut Plattform und der Reflektor werden mit den Trägheitsmomenten J_4 und J_5 nachgebildet. Messtechnisch zur Verfügung stehen die beiden Drehzahlen der Motorgruppen ($\dot{\phi}_1$, $\dot{\phi}_2$) und die Position der Azimut Plattform (ϕ_4). Ferner wurde wegen der schnellen Stromregelung das Soll- und Istmoment der Motorgruppen gleichgesetzt. Die Regelgröße ist die Plattform-Position ϕ_4 . Für das Fünfmassenmodell wurden die Werte aus Tabelle 9.2 für die Trägheitsmomente, die Federsteifigkeiten und den Dämpfungskonstanten ermittelt (Daten der Fa. MAN Technologie).

Die Zustandsgleichungen des Systems lassen sich aus dem Schemabild 9.2 des Radioteleskops herleiten. Die beiden Stellgrößen u_1 und u_2 stellen die Motormomente der beiden

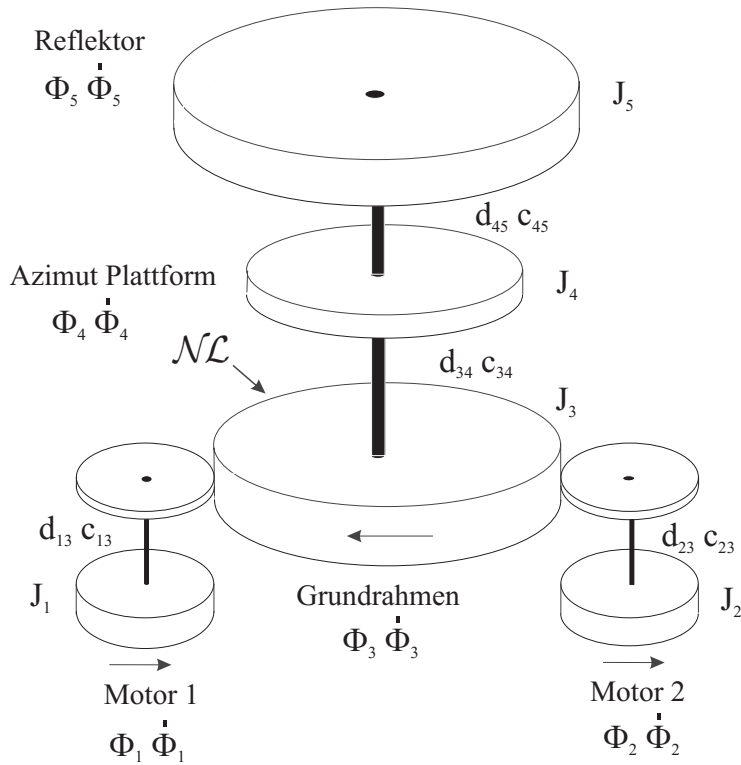


Bild 9.2: Schemabild des Radioteleskops

$J_1 = 8 \cdot 11.5 \cdot 10^6$	$kg \cdot m^2$		
$J_2 = 8 \cdot 11.5 \cdot 10^6$	$kg \cdot m^2$		
$J_3 = 7.04 \cdot 10^7$	$kg \cdot m^2$		
$J_4 = 3.03 \cdot 10^7$	$kg \cdot m^2$		
$J_5 = 20.7 \cdot 10^7$	$kg \cdot m^2$		
$c_{13} = 2 \cdot 8 \cdot 3.5 \cdot 10^6 \cdot (198/5)^2$	Nm/rad	$d_{13} = 0.03 \cdot c_{13}$	$Nm \cdot s/rad$
$c_{23} = 2 \cdot 8 \cdot 3.5 \cdot 10^6 \cdot (198/5)^2$	Nm/rad	$d_{23} = 0.03 \cdot c_{23}$	$Nm \cdot s/rad$
$c_{34} = 198.3 \cdot 10^9$	Nm/rad	$d_{34} = 0.03 \cdot c_{34}$	$Nm \cdot s/rad$
$c_{45} = 2 \cdot 17.6 \cdot 10^9$	Nm/rad	$d_{45} = 0.03 \cdot c_{45}$	$Nm \cdot s/rad$

Tabelle 9.2: Daten des Fünfmassensystems

Motorgruppen dar. Die Umrichterndynamik wurde dabei vernachlässigt.

$$\underbrace{\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \\ \dot{\phi}_5 \\ \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \\ \dot{\phi}_5 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ A_{61} & 0 & A_{63} & 0 & 0 & A_{66} & 0 & A_{68} & 0 & 0 \\ 0 & A_{72} & A_{73} & 0 & 0 & 0 & A_{77} & A_{78} & 0 & 0 \\ A_{81} & A_{82} & A_{83} & A_{84} & 0 & A_{86} & A_{87} & A_{88} & A_{89} & 0 \\ 0 & 0 & A_{93} & A_{94} & A_{95} & 0 & 0 & A_{98} & A_{99} & A_{910} \\ 0 & 0 & 0 & A_{104} & A_{105} & 0 & 0 & 0 & A_{109} & A_{1010} \end{bmatrix}}_{\mathbf{A}} \cdot \underbrace{\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \\ \dot{\phi}_4 \\ \dot{\phi}_5 \end{bmatrix}}_{\mathbf{x}}$$

$$\begin{aligned}
& + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{J_1} & 0 \\ 0 & \frac{1}{J_2} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{\mathbf{b}} \cdot \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}_{\mathbf{u}} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{J_3} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{k}} \cdot \mathcal{N}(\dot{\phi}_3) \quad (9.22)
\end{aligned}$$

mit den Matrixeinträgen:

$$\begin{array}{lll}
A_{61} = -c_{13}/J_1 & A_{63} = c_{13}/J_1 & A_{66} = -d_{13}/J_1 \\
A_{68} = d_{13}/J_1 & & \\
A_{72} = -c_{23}/J_2 & A_{73} = c_{23}/J_2 & A_{77} = -d_{13}/J_2 \\
A_{78} = d_{23}/J_2 & & \\
A_{81} = c_{13}/J_3 & A_{82} = c_{23}/J_3 & A_{83} = -(c_{13} + c_{23} + c_{34})/J_3 \\
A_{84} = c_{34}/J_3 & A_{86} = d_{13}/J_3 & A_{87} = d_{23}/J_3 \\
A_{88} = -(d_{13} + d_{23} + d_{34})/J_3 & A_{89} = d_{34}/J_3 & \\
A_{93} = c_{34}/J_4 & A_{94} = -(c_{34} + c_{45})/J_4 & A_{95} = c_{45}/J_4 \\
A_{98} = d_{34}/J_4 & A_{99} = -(d_{34} + d_{45})/J_4 & A_{910} = d_{45}/J_4 \\
A_{104} = c_{45}/J_5 & A_{105} = -c_{45}/J_5 & A_{109} = d_{45}/J_5 \\
A_{1010} = -d_{45}/J_5 & &
\end{array}$$

Die Ausgangsgleichung ergibt sich zu:

$$y = \phi_4 = [0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0] \cdot \mathbf{x} \quad (9.23)$$

Für den Entwurf einer prädiktiven Regelung wird die in [57] gelernte Reibkennlinie des Radioteleskops aus der Überlagerung einer Geraden- und einer sehr steil verlaufenden Arcustangens-Funktion angenähert. Dies ist für die Differenzierbarkeit und die Näherung erster Ordnung der Nichtlinearität nach Kap. 6.1.2 notwendig. Die genäherte Reibkennlinie lautet:

$$\mathcal{N}(\dot{\phi}_3) = 164525.44 \cdot \arctan(100000 \cdot \dot{\phi}_3) + 4.3 \cdot 10^8 \cdot \dot{\phi}_3 \quad [Nm] \quad (9.24)$$

Das Einwirken von Haftreibung bedeutet einen unstetigen Verlauf der Reibkennlinie bei der Drehzahl null. Dies wird durch Gleichung (9.24) nicht exakt nachgebildet. Der Stick-Slip-Effekt in den Positions- und Drehzahlverläufen ist jedoch nahezu identisch, so dass die durchgeführte Näherung für die Reibkennlinie zulässig ist. Die verwendete Näherung für die Reibkennlinie ist in Bild 9.3 dargestellt.

9.2.2 Gegenüberstellung von prädiktivem Regler und PI-Regler am Radioteleskop

Das Radioteleskop in Effelsberg wird momentan mit einem kaskadierten PI-Regler betrieben. Als Messgrößen stehen die Drehzahlen der beiden Motorgruppen und die Position

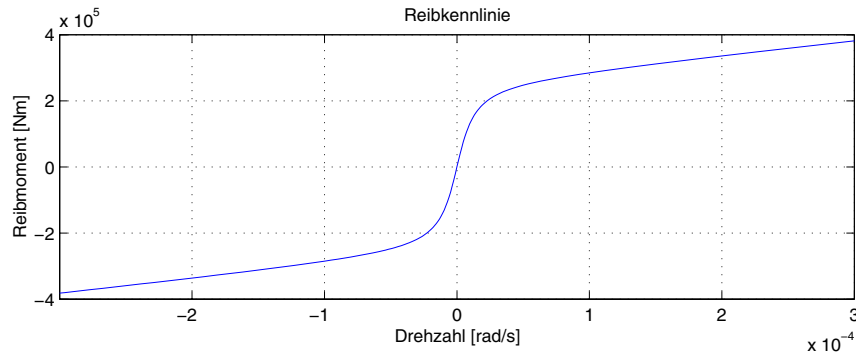


Bild 9.3: *Reibkennlinie am Radioteleskop*

und Drehzahl der Azimut Plattform zur Verfügung. Da das Folgeverhalten sehr wichtig ist, wird neben der Sollposition auch eine globale Solldrehzahl und -beschleunigung durch Differentiation generiert. Der innere Drehzahlregelkreis erzeugt den Sollstrom für die netzgeführten Stellglieder der Gleichstrommaschinen. Als Regelgröße wird der Mittelwert der Drehzahlen der beiden Motorgruppen verwendet. Die Einstellung der Reglerparameter erfolgt nach dem symmetrischen Optimum, wobei das gesamte schwingungsfähige Teleskop durch eine Ersatzzeitkonstante für die Reglerauslegung angenähert wird. Um das Folgeverhalten zu verbessern, erfolgt eine Vorsteuerung des Sollstroms mit Hilfe der Sollbeschleunigung. Um die Schwingungsneigung des Teleskops zu verringern, werden zwei zusätzliche Drehzahldifferenzregler als P-Regler ausgeführt. Die Drehzahldifferenz zwischen den beiden Motorgruppen und zwischen dem Mittelwert der Motorgruppen und der Azimut Plattform wird auf null geregelt. Dies erhöht die Dämpfung der Schwingungsbewegung der einzelnen Körper untereinander.

Der äußere Positionsregelkreis ist ein PI-Regler, der die Solldrehzahl für den inneren Drehzahlregelkreis erzeugt. Um das Folgeverhalten zu optimieren, erfolgt eine Vorsteuerung mit Hilfe der globalen Solldrehzahl des Teleskops. Die Einstellungen des Positionsreglers erfolgten experimentell für möglichst gutes Folgeverhalten bei häufig auftretenden Testtrajektorien. Um Messrauschen und Einstreuungen durch die leistungselektronischen Stellglieder zu unterdrücken, werden alle Messsignale mit einem Butterworth-Tiefpassfilter mit Grenzfrequenz 100 Hz gefiltert. Zur Verringerung der Anregung von Eigenfrequenzen der mechanischen Struktur sind alle Sollwertkanäle mit Anstiegsbegrenzungen ausgestattet. Diese aufwändige Reglerstruktur ist nötig, um die geforderten Genauigkeiten in der Positionierung einzuhalten. Es handelt sich nicht um einen Standard-Regelkreis, vielmehr wurden durch Erweiterungen bereits Optimierungspotentiale ausgeschöpft.

Es ist nicht auszuschließen, dass die verwendete kaskadierte PI-Regelung weitere Verbesserungspotentiale durch veränderte Parametereinstellungen bietet, die beim Geschwindigkeits-Nulldurchgang auftretende Stick-Slip-Bewegung konnte jedoch bisher nicht beseitigt werden. Diese verursacht Abweichungen bei der Positionierung des Reflektors. Aufgrund des generell guten Folgeverhaltens des prädiktiven Reglers liegt es nahe, den PI-Regler des Radioteleskops durch eine prädiktive Regelung zu ersetzen. In diesem Abschnitt soll das Teleskop mit einem PI-Regler, einem PI-Regler mit Störgrößenkompensation und einem prädiktiven Regler simuliert und die Ergebnisse verglichen werden. Die Einstellungen des

PI-Regler wurden gemäß den tatsächlichen Einstellungen in Effelsberg gewählt. Zur Beurteilung des Folgeverhaltens wird das System mit einem Sinusverlauf angeregt.

