

# Anwendung künstlicher neuronaler Netze zur nichtlinearen adaptiven Regelung

*Dissertation*

zur

Erlangung des Grades eines  
Doktoringenieurs

der

Fakultät für Elektrotechnik und Informationstechnik  
an der Ruhr-Universität Bochum

von

**Torsten Knohl**

Marl

Bochum 2000

Dissertation eingereicht am : 23. Oktober 2000

Tag der mündlichen Prüfung : 12. Januar 2001

Referent : Prof. Dr.-Ing. H. Unbehauen

Korreferent : Prof. Dr.-Ing. H. D. Fischer

# Vorwort

Das vorliegende Buch entstand als Dissertation während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Elektrische Steuerung und Regelung der Ruhr-Universität Bochum.

Herrn Professor Dr.-Ing. H. Unbehauen bin ich für die Anregung zu dieser Arbeit und sein stetes Interesse am Fortgang der Untersuchungen zu großem Dank verpflichtet.

Herrn Professor Dr.-Ing. H. D. Fischer danke ich für die bereitwillige Übernahme des Mitberichtes und das Interesse an der Arbeit.

In den gut fünf Jahren, die ich am Lehrstuhl ESR gearbeitet habe, bildete das hervorragende Arbeitsklima am Lehrstuhl eine optimale Basis für die entstandene Arbeit. Die Kollegen akzeptierten sogar, dass ich nie in der ESR-Fußballmannschaft mitgespielt habe. In vielen Diskussionen über die Arbeit und das Geschehen außerhalb des Lehrstuhls wurden Gedanken ausgetauscht sowie Ideen entwickelt und verworfen. Hierfür bedanke ich mich bei den Kollegen, wobei ich insbesondere die Herren Dr.-Ing. C. Bohn, Dr.-Ing. J. Dastyh, Dr.-Ing. A. Gambier, Dr.-Ing. P. Kortmann und Dipl.-Ing. J. Uhlig nennen möchte.

Bei Herrn Professor Dr.-Ing. C. Schmid bedanke ich mich für die kritische und hilfreiche Durchsicht des Manuskripts. Für die Erstellung zahlreicher Bilder und das gründliche Korrekturlesen gilt mein Dank Frau A. Marschall und Frau S. Romberg. Allen Studenten, die im Rahmen von Studien- und Diplomarbeiten ihren Teil zur Entstehung dieser Arbeit beigetragen haben, gilt ebenfalls mein Dank.

Meiner Frau Nicola und meinen Kindern Philipp Jonas und Momo Carina möchte ich meinen besonderen Dank aussprechen. Meine Frau hat mich immer bei der Entstehung dieser Arbeit unterstützt und mir, speziell in der Endphase dieser Arbeit, den Rücken freigehalten. In regelmäßigen Abständen haben die Drei mir sehr liebevoll gezeigt, dass es wichtigere Dinge im Leben gibt als künstliche neuronale Netze und adaptive Regler.

Bochum, Januar 2001

Torsten Knohl

Für Nicola, Jonas und Momo

*Laß' dir von keinem Fachmann imponieren, der dir erzählt: „Lieber Freund, das mache ich schon seit 20 Jahren so!“ Man kann eine Sache auch 20 Jahre lang falsch machen.*

KURT TUCHOLSKY

*Wenn man zwei Stunden lang mit einem netten Mädchen zusammensitzt, meint man, es wäre eine Minute. Sitzt man eine Minute auf einem heissen Ofen, meint man, es wären zwei Stunden. Das ist Relativität.*