Bild 9.4 zeigt die Positionstrajektorie r_ϕ und die die Position ϕ der Plattform sowie die Geschwindigkeitstrajektorie r_Ω und die mittlere Geschwindigkeit Ω der beiden Motorgruppen bei konventioneller PI-Regelung. Im unteren linken Bild sind die Verläufe der Positionen vergrößert, und im unteren rechten Bild ist die mittlere Geschwindigkeit der Motorgruppen beim Nulldurchgang vergrößert dargestellt. Es ist deutlich zu erkennen, dass die Stick-Slip-Effekte vom PI-Regler nicht besonders gut beherrscht werden. Beim Nulldurchgang der Geschwindigkeit kommt das Teleskop für eine kurze Zeit zum Stehen, was zu erheblichen Abweichungen zwischen Positionstrajektorie r_ϕ und Regelgröße ϕ führt.

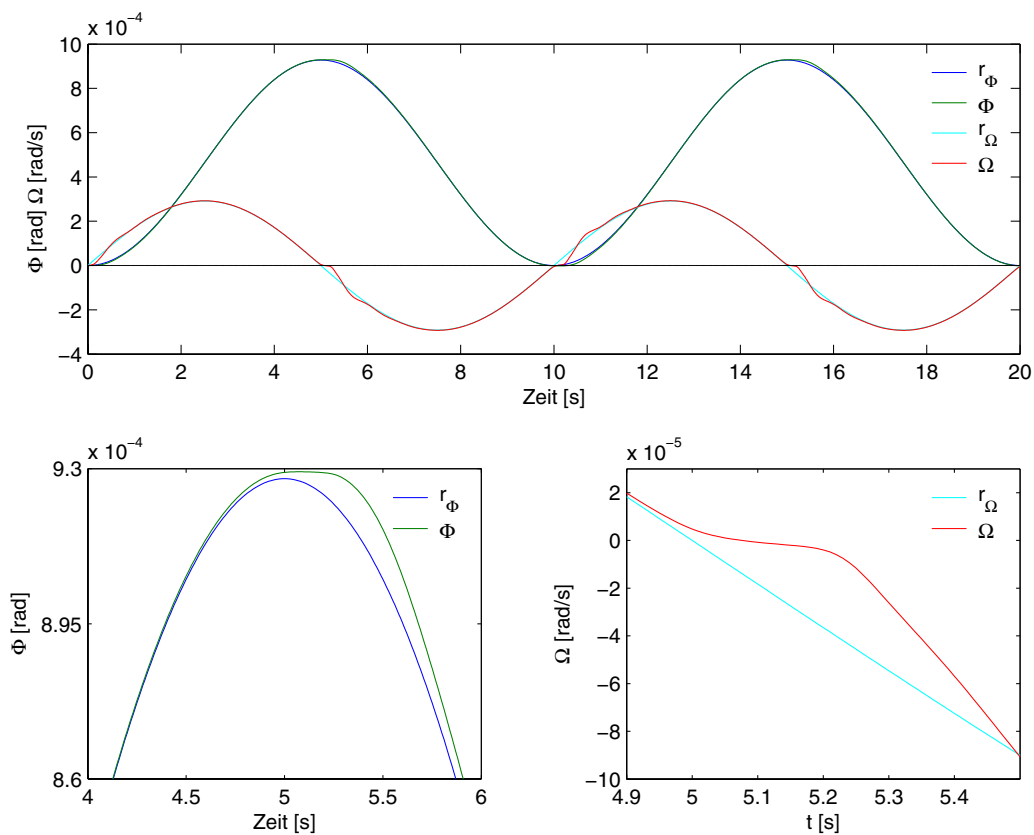


Bild 9.4: *PI-Regler ohne Störgrößenkompensation*

Beim PI-Regler mit Störgrößenkompensation wird der Einfluss der nichtlinearen Reibung mittels einer Kompensationsschaltung reduziert. Das jeweils aktuelle Reibmoment am Grundrahmen wird als zusätzliches Sollmoment auf die beiden Motorgruppen geschaltet (zu gleichen Anteilen). Dieses Verfahren ist jedoch nur stationär genau, nicht aber dynamisch, da die Steifigkeit und Dämpfung zwischen Motor und Grundrahmen nicht berücksichtigt wurden. In Bild 9.5 ist der Lage- und Geschwindigkeitsverlauf des Radioteleskops über der Zeit aufgetragen. Beim Geschwindigkeits-Nulldurchgang kommt es zwischen Sollwerttrajektorie r_ϕ und Ausgangsgröße ϕ im Vergleich zum PI-Regler ohne Störgrößenkompensation zu einer geringeren Abweichung. Aber auch hier ist es nicht möglich, den Regelfehler, bedingt durch den Stick-Slip-Effekt, vollständig zu eliminieren.

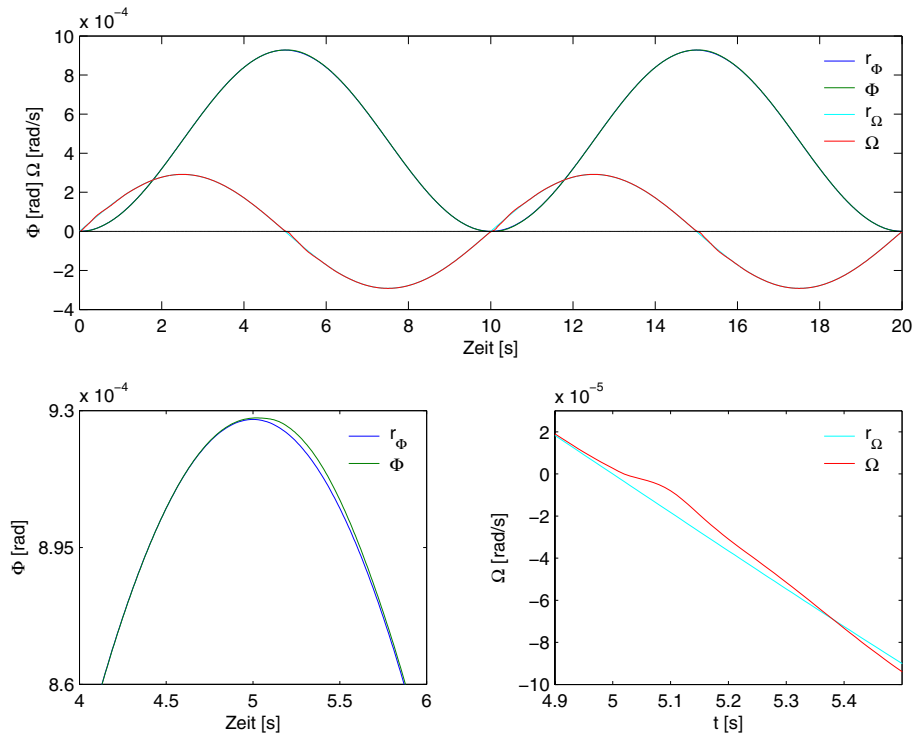


Bild 9.5: *PI-Regler mit Störgrößenkompensation*

Da beim Radioteleskop bereits bei geringen Abweichungen in der Position das Ziel verfehlt werden kann, ist eine Verbesserung der Regelung wünschenswert.

Der prädiktive Regler soll nun dem PI-Regler mit Störgrößenkompensation gegenüber gestellt werden. Zum Reglerentwurf wird die Theorie zur prädiktiven Regelung aus Kap. 9.1 angewendet. Die Ausgangsgröße ist die Position ϕ_4 der Plattform, die Stellgrößen sind die Momente der beiden Motorgruppen M_1 und M_2 . Es wird eine prädiktive Regelung ohne Begrenzungen entworfen, da sich herausgestellt hat, dass die Motormomente noch sehr weit von ihren Grenzwerten entfernt sind. Die Parameter der prädiktiven Regelung lauten:

$$\begin{array}{lll} T_{ab} = 10 \text{ ms} & \lambda_i = 1 \cdot 10^{-25} & \gamma_i = 1 \cdot 10^{-15} \\ N_1 = 3 & N_2 = 35 & N_u = 5 \end{array}$$

Der für die Regelung benötigte Zustandsvektor $\mathbf{x}(k)$ kann durch einen Beobachter gemäß Kap. 3.3 zur Verfügung gestellt werden. Analog zur vorherigen Simulation sind in Bild 9.6 der Geschwindigkeits- und Lageverlauf des Radioteleskops dargestellt. Wird das Radioteleskop mit dem prädiktiven Regler betrieben, kann zwischen Lagetrajektorie r_ϕ und Ausgangsverlauf ϕ keine Abweichung festgestellt werden. Aufgrund der Vorausschau ist es der prädiktiven Regelung möglich, rechtzeitig der Stick-Slip-Bewegung beim Geschwindigkeits-Nulldurchgang entgegenzuwirken, um so eine Regeldifferenz zwischen Trajektorie r_ϕ und Ausgangssignal ϕ zu verhindern. Im rechten unteren Bild ist der Geschwindigkeitsverlauf Ω der Azimut Plattform dargestellt. Aus der Plattform-Geschwindigkeit ist ersichtlich, dass es beim Geschwindigkeits-Nulldurchgang zu einer sehr kleinen Abweichung kommt und der Stick-Slip-Effekt nicht eliminiert werden kann. Der prädiktive Regler weist im Vergleich zum PI-Regler mit Störgrößenkompensation ein wesentlich besseres Folgeverhalten auf.

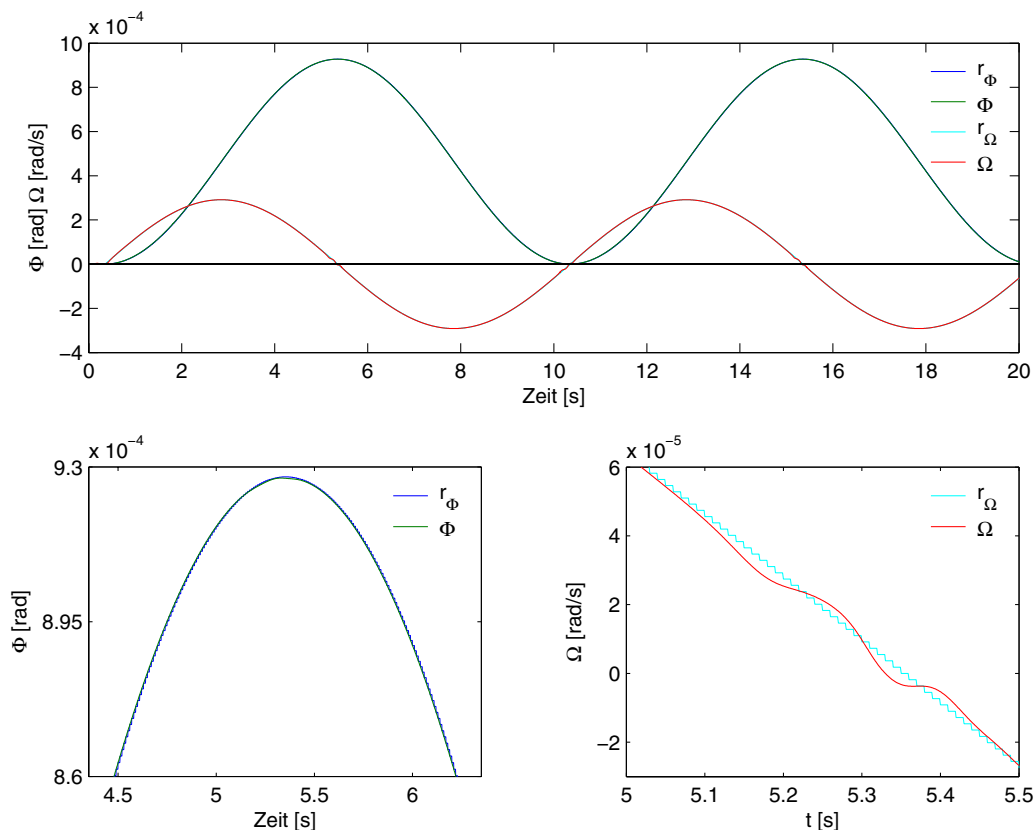


Bild 9.6: Radioteleskop mit prädiktivem Regler

In Bild 9.7 sind die Geschwindigkeitsverläufe beim Nulldurchgang der Geschwindigkeit des Reflektors, des Grundrahmens und der Motorgruppen 1 und 2 dargestellt. Da die Reibung des Radioteleskops am Grundrahmen angreift, müssen sich die Motorgeschwindigkeiten Ω_1 , Ω_2 von der Trajektorie r_Ω unterscheiden, da das Feder-Dämpfer-Element zwischen Motor und Grundrahmen verdreht werden muss, um das Haftmoment zu überwinden. Dadurch ist die genaue Positionierung der Azimut Plattform gewährleistet. Beim Reflektor, dessen Positionierung das eigentliche Regelziel ist, ist die Geschwindigkeitsabweichung beim Nulldurchgang fast nicht mehr wahrnehmbar.

In den Simulationen konnte gezeigt werden, dass der prädiktive Regler im Vergleich zum PI-Regler mit Störgrößenkompensation die Lageregelung weiter verbessern kann. Außerdem handelt es sich um eine typische Anwendung der prädiktiven Regelung, da die Positions- und Geschwindigkeitstrajektorien über einen langen Zeitraum schon im voraus bekannt sind und in das Regelgesetz einfließen können. Typischerweise sind in einer Beobachtungsphase Sollwerte über mehrere Stunden vorab bekannt.

Die prädiktive Regelung konnte erst nach einer Annäherung der unstetigen Haftreibungskennlinie durch eine stetige Funktion realisiert werden. Der Effekt der Stick-Slip-Bewegung wurde dadurch kaum verändert, so dass man nach wie vor von einem sehr guten Systemmodell ausgehen kann. Zur Linearisierung ist die Referenztrajektorie des Grundrahmens nötig. Da sie nicht bekannt ist und nur relativ langsame Bewegungen mit kleinen Verdrehwinkeln relevante Arbeitspunkte darstellen, wurde zur Linearisierung die bekannte Referenztrajek-

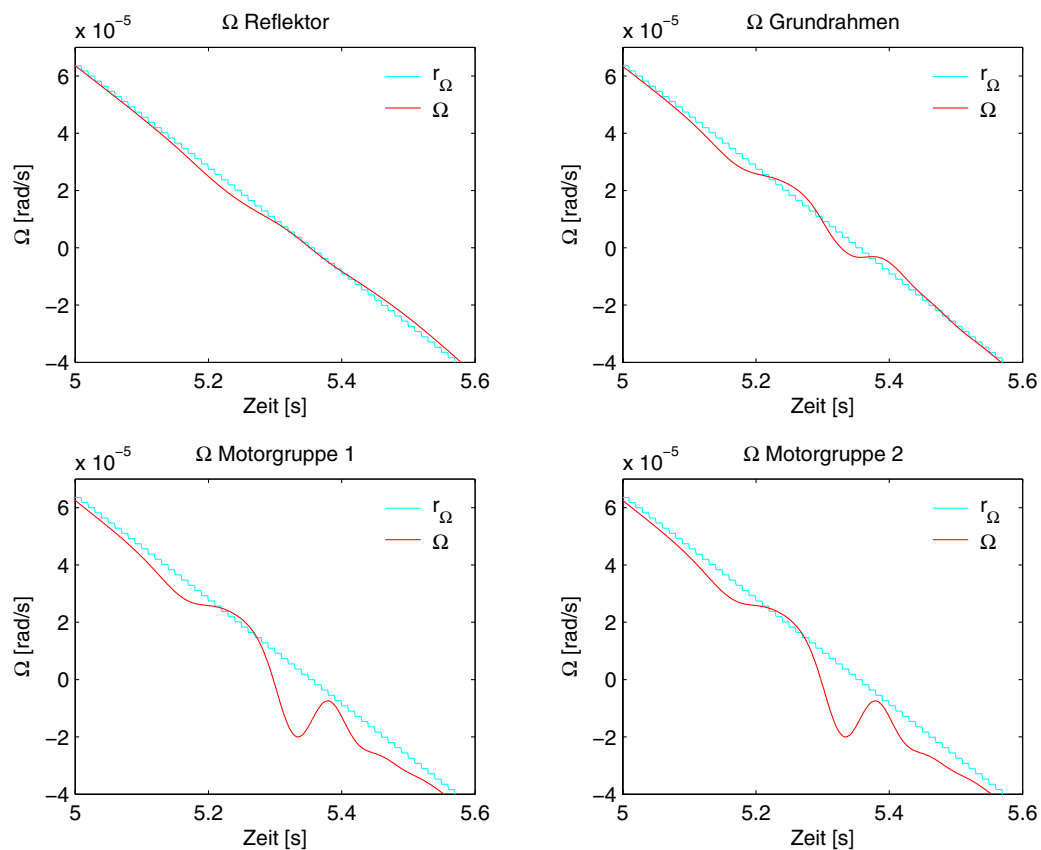


Bild 9.7: Geschwindigkeitsverläufe bei prädiktiver Regelung

torie r_Ω der Azimut Plattform verwendet. Es wurde implizit nahezu Gleichheit der beiden Referenztrajektorien unterstellt, was sich durch das sehr gute Regelergebnis auch bestätigt hat.

10 Zusammenfassung und Ausblick

Als Abschluss der vorliegenden Arbeit werden die wichtigsten Ergebnisse nochmals zusammengefasst, und es wird ein Ausblick auf weitere Einsatzgebiete und zukünftige Forschungsaktivitäten gegeben.

Aufbauend auf bekannten linearen Prozessbeschreibungen wurde ein Identifikationsverfahren für Prozesse, bestehend aus linearem Anteil und isolierter Nichtlinearität, vorgestellt. Nichtlinearitäten werden durch ein *General Regression Neural Network* approximiert und online trainiert. Es konnte gezeigt werden, dass der Lernvorgang garantiert stabil abläuft, unabhängig von der Struktur und Ordnung des Systems. Dazu wurden mehrere Fehlermodelle eingeführt, auf die sich das Identifikationsverfahren zurückführen lässt. Durch ein Entkopplungsfilter war es möglich, mehrere Nichtlinearitäten zu trennen und gleichzeitig zu identifizieren. Um einen effizienten Einsatz des Gradientenverfahrens zum Netzwerktraining zu erreichen, wurde eine optimale, sich selbst einstellende Lernschrittweite entwickelt. Der zusätzliche Informationsgewinn durch die Identifikation erstreckt sich auf die Schätzung der Nichtlinearität, deren Ableitung, wie sie zur prädiktiven Regelung benötigt wird, und Schätzwerte für alle nicht messbaren Systemzustände. Da das gesamte Verfahren onlinefähig ist, handelt es sich um einen lernfähigen Beobachter. Die Online-Identifikation wurde praktisch an einem Prüfstand mit Erfolg erprobt.

Die im Zuge der Identifikation entwickelte Theorie zu lernfähigen neuronalen Netzen kann zur direkten Regelung durch das neuronale Netz eingesetzt werden. Die Regelung einer Anlage alleine durch ein GRNN erscheint schwierig, da die Inbetriebnahmephase schwer handhabbar ist. Daher wurde hier der Weg gegangen, dass das neuronale Netz einen konventionellen Regler (z.B. PI) bei der Kompensation von Nichtlinearitäten unterstützt. Dabei wurde der in der Antriebstechnik häufige Fall periodisch auftretender Nichtlinearitäten untersucht (rotierende Maschinen). Es konnte gezeigt werden, dass die Kompensation periodischer Nichtlinearitäten mittels GRNN garantiert stabil möglich ist. Das GRNN lernt selbsttätig ein optimales Kompensationssignal. Das Verfahren wurde zur Unterdrückung von Unrundheiten in kontinuierlichen Fertigungsanlagen praktisch an einer Modellarbeitsmaschine (MODAM) erprobt. Dadurch sind höhere Produktionsgeschwindigkeiten bei gleichbleibender Produktqualität möglich.

Zur Berücksichtigung allgemeiner (nicht periodischer) Nichtlinearitäten wurde ein nicht-lineares modellbasiertes prädiktives Regelungsverfahren entwickelt. Ausgehend vom bekannten *Generalized Predictive Control* (GPC) Verfahren im Zustandsraum nach [44] wurde die isolierte Nichtlinearität durch Linearisierung um eine Referenztrajektorie in das Prädiktionsmodell integriert. Daraus entstand ein lineares zeitvariantes (LTV) System mit folgenden Vorteilen gegenüber der exakten Berücksichtigung der Nichtlinearität im Prädiktionsmodell:

- Ohne Berücksichtigung von Begrenzungen ist das resultierende Optimierungsproblem analytisch lösbar, was keine iterativen Verfahren nötig macht und einen wichtigen Vorteil beim Online-Einsatz bedeutet.

- Wegen der Linearität des Prädiktionsmodells und des Einsatzes einer quadratischen Gütefunktion, wurde gezeigt, dass genau ein Minimum existiert, welches auch gefunden wird. Dies ist bei nichtlinearen Verfahren nicht ohne weiteres möglich.
- Bei Berücksichtigung von Begrenzungen entsteht ein quadratisches Programm als Optimierungsaufgabe. Dieses kann durch effiziente Verfahren auch bei kleinen Abtastzeiten in Echtzeit gelöst werden.

Als Einschränkungen der Methode der Linearisierung sind zu nennen, dass die Nichtlinearität zwischen zwei Abtastschritten ausreichend genau durch eine lineare Funktion angenähert werden kann, und dass die tatsächlich erreichte Regelgröße in der Nähe der Referenztrajektorie liegen muss. Dies unterstreicht die Bedeutung der Generierung einer für den Prozess realisierbaren Referenztrajektorie. Bei Abhängigkeit der Nichtlinearität von einem internen Systemzustand wurden Methoden erarbeitet, wie zu den zugehörigen Zustandsgrößen Solltrajektorien erzeugt werden können.

Zur Unterdrückung unbekannter Störungen, die in nahezu allen technischen Systemen vorhanden sind, wurde die prädiktive Regelung um einen Integralanteil im Gütefunktional erweitert. Dadurch ergeben sich formal keine Änderungen an den Prädiktionsgleichungen, lediglich die Dimension der Vektoren und Matrizen steigt durch einen zusätzlich eingeführten Zustand. Schließlich wurde noch die Einbeziehung mehrerer Nichtlinearitäten in das Regelgesetz gezeigt.

Die wohl herausragendste Eigenschaft der prädiktiven Regelung stellt die Möglichkeit der Integration von Beschränkungen ins Regelgesetz dar. Im zugehörigen Optimierungsproblem macht sich dies durch die Berücksichtigung von Nebenbedingungen bemerkbar. Selbst bei linearen Prädiktionsmodellen sind dafür numerische Verfahren nötig. Durch das lineare zeitvariante Prädiktionsmodell können zur Lösung effiziente Algorithmen der quadratischen Programmierung, trotz Berücksichtigung der isolierten Nichtlinearität, angewendet werden. Dies stellt einen entscheidenden Vorteil für die Online-Fähigkeit bei schnellen Prozessen, wie elektro-mechanischen Antrieben dar. In dieser Arbeit wurden Stellgrößenänderungs-, Stellgrößen-, Zustandsgrößen- und Ausgangsgrößenbegrenzungen berücksichtigt.

Da Stabilität eine Grundvoraussetzung für jede technisch relevante Regelung darstellt, wurden Stabilitätsaussagen mit dem *Quasi Infinite Horizon* Konzept hergeleitet. Es konnte gezeigt werden, dass Stabilität der entwickelten Regelung dann vorliegt, wenn das LTV-System zu jedem Zeitpunkt steuerbar und beobachtbar ist. Stabilität des unregulierten Systems ist dagegen nicht Voraussetzung.

Zur praktischen Validierung wurde das Regelungskonzept an einem Zweimassensystem-Prüfstand unter Echtzeitbedingungen implementiert. Das Zweimassensystem stellt eine wichtige Teilkomponente größerer Antriebsanordnungen dar. Wegen der immer auftretenden Gleitreibung als Störgröße, wurde auch das Verfahren mit Integralanteil implementiert. Die Berücksichtigung von Beschränkungen bei der Optimierung wurde durch Einsatz des Algorithmus `q1d` aus [86] realisiert. Für alle Regelungsvarianten konnten sehr gute Ergebnisse erzielt werden, wie auch ein Vergleich mit einem Zustands- und einem PI-Regler gezeigt hat. Dadurch wird der zusätzliche Aufwand durch Modellierung und Online-Rechenaufwand gerechtfertigt. Die Untersuchung der Einstellparameter der Regelung ergab, dass die größten Veränderungen durch den oberen Prädiktionshorizont N_2 und

die Gewichtungsfaktoren λ und α hervorgerufen werden; der untere Prädiktionshorizont N_1 und der Stellhorizont N_u haben dagegen minderen Einfluss auf das Regelergebnis.

Um die Möglichkeiten der Identifikation unbekannter Nichtlinearitäten voll ausschöpfen zu können, wurde ein adaptiver prädiktiver Regler implementiert. Der lernfähige Beobachter schätzt Nichtlinearität und nicht messbare Zustände, die von der Regelung verwendet werden. Dadurch ist eine Anwendung auf Systeme mit nicht genau bekannten nichtlinearen Effekten möglich.