ALBERT EINSTEIN

# Inhaltsverzeichnis

Notation	VI
<b>1 Einleitung und Problemstellung</b>	<b>1</b>
<b>2 Nichtlineare adaptive Regelung</b>	<b>10</b>
2.1 Linearisierung durch Zustandsrückführung . . . . .	11
2.2 Linearisierung durch Ausgangsrückführung . . . . .	15
2.3 Adaptive <i>Sliding-Mode</i> -Regelung . . . . .	22
2.4 Regelung mit dem <i>Speed-Gradienten</i> -Verfahren . . . . .	26
2.5 Regelung von Systemen mit statischen Nichtlinearitäten . . . . .	31
2.6 Reglerstrukturen mit künstlichen neuronalen Netzen . . . . .	38
2.6.1 Das Radial-Basis-Funktionen(RBF)-Netz . . . . .	39
2.6.2 Kopieren eines existierenden Reglers . . . . .	43
2.6.3 Adaptive Regelung mit Vergleichsmodell . . . . .	44
2.6.4 Adaptive Regelung mit internem Prozessmodell . . . . .	45
2.7 Zusammenfassung . . . . .	46
<b>3 Adaptive Regelung von Hammerstein-Systemen</b>	<b>48</b>
3.1 Indirekte adaptive Regelung . . . . .	49
3.1.1 Der Reglerentwurf . . . . .	51
3.1.1.1 Parameteridentifikation . . . . .	53
3.1.1.2 Entwurf des LQ-optimalen Reglers . . . . .	55
3.1.1.3 Training von ANN-2 . . . . .	56

---

3.1.2	Die Wahl der Netzparameter . . . . .	57
3.1.3	Stabilitätsbetrachtung . . . . .	58
3.2	Duale Adaptive Regelung . . . . .	59
3.2.1	Der Reglerentwurf . . . . .	61
3.2.1.1	Duale Modifikation . . . . .	62
3.2.1.2	Verbesserung des dualen Reglers . . . . .	69
<b>4</b>	<b>Adaptive Regelung von Wiener-Systemen</b>	<b>71</b>
4.1	Die WANNNA-Reglerstruktur . . . . .	72
4.2	Das Wiener-ANN-Modell . . . . .	74
4.3	Parameteridentifikation mit dem RPE-Verfahren . . . . .	75
4.4	Parameteridentifikation mit dem RLS-Verfahren . . . . .	79
<b>5</b>	<b>Das erweiterte <i>Backstepping</i>-Verfahren</b>	<b>83</b>
5.1	ANN <i>Backstepping</i> -Regelung . . . . .	84
5.2	Stabilität der erweiterten <i>Backstepping</i> - Regelung . . . . .	88
5.3	Simulationsbeispiel . . . . .	91
5.4	Zusammenfassung und Bewertung . . . . .	95
<b>6</b>	<b>Erprobung der Reglerstrukturen</b>	<b>97</b>
6.1	Positionsregelung eines hydraulischen Vorschubzylinders . . . . .	98
6.2	Regelung eines flexiblen Manipulators . . . . .	109
6.3	Duale Regelung eines hydraulischen Vorschubzylinders . . . . .	114
6.4	Simulationsbeispiele mit der RPE-WANNNA-Regelung . . . . .	118
6.5	pH-Wert-Regelung eines Neutralisationsreaktors . . . . .	122
<b>7</b>	<b>Zusammenfassung</b>	<b>127</b>
	<b>Literaturverzeichnis</b>	<b>131</b>

# Notation

## Allgemeines

Matrizen	Fett und Großbuchstaben ( z.B. <b><i>A</i></b> , <b><i>B</i></b> , ... )
Vektoren	Fett und Kleinbuchstaben ( z.B. <b><i>a</i></b> , <b><i>a</i></b> , ... )
Konstanten	Senkrechte Schrift (z.B. $a, b, \dots$ )
Variablen	Kursive Schrift (z.B. $a, b, \dots$ )
Polynome	Kursive Schrift und Großbuchstaben (z.B. $A(z^{-1})$ )

## Akronyme

ANN	Künstliche ( <u>A</u> r <u>tificial</u> ) <u>n</u> euronale <u>N</u> etze
ANNNA	Künstliche ( <u>A</u> r <u>tificial</u> ) <u>n</u> euronale <u>N</u> etze zur <u>n</u> ichtlinearen <u>a</u> daptiven Regelung
BIBO	<u>B</u> ounded- <u>I</u> nput <u>B</u> ounded- <u>O</u> utput
BP	<u>B</u> ack <u>p</u> ropagation-Algorithmus
D-HANNNA	<u>D</u> uale <u>A</u> NNNA-Regelung für <u>H</u> ammerstein-Systeme
HANNNA	<u>A</u> NNNA-Regelung für <u>H</u> ammerstein-Systeme
LIP	<u>L</u> inear <u>i</u> n den <u>P</u> arametern
LS	<u>L</u> ineares <u>S</u> ystem
LSE	<u>L</u> east <u>s</u> quares <u>e</u> rror
LQ	<u>L</u> inearer <u>q</u> uadratisch optimaler Regler
MLP	<u>M</u> ult <u>i</u> layer <u>P</u> erzeptron-Netz