Durch die Verwendung der Darstellung des Prozessmodells im Zustandsraum, war eine Erweiterung auf Mehrgrößenprozesse einfach möglich. Die formalen Prädiktionsgleichungen bleiben erhalten, lediglich ihre Dimensionen erhöhen sich. Als Anwendung wurde ein großes Radioteleskop gewählt. Es konnte eine deutliche Verbesserung im Vergleich zur vorhandenen PI-Regelung erzielt werden. Dabei handelt es sich um einen typischen Anwendungsfall für prädiktive Regelungen, da Referenztrajektorien über einen sehr langen Zeitraum im voraus bekannt sind.

Zusammenfassend betrachtet, wurde ein Identifikations- und ein prädiktives Regelungsverfahren für Systeme mit isolierten Nichtlinearitäten entwickelt. Die Identifikation kann entweder offline oder online, gleichzeitig mit der Regelung ablaufen, was zu einem adaptiven Regelungsverfahren führt. Es wurde sowohl die Nichtlinearität als auch Prozessbegrenzungen explizit beim Regelungsentwurf berücksichtigt. Die nötigen Entwurfsschritte stellen ein systematisches Vorgehen dar und sind auf alle Prozesse der vorausgesetzten Systemklasse anwendbar. Praktische Versuche haben gezeigt, dass der zusätzliche Entwurfs- und Online-Rechenaufwand durch ein deutlich verbessertes Regelergebnis belohnt wird. Als typische Anwendungen sind elektrische Antriebe (Druck, Papierherstellung, Werkzeugmaschinen, alle Arten von drehzahl- und lagegeregelten Antrieben), Robotik und natürlich die Wiege der prädiktiven Regelung, die verfahrenstechnische Industrie, zu nennen.

Mit zunehmender Leistung und Kostenabnahme von Mikroprozessoren ist damit zu rechnen, dass selbst Anwendungen mit extrem kleinen Abtastraten (wie z.B. Strom bzw. Momentenregelungen bei elektrischen Antrieben) für prädiktive Regelungen in Frage kommen. Auch lassen sich dadurch komplexere Prozessmodelle und Optimierungsverfahren in Echtzeit abarbeiten. Eine mögliche weitere Richtung zukünftiger Forschung stellt die Entwicklung online-fähiger nichtlinearer Optimierungsalgorithmen dar. Vor allem die schnelle Konvergenz und eindeutige Lösung muss im Vordergrund stehen. Dadurch entfielen die Notwendigkeit einer Linearisierung und eine genauere Übereinstimmung zwischen Prozess und Modell wäre zu erwarten. Eine weitere wichtige Frage stellt die Überprüfung der Lösbarkeit des Optimierungsproblems zu jedem Zeitschritt dar. Dadurch wird immer eine optimale Stellgröße gewährleistet. Vor allem in sicherheitskritischen Anwendungen (z.B. in der Fahrwerksregelung oder Fahrzeugführung eines KFZ) besitzt diese Forderung eine nicht zu unterschätzende Wichtigkeit. Die Tatsache, dass prädiktive Regelungsverfahren selbst ohne Stabilitätsbeweise seit vielen Jahren zum Einsatz kommen, liegt an der intuitiven Vorgehensweise und der Robustheit des Verfahrens. Gerade die wissenschaftliche Aufarbeitung von Robustheitsuntersuchungen stehen jedoch noch am Anfang ihrer Entwicklung [43].

A Prüfstand Zweimassensystem

Die entwickelten prädiktiven Regelalgorithmen werden an einer antriebstechnischen Versuchsanordnung praktisch erprobt. Dazu dient ein Zweimassensystem, bestehend aus zwei gekoppelten Synchronmaschinen, die beide getrennt angesteuert werden können. Diese universelle Anordnung macht es möglich, eine Maschine als Antrieb einzusetzen und die andere als Last. Durch die individuelle Ansteuerung lassen sich beliebige nichtlineare Effekte nachbilden.

Der eingesetzte Prüfstand wird nun in seinen wesentlichen Komponenten beschrieben. Zunächst wird auf die elektrische Verschaltung eingegangen, bevor der mechanische Teil sowohl messtechnisch als auch durch Identifikationsverfahren parametrisiert wird.

A.1 Prüfstand: Elektrische Daten

Bild A.1 zeigt eine Übersicht der elektrischen Konfiguration des Prüfstandes. Das System 1,

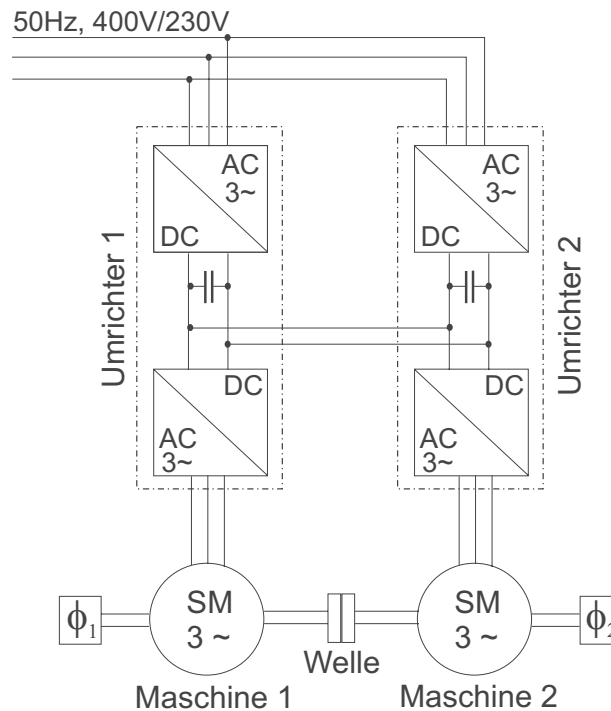


Bild A.1: Übersicht zur elektrischen Konfiguration des Prüfstandes

bestehend aus Umrichter 1 und Maschine 1, wird als Antriebseinheit eingesetzt, während System 2, bestehend aus Umrichter 2 und Maschine 2, zur Aufschaltung von nichtlinearen Einflüssen und Störungen dient.

Die beiden baugleichen Spannungszwischenkreis-Umrichter der Firma Siemens vom Typ

Simovert Master Drives SC 6SE7022-6EC30 steuern zwei permanenterregte Synchronmaschinen an. Die technischen Daten der Umrichter sind in Tabelle A.1 zusammengefasst. Im

Netzspannung	3 AC 380 V bis 460 V ($\pm 15\%$)
Zwischenkreisspannung	DC 510V bis 620 V ($\pm 15\%$)
Ausgangsspannung	3 AC 0 V bis $0.86 \times$ Netzspannung
Netzfrequenz	50 / 60 Hz ($\pm 15\%$)
Ausgangsfrequenz	0 Hz bis 400 Hz
Pulsfrequenz	5 kHz bis 7.5 kHz
Ausgangsbemessungsstrom	25.2 A
Grundlaststrom	23.2 A
Kurzzeitstrom	40.8 A
Verlustleistung	0.43 kW (bei 5 kHz)
Wirkungsgrad	96 bis 98 %

Tabelle A.1: Technische Daten der Umrichter

Umrichter ist eine feldorientierte Drehmomentenregelung integriert, mit der über den Sollwertkanal des Umrichters das Luftspaltdrehmoment der zugehörigen Maschine vorgegeben wird. Aus diesem Grund reduziert sich das Übertragungsverhalten der Synchronmaschine auf ihre Mechanik und eine Ersatzzeitkonstante des Momentenregelkreises.

Bei den beiden Maschinen handelt es sich um permanenterregte Synchronmaschinen der Firma Siemens vom Typ 1FT6068-8AC71-1CD3 mit integrierten Inkremental-Encodern zur Rotorlageerfassung. Die technischen Daten der Maschinen können aus Tabelle A.2 entnommen werden.

Nenndrehzahl	2000 min^{-1}
Nenndrehmoment	22 Nm
Nennstrom	10.9 A
Nennleistung	4.8 kW
Trägheitsmoment	$6.65 \cdot 10^{-3} \text{ kg m}^2$

Tabelle A.2: Technische Daten der Synchronmaschinen

Die Ansteuerung der Umrichter bzw. das Einlesen der Messdaten erfolgt mit einem Standard PC, der mit einem Intel Celeron Prozessor, 1200 MHz, 128 MB Hauptspeicher und dem Echtzeitbetriebssystem *xPC-Target V 1.5* von Mathworks ausgestattet ist. Die Analogausgabe wird mit dem 12 Bit DA-Wandler *DAS-1602* von Keithley Instruments realisiert. Die Auswertung der Positionsgeber erfolgt mit der Interpolationskarte *Heidenhain IK121V*. Als Messsignale stehen die Rotorlagen ϕ_1 und ϕ_2 jeweils in *rad* sowie die Winkelgeschwindigkeiten Ω_1 und Ω_2 jeweils in *rad/s* der Maschinen zur Verfügung. Die Winkelgeschwindigkeiten werden durch softwareseitiges Differenzieren aus den Rotorlagen gewonnen. In Bild A.2 sind die beiden Synchronmaschinen mit ihren aufgeflanschten Schwungscheiben dargestellt. Sie werden mechanisch über einen elastischen Torsionsstab gekoppelt.

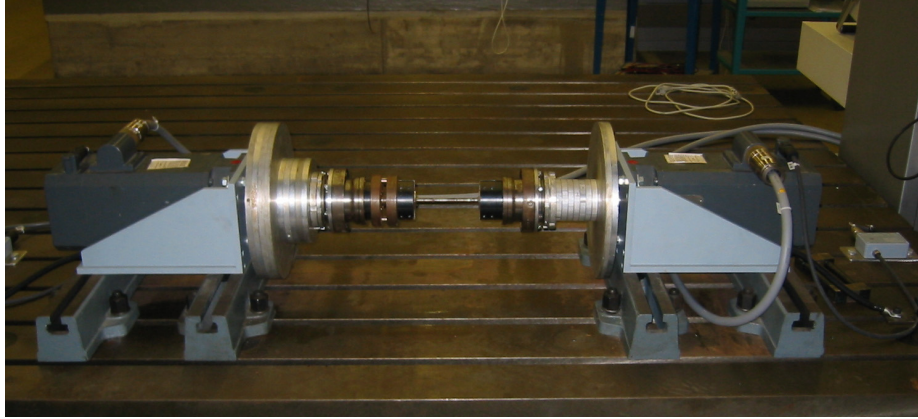


Bild A.2: *Versuchsstand des Zweimassensystems*

A.2 Anlagendaten

Die Massenträgheitsmomente J_1 und J_2 der beiden Maschinen ergeben sich aus der Summation der in Tabelle A.3 aufgelisteten, starr miteinander verbundenen Elementen. Die einzelnen Werte wurden in [51] berechnet.

	Maschine 1	Maschine 2
Rotor	$6.650 \cdot 10^{-3}$	$6.650 \cdot 10^{-3}$
Flansch	$3.626 \cdot 10^{-3}$	$3.626 \cdot 10^{-3}$
Schwungscheibe (groß)	$2 \cdot 68.644 \cdot 10^{-3}$	$4 \cdot 68.644 \cdot 10^{-3}$
Schwungscheibe (klein)	—	$5 \cdot 6.0301 \cdot 10^{-3}$
Distanzscheibe	$7 \cdot 0.082 \cdot 10^{-3}$	—
Kupplungskonstruktion	$17.800 \cdot 10^{-3}$	$17.800 \cdot 10^{-3}$
Gesamt	$165.938 \cdot 10^{-3}$	$332.882 \cdot 10^{-3}$

Tabelle A.3: *Massenträgheitsmomente von Maschine 1 und 2 (in kg m^2)*

A.3 Messtechnische Bestimmung der Reibkennlinie

Die Bestimmung der Reibkennlinie ist zum einen notwendig, um in der Simulation das Modell genauer an die Realität anzupassen und zum anderen wird die Reibkennlinie zur Identifikation der Dämpfungskonstanten d , der Federsteifigkeit c und der Ersatzzeitkonstanten der Umrichter $T_{Umr1/2}$ mit der *Identification-Toolbox* aus Matlab benötigt. Die Reibkennlinie wirkt bei der Identifikation dann wie eine bekannte Störgröße.