---

NL	<u>N</u> icht <u>l</u> inearität
NPSF	<u>N</u> ichtlineare <u>PSF</u> -Form
POF	<u>P</u> arametric- <u>O</u> utput- <u>F</u> eedback-Form
PSF	<u>P</u> arametric- <u>S</u> trict- <u>F</u> eedback-Form
RBF	<u>R</u> adiale <u>B</u> asis <u>f</u> unktionen-Netze
RLS	<u>R</u> ecursive <u>l</u> east <u>s</u> quares
RPE	<u>R</u> ecursive <u>p</u> rediction <u>e</u> rror
SG	<u>S</u> peed- <u>G</u> radienten
SISO	<u>S</u> ingle- <u>I</u> ntput <u>S</u> ingle- <u>O</u> utput
WANNNA	<u>A</u> NNNA-Regelung für <u>W</u> iener-Systeme

## Formelzeichen und Operatoren

$\hat{\phantom{x}}$	Geschätzter Signalwert oder Parameter
$\alpha_i$	Stabilisierungsfunktion
$\beta \cdot$	Ausgangsfunktion einer nichtlinearen Regelstrecke
$\varepsilon(k)$	Gleichungsfehler
$\Gamma$	Gewichtungsmatrix
$\nabla$	Nablaoperator
$\vartheta$	Reglerparametervektor
$\mathfrak{w}$	Formparameter der RBF-Aktivierungsfunktion
$\varphi(\cdot)$	Nichtlineare Vektorfunktion
$\Phi(\cdot)$	Aktivierungsfunktion eines RBF-Neurons
$\Psi$	Gradientenvektor
$\mathbf{A}$	Systemmatrix
$A(s)$	Nennerpolynom einer linearen Regelstrecke
$A(z^{-1})$	Nennerpolynom einer zeitdiskreten linearen Regelstrecke
$b_l, b_r$	Linker und rechter Achsenabschnitt einer Totzone
$\mathbf{B}$	Steuerungsmatrix
$B(s)$	Zählerpolynom einer linearen Regelstrecke
$B(z^{-1})$	Zählerpolynom einer zeitdiskreten linearen Regelstrecke
$\mathbf{C}$	Beobachtungsmatrix



---

<b>c</b>	Zentrum eines RBF-Neurons
<b>D</b>	Durchgangsmatrix
$E\{\cdot\}$	Erwartungswert
$G(s)$	Übertragungsfunktion
$\mathcal{L}\{\cdot\}$	Laplace-Transformation
$\mathcal{L}^{-1}\{\cdot\}$	Inverse Laplace-Transformation
$m_l, m_r$	Linke und rechte Steigung einer Totzone
$m$	Zahl der Neuronen in der versteckten Schicht eines ANN
<b>m</b>	Datenvektor
<b>M</b>	Datenmatrix
$n.$	Ordnung eines Polynoms
<b>p</b>	Parametervektor
$\mathbf{p}_{b_1 \mathbf{p}_0}$	Kovarianz von $b_1$ und $\mathbf{p}_0$
<b>P</b>	Kovarianzmatrix einer Parameterschätzung
$P(z)$	Charakteristisches Polynom
$q$	Kalmanscher Verstärkungsfaktor
$R(z)$	Nennerpolynom eines LQ-Reglers
$S(z)$	Zählerpolynom eines LQ-Reglers
$\cdot^T$	Transponiert Operator
$T(z)$	Beobachterpolynom eines LQ-Reglers
$\text{tr}(\cdot)$	Spur einer Matrix
$\text{sgn}$	Signumfunktion
$w'$	Sollwert für die Regelgröße $y'$
$w_i$	Ausgangsgewicht des $i$ -ten Neurons
<b>x</b>	Zustandsvektor eines Systems
$x_i$	$i$ -ter Zustand eines Systems
$y$	Ausgangssignal eines Systems
$y'$	Ausgangssignal des linearen Teilsystems eines Wiener-Systems
$\mathcal{Z}\{\cdot\}$	z-Transformation
$\mathcal{Z}^{-1}\{\cdot\}$	Inverse z-Transformation
<b>z</b>	Zustandsvektor eines Fehlersystems

# 1 Einleitung und Problemstellung

Die Entwicklung der adaptiven Regelung hat um 1950 mit ersten Veröffentlichungen im Rahmen von Konferenzen begonnen. Das Hauptziel hierfür war die Beherrschung immer komplexerer Regelstrecken, deren Parameter unbekannt waren oder sich mit der Zeit ändern. Derartige Regelstrecken ließen sich nicht mehr befriedigend mit fest eingestellten Reglern beherrschen. Adaptive Regelungen sind den konventionellen, festeingestellten Reglern weit überlegen. Können letztere nur für einen Arbeitspunkt oder einen bestimmten Arbeitsbereich ein optimales Systemverhalten erzielen, kann dies mit selbst anpassenden Regelungen über den gesamten Arbeitsbereich einer Regelstrecke erreicht werden. Seit den Anfängen, als die adaptive Regelung wegen der hohen Realisierungskosten weitgehend nur bei militärischen Problemstellungen eingesetzt werden konnte, hat sie sowohl auf dem Gebiet der Theorie als auch in der industriellen Praxis immer mehr an Bedeutung gewonnen. Speziell durch die schnelle Entwicklung der Mikrorechner und durch die zur Verfügung stehenden Entwicklungs- und Simulationswerkzeuge, wie z.B. *Matlab*, *Mathematica* oder *Maple*, konnten die verschiedenartigen adaptiven Regelverfahren systematisch auf zahlreichen Anwendungsgebieten eingesetzt, verbessert sowie an vielen technischen Regelstrecken mit großem Erfolg erprobt werden.

Obwohl nichtlineare Zusammenhänge der Systemgrößen in technischen Anlagen für das dynamische Verhalten einer Regelstrecke eine große Rolle spielen, werden für den Entwurf adaptiver Regelungen auch heute noch weitgehend *lineare Ansätze* verwendet. Dieses Vorgehen liefert aber nur befriedigende Ergebnisse, wenn die Regelstrecke in einem bekannten Arbeitspunkt oder Ar-

beitsbereich betrieben wird, oder die Änderungen der Regelstreckenparameter relativ langsam erfolgen, so dass die Identifikationsstufe die bestimmten Parameter eines Regelstreckenmodells den wahren Parametern nachführen kann. Bei einer sich schnell ändernden oder gar unbekannten Regelstreckendynamik, z.B. durch einen plötzlichen Arbeitspunktwechsel, führt diese Vereinfachung aufgrund der Linearisierung aber zu einem unvorhersehbaren dynamischen Regelverhalten, zu einer schlechten Regelgüte oder gar zur Instabilität der gesamten Anlage. Derartige Situationen müssen bei technischen Anlagen in jedem Fall vermieden werden. Hierbei sind jedoch nicht nur die wichtigen Aspekte der Stabilität und hoher Regelgüte der Anlage zu berücksichtigen, es muss auch die Minimierung der Amplitude der Stellgröße bzw. des Stellaufwandes im Sinne eines zuvor festgelegten Kriteriums gewährleistet werden. Ein nicht minimaler Stellaufwand oder eine geringe Regelgüte bedeuten stets erhöhte Energie- und Betriebskosten sowie eine Verminderung der Produktivität.

Aus den oben aufgeführten Nachteilen der Linearisierung von nichtlinearen Regelstrecken folgt eine nur begrenzte Einsatzfähigkeit solcher Ansätze. Die systematische Entwicklung von Adaptivreglern auf der Basis *nichtlinearer* Modelle begann erst Anfang der 90er Jahre z. B. mit Beiträgen von Kannelakopoulos et al. (1990), Nijmeijer und Campion (1993) und Seto et al. (1992). Die hierfür heute allgemein benutzte Bezeichnung NICHLINEARE ADAPTIVE REGELUNG (*nonlinear adaptive control* NAC oder im Weiteren auch als NA-Regelung bezeichnet) ist zwar ein wenig unglücklich, da sie die Sache selbst schlecht beschreibt. Auch bei Verwendung eines linearen Ansatzes für die Regelstrecke, ist jede adaptive Regelung stets ein nichtlineares zeitvariantes System. Trotzdem soll aber diese Bezeichnung beibehalten werden.