Die Mechanik einer leerlaufenden elektrischen Maschine, auf die nur ein drehzahlabhängiges Reibmoment $M_R(\Omega)$ wirkt, lässt sich mit einer Bewegungsdifferentialgleichung beschreiben.

$$\dot{\Omega} = \frac{1}{J} (M - M_R(\Omega)) \quad (\text{A.1})$$

M ist das Maschinenmoment und M_R das Reibmoment. Es wird experimentell bestimmt, indem die Motoren auf ihre maximale Drehzahl Ω_{max} beschleunigt werden. Dann wird das Motormoment M auf null geschaltet, so dass nur noch Reibmomente (Lagerreibung, Haftreibung, Luftreibung) an der Maschine angreifen. Man lässt die Maschinen nun bis zum Stillstand auslaufen, wobei die Winkelgeschwindigkeit Ω und die Winkelbeschleunigung $\dot{\Omega}$ für jeden Abtastschritt aufgezeichnet werden. Die Daten werden für die positive und die negative Drehrichtung aufgenommen. Mit dem Zusammenhang

$$M_R = -J \cdot \dot{\Omega} \quad (\text{A.2})$$

lässt sich das Reibmoment berechnen. Wie aus Bild A.3 ersichtlich ist, können die Reibkennlinien der einzelnen Motoren dargestellt werden, indem auf der vertikalen Achse das Moment M_R und auf der horizontalen Achse die zugehörige Winkelgeschwindigkeit Ω aufgetragen wird. Die vier Äste der Reibkennlinien können nun durch mathematische Funk-

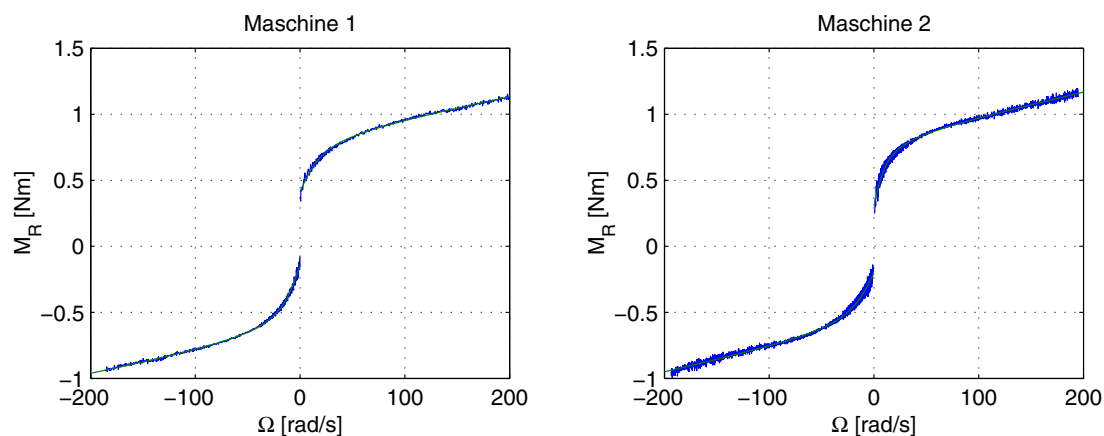


Bild A.3: Reibkennlinien für beide Maschinen

tionen ersetzt werden, wobei die Reibkennlinie von Maschine 1 und Maschine 2 aufgrund der äußerst geringen Unterschiede gleich gesetzt werden. Für die positiven und negativen Äste der Reibkennlinie ergibt sich:

$$M_{Rpos}(\Omega) = 0.3029 + 0.3293 \cdot \arctan(0.1048 \cdot \Omega) + 0.0018 \cdot \Omega \quad [Nm] \quad (\text{A.3})$$

$$M_{Rneg}(\Omega) = -0.2025 + 0.2735 \cdot \arctan(0.0532 \cdot \Omega) + 0.0017 \cdot \Omega \quad [Nm] \quad (\text{A.4})$$

Diese Reibkennlinien werden zur genauen Abstimmung zwischen Simulationsmodell und Realität in das Simulationsmodell eingefügt.

A.4 Lineare Parameteridentifikation

Die Dynamik der beiden Umrichter wird als PT_1 -Glied modelliert. Die elastische Verbindung beider Motoren stellt ein Feder-Dämpfer-Element dar. Zur Bestimmung der Zeitkonstanten der Umrichter $T_{Umr1/2}$, der Dämpfung d und der Federsteifigkeit c wird das Zweimassensystem mit einem PI-Regler auf konstanter Drehzahl gehalten, damit sich während

der Identifikation keine Haftreibungseinflüsse bemerkbar machen. Zusätzlich wird dem Ausgangssignal des Reglers ein Rauschsignal zur Anregung überlagert. In Bild A.4 ist der für die Identifikation verwendete Signalflussplan dargestellt. Der Hardwarezugriff erfolgt über das Echtzeitsystem *xPC-Target*. Als Messsignale werden M_{Soll} , M_{Reib1} , M_{Reib2} , Ω_1 und Ω_2 aufgezeichnet. Die beiden Reibmomente sind aus den Auslaufversuchen in Anhang A.3 bekannt. Der selbe Versuch wird mit einer Anregung von Motor 2 wiederholt.

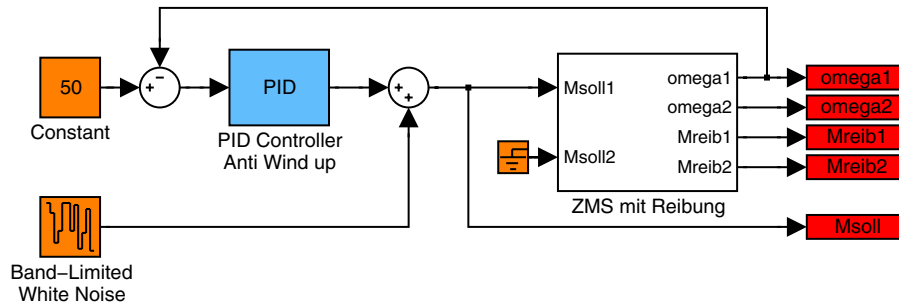


Bild A.4: Identifikation des Zweimassensystems

Mit der *System Identification Toolbox* aus Matlab können unbekannte Parameter in der Zustandsdarstellung identifiziert werden [49]. Da für das System die Reibkennlinie sowie die Trägheitsmomente J_1 und J_2 bereits bekannt sind, kann mit dem Matlab Kommando `pem` die Identifikation der fehlenden Parameter durchgeführt werden. Der Vorteil von `pem` gegenüber anderen Algorithmen ist, dass bekannte Parameter vorab festgelegt werden können. Die Parameter werden nach einem Least-Square-Verfahren geschätzt (Kap. 2.2.4). Der Identifikationsvorgang wird 10 mal wiederholt und zur endgültigen Parameterbestimmung wird der Mittelwert aus den einzelnen Versuchsreihen verwendet. Dies sorgt für eine zusätzliche Ausmittelung von Messfehlern. Für die Federsteifigkeit c , die Dämpfung d und die Ersatzzeitkonstanten der beiden Umrichter $T_{Umr1/2}$ wurden folgende Werte ermittelt:

$$d = 0.25 [Nm s/rad] \quad c = 1128 [Nm/rad] \quad T_{Umr1/2} = 2.0 [ms] \quad (A.5)$$

A.5 Systembeschreibung im Zustandsraum

In den Simulationen und praktischen Versuchen zur prädiktiven Regelung hat sich gezeigt, dass sich im Regelergebnis kein Unterschied zeigt, wenn die Umrichterdynamik im Prozessmodell nicht berücksichtigt wird. Aus diesem Grund kann das Modell um die Umrichterdynamik reduziert werden, ohne an Genauigkeit zu verlieren.

Des weiteren wird die natürlich vorhandene Reibung nicht in das Prozessmodell integriert, sondern wirkt als Störgröße. Dadurch kann die Funktion des prädiktiven Reglers mit Integralanteil nach Kap. 6.4 überprüft werden.

Das Prozessmodell beinhaltet demnach die beiden rotierenden Maschinen und ihre elastische Kopplung. Maschine 1 wird als Antrieb verwendet, Maschine 2 als Lastmaschine. Um der Anordnung nichtlineares Verhalten einzuprägen, wird Maschine 2 mit einer nichtlinearen Kennlinie angesteuert. Diese stellt eine extrem stark ausgeprägte Reibkennlinie dar

und ist durch Gleichung (A.6) gegeben.

$$M_2(\Omega_2) = -\mathcal{NL}(\Omega_2) = -6.4 [Nm] \cdot \arctan(10 [s/rad] \cdot \Omega_2) \quad (\text{A.6})$$

Der Signalflussplan des Prozessmodells für die prädiktive Regelung ist in Bild A.5 dargestellt [29]. Die Regelgröße bzw. Ausgangsgröße ist die Drehzahl Ω_2 von Maschine 2. Da die

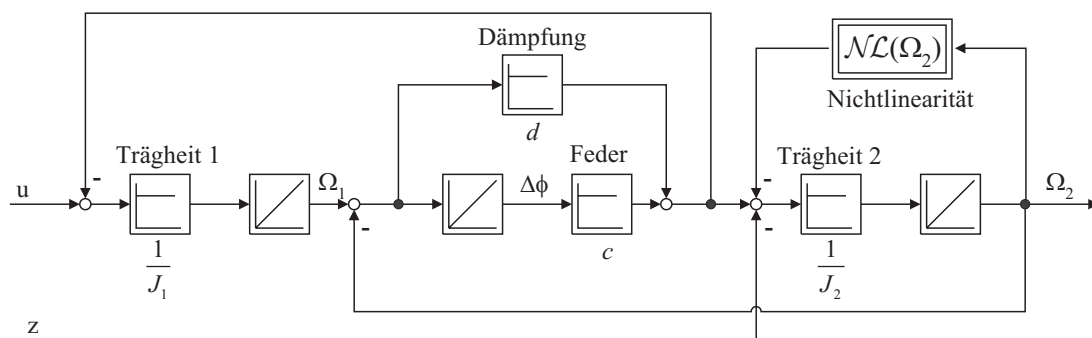


Bild A.5: Signalflussplan des Zweimassensystems mit Nichtlinearität

Nichtlinearität von Ω_2 abhängt, handelt es sich hier um den Fall einer Nichtlinearität, die abhängig vom Prozessausgang ist (siehe Kap. 6.1). Wegen der Messbarkeit von ϕ_1 , ϕ_2 , Ω_1 und Ω_2 liegt ferner vollständige Zustandsmessbarkeit vor, wodurch ein Zustandsbeobachter nicht notwendig ist. Zur Demonstration der Funktionsfähigkeit kommt dieser trotzdem zum Einsatz. Die Stellgröße u ist das Motormoment M_1 von Maschine 1. Über den Eingang z können zusätzliche Störgrößen in das System eingebracht werden. Das Sollmoment für Maschine 2 ergibt sich dann zu:

$$M_2 = -\mathcal{NL}(\Omega_2) - z \quad (\text{A.7})$$

Die Zustandsgleichungen des Zweimassensystems lassen sich aus dem Signalflussplan herleiten. Die verwendeten Parameter sind in Tabelle A.4 zusammengefasst.

J_1	=	0.166	$kg\,m^2$
J_2	=	0.333	$kg\,m^2$
c	=	1128	Nm/rad
d	=	0.25	$Nm\,s/rad$

Tabelle A.4: Parameter des Zweimassensystems

$$\underbrace{\begin{bmatrix} \dot{\Omega}_1 \\ \Delta\dot{\phi} \\ \dot{\Omega}_2 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} -\frac{d}{J_1} & -\frac{c}{J_1} & \frac{d}{J_1} \\ 1 & 0 & -1 \\ \frac{d}{J_2} & \frac{c}{J_2} & -\frac{d}{J_2} \end{bmatrix}}_{\mathbf{A}} \cdot \underbrace{\begin{bmatrix} \Omega_1 \\ \Delta\phi \\ \Omega_2 \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{b}} \cdot u \quad (\text{A.8})$$

$$+ \underbrace{\begin{bmatrix} 0 \\ 0 \\ -\frac{1}{J_2} \end{bmatrix}}_{\mathbf{k}} \cdot \mathcal{NL}(\Omega_2) + \underbrace{\begin{bmatrix} 0 \\ 0 \\ -\frac{1}{J_2} \end{bmatrix}}_{\mathbf{k}_z} \cdot z$$

$$y = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{\mathbf{c}^T} \cdot \mathbf{x} \quad (\text{A.9})$$

Die durchgeführten Messreihen finden alle auf dem hier beschriebenen Zweimassensystem statt. Die Regelung wurde auf der Anlage unter Echtzeitbedingung mittels *xPC-Target* realisiert. Dazu muss die numerische Lösung der quadratischen Optimierungsprobleme in der Programmiersprache C realisiert werden, wenn Nebenbedingungen berücksichtigt werden sollen. Dieser Algorithmus wurde von Prof. Schittkowski zur Verfügung gestellt [86] und als S-Funktion in das Simulink Modell eingebunden.