Anders als bei den herkömmlichen Entwurfsverfahren wird bei diesen Verfahren nicht das linearisierte Regelstreckenmodell beim Entwurf verwendet, sondern das nichtlineare Gesamtmodell, das die nichtlinearen Eigenschaften der Regelstrecke beschreibt. Der Vorteil dieser Vorgehensweise ist leicht einzusehen. Gelingt es, einen adaptiven Regler anhand des nichtlinearen Gesamtmodells zu entwickeln, ist dieser für beliebige Arbeitspunkte geeignet und kann Parameteränderungen folgen, die z. B. durch Verschleiß oder Umweltänderungen auftreten.

Ein in der Literatur viel beachtetes Entwurfsverfahren zur adaptiven Regelung von nichtlinearen Regelstrecken auf der Basis nichtlinearer Regelstreckenmodelle ist das *Adaptive Backstepping*-Verfahren. Es wurde erstmals von Kanellakopoulos et al. (1991c) beschrieben und in zahlreichen Veröffentlichungen (Kanellakopoulos et al., 1991a; Kokotović, 1991; Kokotović et al., 1992; Marino et al., 1993; Kanellakopoulos et al., 1993; Seto et al., 1994; Krstić und Kokotović, 1995) immer wieder aufgegriffen und erweitert. Voraussetzung für dieses Verfahren ist, dass die Regelstrecke in der Parametric-Strict-Feedback-Zustandsraumbeschreibung (PSF-Form) vorliegt. Bei dieser Darstellung hängt jeder Zustand linear vom folgenden und über einen nichtlinearen Zusammenhang von allen zeitlich zurückliegenden Zuständen ab. Für die Modellunsicherheiten wird angenommen, dass sie multiplikativ mit den Nichtlinearitäten verknüpft sind. Es ergibt sich so eine „dreiecksförmige“ Zustandsraumdarstellung (siehe Abschnitte 2.1 und 2.2). Der Grundgedanke ist die sequenzielle Stabilisierung jedes einzelnen Zustandes durch den Ansatz einer geeigneten Ljapunow-Funktion. Durch geschickte Erweiterung dieser Ljapunow-Funktion können zusätzlich die Adaptiongesetze für die unbekannten Modellparameter bestimmt werden. Vor dem praktischen Einsatz dieser Verfahren müssen noch einige Fragestellungen geklärt werden. Dies sind beispielsweise die Realisierung der benötigten Rechnerleistung und die optimale Einstellung der zahlreichen Reglerparameter. Als weiteres wird für den Stabilitätsbeweis Stellenergie in beliebiger Größe vorausgesetzt, was in der Realität natürlich nicht gegeben ist.

Eine ganz andere Vorgehensweise ist die Methode zum Entwurf einer nichtlinearen adaptiven *Sliding-Mode*-Regelung. Hierbei wird das Stellgesetz mit Hilfe des schon für sich robusten, konventionellen *Sliding-Mode*-Reglers (Utkin, 1992; Slotine und Li, 1991) entworfen. Dieser Entwurf ist aber nur für geringe Parameterabweichungen sinnvoll, da sonst unnötig hohe Stellsignale benötigt werden. Aus diesem Grund werden bei der adaptiven *Sliding-Mode*-Regelung (Li und Lee, 1992; Orlov, 1993; Llanes-Santiago et al., 1993) anstatt fester Regelstreckenparameter deren Schätzwerte verwendet und diese mittels eines Adaptiongesetzes eingestellt. Durch diese Erweiterung können Regel-

strecken mit sowohl schnellen Parameteränderungen als auch mit unbekannten Parametern geregelt werden.