B Daten und Aufbau der MODAM

Dieser Abschnitt beinhaltet die der Arbeit zugrunde gelegten technischen Daten der Modelarbeitsmaschine (MODAM), wobei vor allem die Parameter des Papiers sich bei Verwendung einer neuen Papierrolle verändern können. Im Folgenden werden allgemeine Daten, Daten zum Bahnsystem und Papier, zum Wickler-Antriebssystem, zum angenommenen Zweimassenschwinger und zum Echtzeitsystem dSpace unterschieden. Bild B.1 zeigt die Zuordnung der Parameter und Variablen bezüglich der Papierbahn.

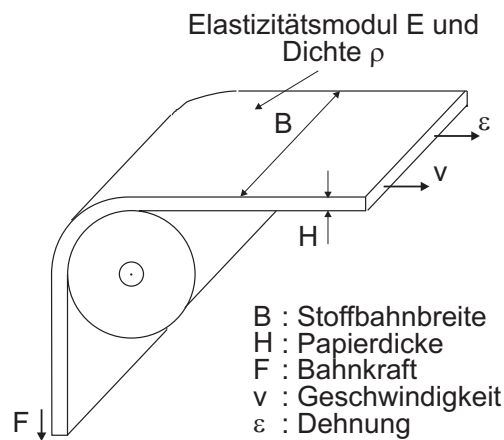


Bild B.1: Papierbahn

B.1 Allgemeine Daten

Gesamtlänge:	$9m$
maximale Transportgeschwindigkeit:	$v_{max} = 240 \text{ m/min} = 4 \text{ m/s}$
maximaler Bahnzug:	$F_{max} = 1000 \text{ N}$
gesamte Motorleistung:	59 kW

B.2 Daten zum Bahnsystem und Papier

Breite:	$B = 0,5 \text{ m}$
Dicke:	$H = 6,0 \cdot 10^{-5} \text{ m}$
Elastizitätsmodul Papier allgemein:	$E = 0,4 - 14 \cdot 10^9 \text{ N/m}^2$
Elastizitätsmodul verwendetes Papier:	$E = 8,5 \cdot 10^9 \text{ N/m}^2$
Im Wickler gespeicherte Dehnung ($F = 150 \text{ N}$):	$\epsilon_{01} = 0.00057$
Papierdichte:	$\rho = 1170 \text{ kg/m}^3$
Flächengewicht des Papiers:	$m_A = 70 \text{ g/m}^2$

Maximale Dehnung:	$\epsilon_{max} \approx 3.9^{-3}$
Länge der freien Stoffbahn zwischen Wickler und Leitantrieb:	$L_n(R) = 2,93\,m \dots 3,04\,m$

B.3 Daten des Wickler-Antriebssystems

Motornennleistung:	$P_N = 22\,kW$
Motornennstrom:	$I_N = 51,5\,A$
Motornennspannung:	$U_N = 340\,V$
Motornennmoment:	$M_{iN} = 140\,Nm$
Motornenndrehzahl:	$N_{0N} = 1500\,1/min = 25\,1/s$
max. Motordrehzahl:	$N_{max} = 3056\,1/min = 50,93\,1/s$
Motornennfrequenz:	$f_N = 50\,Hz$
Ersatzzeitkonstante Momentenregelung:	$T_i = 3\,ms$
Getriebeübersetzung:	$\ddot{u} = 4$
max. Wicklerradius:	$R_{max} = 0,5\,m$
min. Wicklerradius:	$R_{min} = 0,05\,m$

B.4 Daten des Zweimassenschwingers

Trägheitsmoment Motor:	$J_1 = 0.13\,kg \cdot m^2$
Federsteifigkeit des Zweimassensystems:	$c_f = 1250\,Nm/rad$
Dämpfungskonstante des Zweimassensystems:	$d_f = 0.02\,Nm \cdot s/rad$
Getriebeübersetzung:	$\ddot{u} = 4$
Trägheitsmoment Wickler (motorbezogen):	$J_2 = f(R_m) = 0.1\,kgm^2 \dots 3.589\,kgm^2$

B.5 Daten des Regelsystems dSpace

Anzahl der Prozessoren:	2
Taktrate (Master):	40 MHz
Taktrate (Slave):	500 MHz
Bitanzahl (Master):	32 Bit
Bitanzahl (Slave):	64 Bit
Abtastzeit:	1.25 ms

B.6 Teilkomponenten des Versuchsstandes

In Bild B.2 ist der Aufbau der gesamten Anlage und das Zusammenspiel der einzelnen Komponenten dargestellt. Diese sollen im Folgenden jeweils kurz beschrieben werden.

Regel- und Steuersystem SIMADYN-D

SIMADYN D ist ein digitales, frei projektierbares und modular aufgebautes Regelungssystem, das in erster Linie für die Antriebstechnik konzipiert wurde. Mit Hilfe der großen Anzahl von Logikbausteinen können auch größere Steuerungen entworfen werden. Im einzelnen beinhaltet dies zyklische Steuer- und Regelungsaufgaben, alarmgesteuerte Steuerungsaufgaben, arithmetische Berechnungen mit Gleitkommazahlen sowie Protokoll- und Kommunikationsaufgaben.

SIMADYN D wird über die grafische Sprache STRUC G programmiert, die compilierten Programme werden auf EPROM-Karten geladen und den Prozessoren zugeführt. SIMADYN D hat in dieser Anlage lediglich Überwachungs- und Freigabefunktion (Nothalt, Reglerfreigabe, Gruppen- Einzelbetrieb, Walzen heben und senken), alle Regelalgorithmen werden auf dem Echtzeitsystem dSpace ausgeführt. Die Bedienung von SIMADYN D erfolgt über das Bediensystem COROS LS-B.

Echtzeitsystem dSpace

Alle Regelalgorithmen laufen auf dem Echtzeitsystem dSpace ab. Die Motordrehzahlen und Bahnkräfte werden über Eingabekarten gemessen, die Sollmomente werden an den Umrichter SIMODRIVE als Analogsignale ausgegeben. dSpace erhält von SIMADYN D die Freigabesignale für die einzelnen Regler.

Das dSpace-System besteht aus einem Master Prozessor (TI-C40, 32 Bit, 40 MHz), der die gesamte Ein- Ausgabe steuert und einem Slave Prozessor (DEC Alpha, 64 Bit, 500 MHz), auf dem alle sonstigen Algorithmen ablaufen. Zusätzlich sind AD-Wandlerkarten zum Einlesen der Bahnkräfte, DA-Wandlerkarten zur Ausgabe der Sollmomente, digitale Eingabekarten zum Einlesen der Freigabesignale aus SIMADYN D und Encoder-Auswertekarten zum Abfragen der Inkrementalgeber (Positionsmessung) der Asynchronmaschinen vorhanden. Die Prozessoren und Peripheriekarten kommunizieren über ein schnelles Bussystem (PHS-Bus).

Das Echtzeitsystem dSpace wird über den Host-PC programmiert und bedient und beobachtet. Die Regelalgorithmen können in Simulink entwickelt werden und mittels des Real-Time-Workshop und eines C-Compilers in lauffähigen Code übersetzt werden. Das lauffähige Programm wird anschließend über einen ISA-Bus Extender auf die beiden Prozessoren von dSpace geladen.

Die Bedienung und Beobachtung des dSpace Echtzeitsystems erfolgt über die auf dem Host-PC ablaufende Software ControlDesk. Damit lassen sich alle Sollwerte und Parameter verändern sowie Messwerte grafisch darstellen und auf der Festplatte zur Weiterverarbeitung (z.B. in Matlab) speichern.

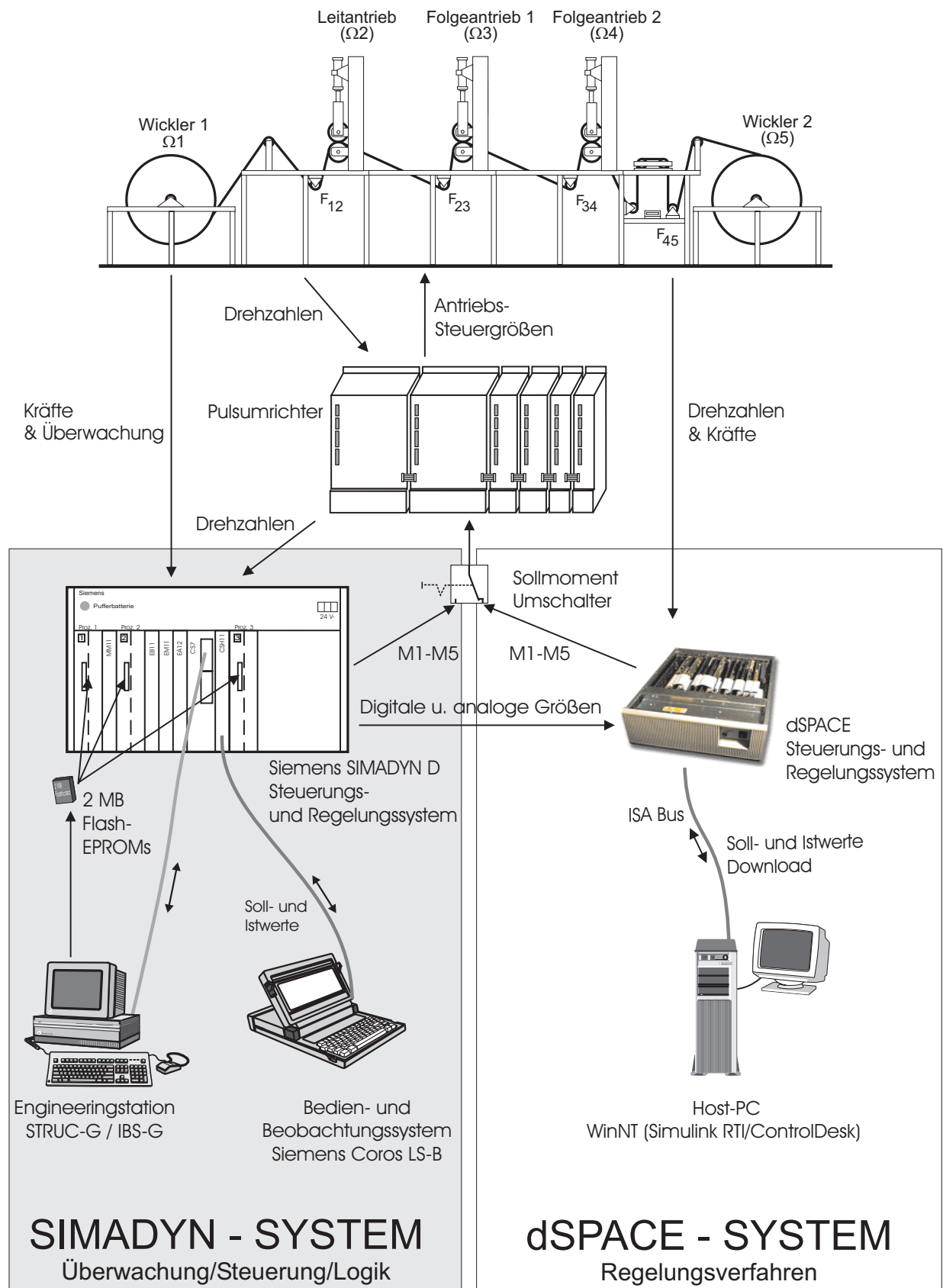


Bild B.2: Komponenten der Gesamtanlage

Bezeichnungen

Typografische Kennzeichnungen

Vektoren	Kleinbuchstaben in Fettschrift	z.B. $\mathbf{x}$
Matrizen	Großbuchstaben in Fettschrift	z.B. $\mathbf{A}$
nichtlineare Funktionen	kalligrafische Großbuchstaben	z.B. $\mathcal{N}\mathcal{L}(y)$
geschätzte Größen	versehen mit $\hat{}$	z.B. $\hat{\mathbf{x}}$
prädizierte Größen	versehen mit $\tilde{}$	z.B. $\tilde{y}$
Arbeitspunktgrößen	versehen mit $_0$	z.B. $\mathbf{x}_0$
Übertragungsfunktionen	Großbuchstaben	z.B. $H(s)$
Änderung einer Größe	vorangestelltes Δ	z.B. Δu

Abkürzungen

BO	Betragsoptimum
CARIMA	Controlled Auto-Regressive Integrated and Moving Average
DMC	Dynamic Matrix Control
DO	Dämpfungsoptimum
FIR	Finite Impulse Response
GNB	Gleichungsnebenbedingung
GPC	Generalized Predictive Control
GRNN	General Regression Neural Network
IIR	Infinite Impulse Response
LDMC	Linear Dynamic Matrix Control
LQI	Linear Quadratisch mit Integralanteil
LS-Verfahren	Least-Squares-Verfahren
LTi	Linear Time Invariant
LTV	Linear Time Variant
MIMO	Multiple-Input/Multiple-Output
MISO	Multiple-Input/Single-Output
MKQ	Methode der kleinsten Quadrate
MLP	Multi Layer Perceptron
MODAM	Modellarbeitsmaschine
MPC	Model Predictive Control
NB	Nebenbedingung
OE-Modell	Output-Error-Modell
QDMC	Quadratic Programming Dynamic Matrix Control
RBF	Radial Basis Function
SISO	Single-Input/Single-Output
SM	Synchronmaschine

SPR	Streng Positiv Reell (Strictly Positive Real)
SO	symmetrisches Optimum
UNB	Ungleichungsnebenbedingung
ZMS	Zweimassensystem

Formelzeichen

α	Gewichtungsfaktor für den aufsummierten Fehler e_s
ϵ	erweitertes Fehlersignal
ϵ_{01}	im Wickel gespeicherte Dehnung
ϵ_{12}	Dehnung des Papiers zwischen Antrieb 1 und 2
ϵ_n	Nenndehnung des Papiers
ζ	verzögerte Aktivierung beim GRNN
η	Lernschrittweite
Θ	konstanter Stützwertevektor der Nichtlinearität
$\hat{\Theta}$	Stützwertevektor des GRNN
$\tilde{\Theta}$	virtueller Stützwertevektor des GRNN
λ	Gewichtungsfaktor für Δu
ρ	Dichte von Papier
σ	Glättungsfaktor des GRNN
σ_{norm}	normierter Glättungsfaktor des GRNN
ϕ	Winkelposition
$\Delta\phi$	Verdrehwinkel am Zweimassensystem
Φ	Parameterfehlervektor
χ	Stützstellen des GRNN
ω_0	Grundkreisfrequenz periodischer Größen
Ω	Winkelgeschwindigkeit
Ω_1	Winkelgeschwindigkeit des Motors
Ω_2	Winkelgeschwindigkeit der Last
$\mathcal{A}$	Aktivierungsvektor
$\mathcal{A}_i$	einzelne Aktivierungsfunktion
$\mathbf{A}$	Systemmatrix
a	Antwortlänge eines FIR-Modells
$A(z^{-1})$	Polynom in z^{-1}
a_i	Polynomkoeffizienten
$\mathbf{b}, \mathbf{B}$	Einkoppelvektor, Einkoppelmatrix
B	Breite der Papierbahn
$\mathcal{B}$	lokale Basisfunktion
$B(z^{-1})$	Polynom in z^{-1}
b_i	Polynomkoeffizienten
$\mathbf{c}, \mathbf{C}$	Auskoppelvektor, Auskoppelmatrix
c, c_f	Federsteifigkeit
$\mathbf{C}$	positiv definite Matrix einer quadratischen Form
$\mathcal{C}$	Abstandsquadrat

$C(z^{-1})$	Polynom in z^{-1}
c_i	Polynomkoeffizienten
$d, \mathbf{D}$	Durchgriff, Durchgriffsmatrix
d, d_f	Dämpfungskonstante
e	Fehlersignal, Adaptionfehler
e_s	aufsummierter Regelfehler
E	Elastizitätsmodul von Papier
$\mathbf{E}$	Einheitsmatrix
$\mathbf{e}_Z$	Zustandsfehlervektor
$f(\cdot)$	beliebige Funktion
F_{ij}	Bahnkräfte zwischen den Antrieben i und j an der MODAM
$\mathbf{F}$	Prädiktionsmatrix für die freie Prozessantwort
$\mathbf{g}$	Vektor der Ungleichungsnebenbedingungen
$G(s)$	Prozess- bzw. Streckenübertragungsfunktion
$G_i(s)$	Teilübertragungsfunktion
$G(z^{-1})$	zeitdiskrete Übertragungsfunktion
$\mathbf{G}$	Prädiktionsmatrix für $\mathbf{v}$
H	Dicke von Papier
$H(s)$	Fehlerübertragungsfunktion
$\mathbf{H}$	Prädiktionsmatrix für $\Delta \mathbf{u}$
i, j, k, n	allgemeine Zählvariable
J	Gütefunktion, Kostenfunktion
J	Trägheitsmoment
J_1	Trägheitsmoment des Antriebs, Trägheitsmoment des Motors
J_2	Trägheitsmoment der Last, Trägheitsmoment des Wicklers
k	zeitdiskreter Zählindex
$\mathbf{k}$	Einkoppelvektor der isolierten Nichtlinearität
$\mathbf{K}$	Matrix des prädiktiven Regelgesetzes
$\mathbf{l}$	Beobachter-Rückführvektor, Luenbergervektor
$\mathbf{L}$	Matrix des prädiktiven Regelgesetzes
L_n	freie Bahnlänge
m	Zahl der Ausgangssignale eines MIMO-Systems
M_{gegen}	Amplitude des periodischen Gegenmoments
M_{soll}	Sollmoment
M_T	Torsionsmoment am Zweimassensystem
N	Systemordnung
$\mathbf{N}$	Matrix der Ungleichungsnebenbedingungen
N_1	unterer Prädiktionshorizont
N_2	oberer Prädiktionshorizont
N_u	Stellhorizont
$\mathcal{N}$	isolierte Nichtlinearität
$\widehat{\mathcal{N}}$	geschätzte isolierte Nichtlinearität
o	Zahl der Eingangssignale eines MIMO-Systems
$\mathbf{p}$	Eingangsvektor der periodischen Nichtlinearität
p	Stützwertezahl des GRNN

$\mathbf{P}$	Matrix zur Endzustandsgewichtung
$\mathbf{Q}$	Gewichtungsmatrix
$\mathbf{r}$	Referenztrajektorie
$\mathbb{R}$	Menge der reellen Zahlen
$\mathbb{R}^{n \times m}$	Menge der reellwertigen $n \times m$ Matrizen
$\Re\{.\}$	Realteil
R	Wicklerradius
ΔR	Radiusabweichung, Unrundheit
R_m	mittlerer Radius des Wicklers
t	kontinuierliche Zeit in Sekunden
T	Periodendauer
$\mathbf{T}$	Matrix des prädiktiven Regelgesetzes
T_{ab}	Abtastzeit
T_{ers}	Ersatzzeitkonstante
T_F	Zeitkonstante der Führungsglättung
T_i	Ersatz-Zeitkonstante des Stromregelkreises an der MODAM
$T_{Umr1/2}$	Ersatz-Zeitkonstanten der Stromregelkreise des Zweimassensystems
T_n	Nachstellzeit eines PI-Reglers, Bahnzeitkonstante an der MODAM
$u, \mathbf{u}$	Prozesseingang
Δu	Stellgrößenänderung
$\Delta \mathbf{u}$	Stellgrößenänderungsvektor innerhalb des Stellhorizonts
$\ddot{u}$	Getriebeübersetzung an der MODAM
v	Störgröße, die die Nichtlinearität repräsentiert
$\mathbf{v}$	Störgrößenvektor, der die Nichtlinearität innerhalb des gesamten Prädiktionshorizonts repräsentiert
V	Lyapunov-Funktion
v_0	Arbeitspunktgeschwindigkeit der MODAM
v_2	Leitgeschwindigkeit der MODAM
$\dot{v}_{2soll}$	Sollbeschleunigung der MODAM
v_{ab}	Abwickelgeschwindigkeit
V_R	Verstärkung eines PI-Reglers
w	Sollwert
$\mathbf{x}$	Zustandsvektor
$\mathbf{x}_E$	Eingangsraum der Nichtlinearität
$y, \mathbf{y}$	Prozessausgang, Regelgröße
$\tilde{y}(k+j)$	prädizierte Regelgröße für den Zeitpunkt $k+j$
$\tilde{\mathbf{y}}$	prädizierter Regelgrößenvektor innerhalb des Prädiktionshorizonts
$\tilde{y}(k+j k)$	prädiziertes y für den Zeitpunkt $k+j$, berechnet zum Zeitpunkt k
z	Störgröße
z^{-1}	zeitdiskreter Ein-Schritt Rückwärts-Schiebeoperator, Variable der z-Transformation

Literaturverzeichnis

Bücher

- [1] J. ACKERMANN: *Abtastregelung*. Springer Verlag, Berlin Heidelberg, 1972.
- [2] J. ACKERMANN: *Abtastregelung*. Springer Verlag, Berlin Heidelberg, 1988.
- [3] F. ALLGÖWER, A. ZHENG: *Nonlinear Model Predictive Control*. Birkhäuser Verlag, Basel Boston Berlin, 2000.
- [4] A. ANGERMANN, M. BEUSCHEL, M. RAU, U. WOHLFARTH: *Matlab - Simulink - Stateflow, Grundlagen, Toolboxes, Beispiele*. Oldenbourg Verlag, München, 2003.
<http://www.matlabbuch.de>
- [5] R. BRAUSE: *Neuronale Netze – Eine Einführung in die Neuroinformatik*. Teubner Verlag, Stuttgart, 1995.
- [6] E.F. CAMACHO, C. BORDONS: *Model Predictive Control*. Springer-Verlag, London Berlin Heidelberg, 1999.
- [7] S. ENGELL: *Entwurf nichtlinearer Regelungen*. R. Oldenbourg Verlag, München Wien, 1995.
- [8] O. FÖLLINGER: *Nichtlineare Regelungen I*. R. Oldenbourg Verlag, München Wien, 1993.
- [9] O. FÖLLINGER: *Nichtlineare Regelungen II*. R. Oldenbourg Verlag, München Wien, 1993.
- [10] O. FÖLLINGER: *Regelungstechnik*. Hütinger Buch Verlag, Heidelberg, 1990.
- [11] R. ISERMANN: *Identifikation dynamischer Systeme – Band 1*. Springer Verlag, Berlin, 1992.
- [12] R. ISERMANN: *Identifikation dynamischer Systeme – Band 2*. Springer Verlag, Berlin, 1992.
- [13] L. LJUNG: *System Identification*. Prentice Hall, New Jersey, 1999.
- [14] G. LUDYK: *Theoretische Regelungstechnik 1*. Springer Verlag, Berlin, 1995.
- [15] G. LUDYK: *Theoretische Regelungstechnik 2*. Springer Verlag, Berlin, 1995.
- [16] K. MEYBERG, P. VACHENAUER: *Höhere Mathematik 1 und 2*. Springer Verlag, Berlin, Heidelberg, 1991.

- [17] K. S. NARENDRA, A. M. ANNASWAMY: *Stable Adaptive Systems*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1989.
- [18] O. NELLES, S. ERNST, R. ISERMANN: *Neuronale Netze zur Identifikation nicht-linearer Systeme: Ein Überblick*. at - Automatisierungstechnik, Nr. 45, 6/1997, Oldenburg Verlag, München, 1997.
- [19] O. NELLES: *Nonlinear System Identification*. Springer-Verlag, Berlin Heidelberg, 2001.
- [20] M. NIKOLAOU: *Model Predictive Controllers: A Critical Synthesis of Theory and Industrial Needs*. Advances in Chemical Engineering Series, Academic Press, 2001.
<http://www.chee.uh.edu/faculty/nikolaou/MPCtheoryRevised.pdf>
- [21] M. NØRGAARD, O. RAVN, N.K. POULEN, L.K. HANSEN: *Neural Networks for Modelling and Control of Dynamic Systems*. Springer-Verlag, London Berlin Heidelberg, 2000.
- [22] K. OGATA: *Discrete-Time Control Systems*. Prentice-Hall, Englewood Cliffs, 1987.
- [23] A. OPPENHEIM, A. WILLSKY: *Signale und Systeme*. VCH, Weinheim, 1992.
- [24] M. PAPAGEORGIOU: *Optimierung: Statische, Dynamische, Stochastische Verfahren für die Anwendung*. Oldenbourg Verlag, München, 1996.
- [25] G. SCHMIDT: *Grundlagen der Regelungstechnik*. Springer-Verlag, Berlin Heidelberg New York, 1991.
- [26] G. SCHMIDT: *Lernverfahren in der Automatisierungstechnik*. Vorlesungsskriptum, Lehrstuhl für Steuerungs- und Regelungstechnik, TU-München, 1997.
- [27] J.M. SÁNCHEZ, J. RODELLAR: *Adaptive Predictive Control*. Prentice Hall, London New York, 1996.
- [28] D. SCHRÖDER: *Elektrische Antriebe – Grundlagen*. Springer-Verlag, Berlin Heidelberg New York, 2. Auflage, 2000.
- [29] D. SCHRÖDER: *Elektrische Antriebe, Regelung von Antriebssystemen*. Springer-Verlag, Berlin Heidelberg New York, 2. Auflage, 2001.
- [30] D. SCHRÖDER: *Elektrische Antriebe 3, Leistungselektronische Bauelemente*. Springer-Verlag, Berlin Heidelberg New York, 1996.
- [31] D. SCHRÖDER: *Elektrische Antriebe 4, Leistungselektronische Schaltungen*. Springer-Verlag, Berlin Heidelberg New York, 1998.
- [32] D. SCHRÖDER (ED.): *Intelligent Observer and Control Design for Nonlinear Systems*. Springer-Verlag, Berlin Heidelberg New York, 2000.
- [33] J.-J. E. SLOTTINE, W. LI: *Applied Nonlinear Control*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1991.