Das *Speed-Gradient*(SG)-Verfahren wurde von Krasovskii (1976) als allgemeines Optimierungsverfahren vorgestellt und dann später von Fradkov (1979) zur adaptiven Regelung nichtlinearer Systeme eingesetzt. Der Ansatz wurde von russischsprachigen Autoren entwickelt und hat in der westlichen (englischsprachigen) Welt bislang kaum Beachtung gefunden. Anders als bei den bisher besprochenen Methoden ist dieser Ansatz zunächst sehr allgemein gehalten und kann somit an eine große Anzahl von Systemdarstellungen angepasst werden. Ausgehend von einem vorgegebenen Gütefunktional werden die von außen beeinflussbaren Parameter in Richtung des negativen Gradienten der „Gütefunktionalgeschwindigkeit“ geändert, wobei sich die Gütefunktionalgeschwindigkeit die Ableitung des Gütefunktionals nach der Zeit ist. Es wird dabei nicht zwischen Stellgesetz und Parameteradaptionsgesetz unterschieden.

Große Aufmerksamkeit im Bereich der nichtlinearen adaptiven Regelung wurde Systemen mit statischen nichtlinearen Kennlinien im Eingang und/oder Ausgang eines linearen Systems gewidmet. Diese Klasse von Systemen, die auch als Hammerstein-/Wiener-Systeme bezeichnet werden, treten bei realen Anlagen häufig auf. Adaptive Regler eignen sich ebenfalls gut zur Regelung solcher Regelstrecken. Es existieren bereits verschiedene Ansätze, bei denen lineare Systeme mit einer speziellen Eingangsnichtlinearität, wie beispielsweise einer Totzone (Xiong und Unbehauen, 1991b; Tao und Kokotović, 1992), einer Getriebelose (Tao und Kokotović, 1993) oder einer stückweise linearen Kennlinie (Kung und Womack, 1984; Xiong und Unbehauen, 1991a) betrachtet werden. Das prinzipielle Vorgehen bei diesen Ansätzen ist sehr ähnlich. Zunächst wird die Kenntnis der statischen nichtlinearen Kennlinie vorausgesetzt. Diese Eingangsnichtlinearität kann kompensiert werden, indem die Inverse dieser Kennlinie vor den Eingang des Systems geschaltet wird. Im Fall einer idealen Übereinstimmung ist das sich ergebende Gesamtsystem dann linear, und es kann ein linearer Standardregler entworfen werden. Sind sowohl die Parameter der statischen Nichtlinearität als auch die des linearen Teilsystems der Regelstrecke unbekannt, muss noch ein Algorithmus zur Parameteradaption des

Reglers hinzugefügt werden. Die Regelungen von Wiener-Systemen, also linearen Systemen mit einer statischen Ausgangsnichtlinearität, ergeben sich entsprechend (Al-Duwaish et al., 1996; Pajunen, 1992). Der gemeinsame Nachteil dieser „blockorientierten“ Ansätze ist, dass sie nur für eine bestimmte statische Nichtlinearität funktionieren, welche exakt bekannt sein muss und sich mit der Zeit nicht verändern darf. In der Realität sind diese Annahmen häufig nicht erfüllt. Hier entsteht die Eingangsnichtlinearität durch Überlagerung von verschiedenen Effekten, die sich in ihrer Gesamtheit nur sehr schwierig beschreiben lassen (Erken, 1999). Außerdem kann sich durch Umwelteinflüsse, wie beispielsweise Temperaturschwankungen oder Alterung, die Charakteristik der Nichtlinearität wesentlich verändern. Desweiteren werden zur *online*-Identifikation schnelle Verfahren, wie beispielsweise der rekursive *Least-Squares*-Algorithmus (RLS) benutzt. Hierdurch ergibt sich als weitere Beschränkung, dass die Nichtlinearität bezüglich der unbekannten Parameter linear beschreibbar sein muss.

Künstliche neuronale Netze (*artificial neural networks* ANN) und *Fuzzy*-Systeme werden ebenfalls häufig zur Regelung von nichtlinearen Systemen vorgeschlagen (Chen und Billings, 1992; Hagan und Demuth, 1999; Zufiria et al., 1999; Agarwal, 1997). Da mit beiden Konzepten universelle Approximatoren aufgebaut werden können, bieten sie den großen Vorteil, dass man im Prinzip kein Wissen über die Regelstreckendynamik benötigt. Gegenüber den zuvor dargestellten Verfahren werden somit keine analytischen Regelstreckenmodelle für die Synthese benötigt. Dieser häufig hervorgehobene Vorteil ist aber auch gleichzeitig der Nachteil von ANN- und Fuzzy-Reglern. Bis heute existiert kein systematisches Verfahren, welches zu einem gegebenen Problem ein optimales ANN bezüglich der Topologie und der Neuronenanordnung im Eingangsraum liefert. Hierbei unterstützende Verfahren basieren auf Heuristiken, die meistens anhand von einfachen Beispielen erstellt und belegt werden (Junge, 1999). Wie sich diese in hochdimensionalen Räumen verhalten, lässt sich nicht sicher beschreiben. Bei den meisten regelungstechnischen Problemstellungen ist wesentliches Wissen über die Regelstrecke bekannt. Dass dieses Wissen im Allgemeinen bei diesen Ansätzen nicht genutzt werden kann, stellt eine weitere Einschränkung dar.