- [34] T. SOEDERSTROEM, P. STOICA: *System Identification*. University Press Cambridge, 1989.

Dissertationen und Diplomarbeiten

- [35] F. BERCHTENBREITER: *Implementierung einer prädiktiven Regelung für Systeme mit isolierten Nichtlinearitäten*. Diplomarbeit, TU München, Germany, 2002.
- [36] M. BEUSCHEL: *Neuronale Netze zur Diagnose und Tilgung von Drehmomentschwingungen am Verbrennungsmotor*. Dissertation, TU München, Germany, 2000.
- [37] G. BRANDENBURG: *Über das dynamische Verhalten durchlaufender elastischer Stoffbahnen bei Kraftübertragung durch Coulomb'sche Reibung in einem System angetriebener, umschlungener Walzen*. Dissertation, TU München, Germany, 1971.
- [38] H. CHEN: *Stability and Robustness Considerations in Nonlinear Model Predictive Control*. Dissertation Universität Stuttgart, VDI Fortschritt-Berichte, Reihe 8, Nr. 674, Düsseldorf, 1997.
- [39] M. FISCHER: *Fuzzy-modellbasierte Regelung nichtlinearer Prozesse*. Dissertation TU-Darmstadt, VDI Fortschritt-Berichte, Reihe 8, Nr. 750, Düsseldorf, 1999.
- [40] TH. FRENZ: *Stabile Neuronale Online Identifikation und Kompensation statischer Nichtlinearitäten am Beispiel von Werkzeugmaschinenvorschubantrieben*. Dissertation, TU München, Germany, 1998.
- [41] CH. HINTZ: *Identifikation und Regelung nichtlinearer mechatronischer Antriebssysteme mit neuronalen Ansätzen*. Diplomarbeit, TU München, Germany, 1998.
- [42] W. HÖGER: *Ein Beitrag zur Systemdynamik von Wickelantrieben unter Berücksichtigung elastischer Kopplungen*. Dissertation, TU München, Germany, 1986.
- [43] I. JENAYEH: *Robuste Modellgestützte Prädiktive Regelung mit Hilfe von Linearen Matrixungleichungen*. Dissertation RWTH-Aachen, VDI Fortschritt-Berichte, Reihe 8, Nr. 839, Düsseldorf, 2000.
- [44] P. KRAUSS: *Prädiktive Regelung mit linearen Prozeßmodellen im Zustandsraum*. Dissertation RWTH-Aachen, VDI Fortschritt-Berichte, Reihe 8, Nr. 560, Düsseldorf, 1996.
- [45] J. KURTH: *Identifikation nichtlinearer Systeme mit komprimierten Volterra-Reihen*. Dissertation RWTH-Aachen, VDI Fortschritt-Berichte, Reihe 8, Nr. 459, Düsseldorf, 1995.
- [46] U. LENZ: *Lernfähige neuronale Beobachter für eine Klasse nichtlinearer dynamischer Systeme und ihre Anwendung zur intelligenten Regelung von Verbrennungsmotoren*. Dissertation, TU München, Germany, 1998.

- [47] MATHWORKS: *Control System Toolbox Users's Guide Version 5*. Mathworks, 2002.
- [48] MATHWORKS: *Optimization Toolbox Users's Guide*. Mathworks, 2002.
- [49] MATHWORKS: *System Identification Toolbox Users's Guide*. Mathworks, 2002.
- [50] V. NEVISTIĆ: *Constrained Control of Nonlinear Systems*. Dissertation ETH No. 12042, Zürich, 1997. <http://www.aut.ee.ethz.ch/~vesna/>
- [51] H. POMMER: *Experimentelle Identifikation und Kompensation von Lose*. Diplomarbeit, TU München, Germany, 1998.
- [52] M. RAU: *Identifikation und Kompensation von Nichtlinearitäten an der MODAM mit neuronalen Ansätzen*. Diplomarbeit, TU München, Germany, 1997.
- [53] C. SALCHER: *Implementierung eines echtzeitfähigen Regelungssystems an einer kontinuierlichen Fertigungsanlage*. Diplomarbeit, TU München, Germany, 1999.
- [54] C. SCHÄFFNER: *Analyse und Synthese neuronaler Regelungsverfahren*. Herbert Utz Verlag Wissenschaft, München, Germany, 1996.
- [55] H. SCHUSTER: *Prädiktive Regelung für dynamische Systeme mit isolierten Nichtlinearitäten*. Diplomarbeit, TU München, Germany, 2002.
- [56] S. STRAUB: *Entwurf und Validierung neuronaler Beobachter zur Regelung nichtlinearer dynamischer Systeme im Umfeld antriebstechnischer Problemstellungen*. Dissertation, TU München, Germany, 1998.
- [57] D. STROBL: *Identifikation nichtlinearer mechatronischer Systeme mittels neuronaler Beobachter*. Dissertation, TU München, Germany, 1999.
- [58] M. WELLERS: *Nichtlineare Modellgestützte Prädiktive Regelung auf Basis von Wiener- und Hammerstein-Modellen*. Dissertation RWTH-Aachen, VDI Fortschritt-Berichte, Reihe 8, Nr. 742, Düsseldorf, 1998.
- [59] W. WOLFERMANN: *Mathematischer Zusammenhang zwischen Bahnzugkraft und inneren Spannungen beim Wickeln von elastischen Stoffbahnen*. Dissertation am Lehrstuhl für Elektrische Antriebstechnik der TU München, 1976.

Aufsätze und Berichte

- [60] J. ACKERMANN: *Der Entwurf linearer Regelungssysteme im Zustandsraum*. Regelungstechnik, Nr. 20, 1972.
- [61] F. ALLGÖWER, R. FINDEISEN, Z. NAGY, M. DIEHL, G. BOCK, J. SCHLÖDER: *Nonlinear Model Predictive Control for Large Scale Systems*. Proc. of the 6th Int. Conf. on Methods and Models in Automation and Robotics, MMAR 2000, pp. 43-54, Poland, 2000.

- [62] A. BOLLIG, S. MANN, M. ENNING, S. KAIERLE: *Modellgestützte Prädiktive Regelung beim Laserschweißen*. Automatisierungstechnische Praxis atp, Heft 7, pp. 35-39, 2001.
- [63] H. CHEN, F. ALLGÖWER: *A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability*. Automatica, Vol. 34, No. 10, pp. 1205-1217, 1998.
- [64] D. CLARKE, C. MOHTADI, P. TUFFS: *Generalized Predictive Control - Part I. The Basic Algorithm*. Automatica, Vol. 23, No. 2, pp. 137-148, 1987.
- [65] D. CLARKE, C. MOHTADI, P. TUFFS: *Generalized Predictive Control - Part II. Extensions and Interpretations*. Automatica, Vol. 23, No. 2, pp. 149-160, 1987.
- [66] C. CUTLER, B. RAMAKER: *Dynamic Matrix Control – a computer control algorithm*. Proc. Joint Automatic Control Conference, San Francisco, 1980.
- [67] H. DEMIRCIÖGLU, P. GAWTHROP: *Muliti-variable Continuous-time Generalized Predictive Control (MCGPC)*. Automatica, Vol. 28, No. 4, 1992.
- [68] F. FONTES: *Overview of Nonlinear Model Predictive Control Schemes leading to Stability*. <http://www.mct.uminho.pt/ffontes/pub/controlo2k.ps>
- [69] C. GARCIA, A. MORSHEDI: *Quadratic Programming Solution of Dynamic Matrix Control (QDMC)*. Chem. Engineering Community, Vol. 46, 1986.
- [70] C. GARCIA, D. PRETT, M. MORARI: *Model Predictive Control: Theory and Practice – a Survey*. Automatica, Vol. 25, No. 3, pp. 335-348, 1989.
- [71] F. D. HANGL, U. LENZ, D. SCHRÖDER: *Theorie des systematischen Entwurfs lernfähiger Beobachter für eine Klasse nichtlinearer Strecken*. GMA-Fachausschuß 1.4 Theoretische Verfahren der Regelungstechnik, Workshop, Interlaken, Switzerland, 1997.
- [72] R. E. KALMAN, R. S. BUCY: *New results in linear filtering and prediction theory*. Transactions on ASME, Vol. 83 D, 1961, pp. 95-108.
- [73] V. KURKOVÁ, R. NERUDA: *Uniqueness of Functional Representation by Gaussian Basis Function Networks*. ICANN, Sorrento, 1994.
- [74] D. G. LUENBERGER: *Observing the State of Linear Systems*. IEEE Transactions on Military Electronics, 1964, pp. 74-80.
- [75] D. G. LUENBERGER: *Observers for Multivariable Systems*. IEEE Transactions on Automatic Control, 1966, pp. 190-197.
- [76] P. LUNDSTRÖM, J. LEE, M. MORARI, S. SKOGESTAD: *Limitations of Dynamic Matrix Control*. Automatica, Vol. 19, No. 4, 1995.
- [77] A. M. LYAPUNOV: *The General Problem of the Stability of Motion*. Taylor & Francis, London and Washington, 1992.

- [78] D. MAYNE, J. RAWLINGS, C. RAO, P. SCOKAERT: *Constrained model predictive control: Stability and optimality*. Automatica, Vol. 36, No. 6, pp. 789-814, 2000.
- [79] D. MAYNE, H. MICHALSKA: *Receding Horizon Control of Nonlinear Systems*. IEEE Transactions on Automatic Control, Vol. 35, No. 7, pp. 814-824, 1990.
- [80] H. MICHALSKA, D. MAYNE: *Robust Receding Horizon Control of Constrained Nonlinear Systems*. IEEE Transactions on Automatic Control, Vol. 38, No. 11, pp. 1623-1633, 1993.
- [81] A. MORSHEDI, C. CUTLER, T. SKROVANEK: *Optimal solution of dynamic matrix control with linear programming techniques (LDMC)*. American Control Conference, 1985.
- [82] Z. NAGY, R. FINDEISEN, M. DIEHL, F. ALLGÖWER, H. BOCK, S. AGACHI, J. SCHLÖDER, D. LEINWEBER: *Real-Time Feasibility on Nonlinear Predictive Control for Large Scale Processes – a Case Study*. IST Report No. 200-4, Institut für Systemtheorie, Universität Stuttgart, 2000.
- [83] M. RAU, D. SCHRÖDER: *Adaptive Observer and Control Design for a Class of Nonlinear Systems*. 7th European Congress on Intelligent Techniques & Soft Computing EUFIT, Aachen, Germany, 1999.
- [84] J. RAWLINGS: *Tutorial Overview of Model Predictive Control*. IEEE Control Systems Magazine, Vol. 20, No. 3, pp. 38-52, 2000.
- [85] J. RAWLINGS, E. MEADOWS, K. MUSKE: *Nonlinear Model Predictive Control: A Tutorial and Survey*. Proceedings of ADCHEM 94, Kyoto, Japan, 1994.
- [86] K. SCHITTKOWSKI: *Solution of quadratic Programming Problems*. Mathematisches Institut, Universität Bayreuth.
<http://www.uni-bayreuth.de/departments/math/~kschittkowski/software.htm>
- [87] A. SCHWARM, M. NIKOLAOU: *Chance Constrained Model Predictive Control*. AI-ChE Journal, Vol. 45, No. 8, pp. 1605-1844, 1998.
<http://www.chee.uh.edu/faculty/nikolaou/chanceconstrained.pdf>
- [88] P. SCOKAERT, J. RAWLINGS: *On Infeasibilities in Model Predictive Control*. 5th International Conference on Chemical Process Control CPC, 1996.
- [89] R. SOETERBOEK: *Predictive Control – A Unified Approach*. Dissertation, TU-Delft, 1990.
- [90] D. F. SPECHT: *A General Regression Neural Network*. IEEE Transactions on Neural Networks, Vol. 2, No. 6, 1991.
- [91] S. TZAFESTAS, G. VAGELATOS: *A new generalized predictive control algorithm*. Conference on Decision and Control, Brighton, 1991.
- [92] M. ZEITZ: *Nichtlineare Beobachter für chemische Reaktoren*. Fortschrittsberichte der VDI-Zeitschriften, Reihe 8, Nr. 27, VDI-Verlag, Düsseldorf, 1977.