Zusammengefasst existieren zur adaptiven Regelung von nichtlinearen Regelstrecken zum einen Methoden, für die ein ganz spezielles Regelstreckenmodell benötigt wird. Diese Verfahren funktionieren also nur zuverlässig bei Parameteränderungen, nicht aber wenn sich die Struktur selbst verändert. Zum anderen werden Reglerstrukturen mit universellen Approximatoren vorgeschlagen, bei denen das große Problem der richtigen Initialisierung besteht.

Neben einer kompakten Einführung in die Verfahren zur nichtlinearen adaptiven Regelung werden in der vorliegenden Arbeit Reglerverfahren vorgestellt, bei denen die günstigen Eigenschaften der ANN-Regelung und der modellbasierten NA-Regelung zu einer ANNNA-Regelung kombiniert werden. Die Grundidee hierbei ist, existierende Strukturkenntnis zu berücksichtigen und ANN nur dort zu verwenden, wo die Regelstrecke unbekanntes Verhalten aufweist.

Die zunächst behandelte Struktur ist ein Regler für Hammerstein-Systeme, die aus einem linearen Teilsystem und einer vorgeschalteten Nichtlinearität bestehen. Bei diesem im Weiteren als „HANNNA-Regler“ bezeichneten Regler wird angenommen, dass die Ordnung des linearen Teilsystems bekannt ist. Zur *online*-Identifikation lernt ein erstes Netz den Verlauf der Eingangsnichtlinearität und ein zweites Netz, basierend auf dem Trainingserfolg des ersten, deren inversen Verlauf. Die vorgeschlagene HANNNA-Regelung unterscheidet sich wesentlich von den bekannten Standardstrukturen, bei denen mit Hilfe von künstlichen neuronalen Netzen das Gesamtsystem invertiert wird (Hunt et al., 1992). Anders als beim HANNNA-Regler wird dort auch das dynamische Verhalten der Regelstrecke erlernt und invertiert. Dies kann nur funktionieren, wenn das System eine stabile Nulldynamik besitzt, deren Nachweis bei nichtlinearen Systemen sehr rechenintensiv ist. Durch die Dimension des Eingangsraums, die sich aus aktuellen und vergangenen Ein- und Ausgangssignalen des Systems ergibt, sind die zu trainierenden Netze komplex und sowohl das Training als auch die Einstellung der Initialisierungsparameter werden mit steigender Dimension zunehmend schwieriger. Anders ist es beim HANNNA-Regler. Hier ist die Netzkonfiguration unabhängig von der Dynamik des Systems. Da aber gerade in der Anfangsphase und bei Parametervariationen die

Schätzungen Fehler aufweisen und der damit entworfene Regler nicht mehr optimal ist, wird das Verfahren zu einem dualen Regler erweitert (D-HANNNA). Bei der dualen Regelung wird versucht, sowohl eine hohe Regelgüte als auch eine schnelle Parameterkonvergenz zu erreichen. Eine mögliche Realisierung ist beispielsweise die „vorsichtige“ Erzeugung der Stellgröße bei Parameterunsicherheit (*cautious control*) und gleichzeitig das Einführen einer Erregung.

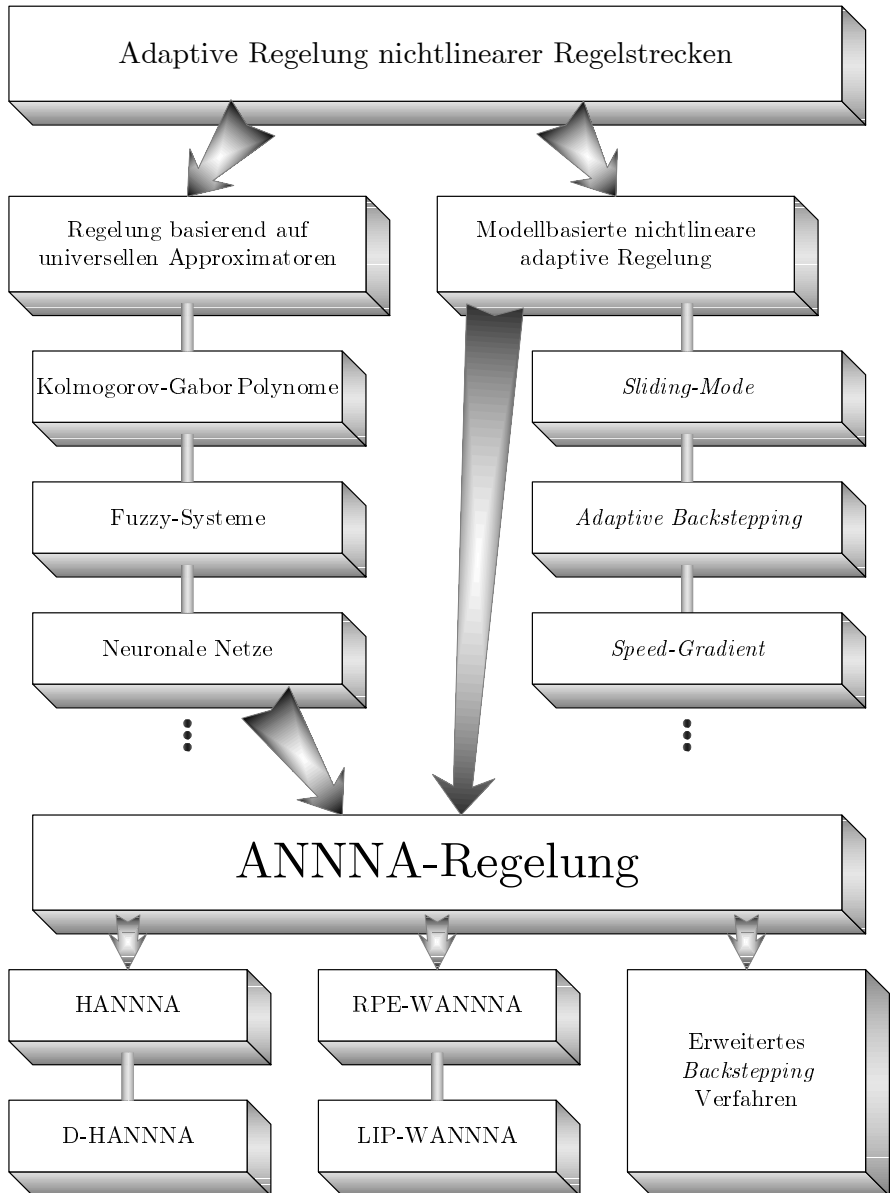
Zur Regelung eines Wiener-Systems wird basierend auf den Ergebnissen der HANNNA-Struktur der WANNNA-Regler entwickelt. Das wesentliche Problem hierbei ist die Schätzung der Modellparameter, da das Wiener-System nicht mehr linear in den Parametern (LIP) ist. Neben der Schätzung mit dem rekursiven Prädiktionsfehlerverfahren (*recursive prediction error* (RPE)) wird ein Verfahren beschrieben, bei dem das ANN zur Kompensation direkt lernt. Der Vorteil hierbei ist, dass das resultierende System wieder vom LIP-Typ ist und damit Standard-Schätzverfahren (RLS) verwendet werden können.

Bei einem weiteren Verfahren wird die *Backstepping*-Methode für PSF-Systeme durch ANN erweitert (Knohl und Unbehauen, 1996). Hierbei werden anstatt der nichtlinearen Funktionen in den einzelnen Zuständen ANN zu dessen Approximation eingesetzt. Dies hat wieder den Vorteil, dass die Netzstrukturen klein gehalten werden können und das Regelsystem flexibel auf Strukturänderungen reagieren kann. Desweiteren wird bewiesen, dass trotz dieser Erweiterung alle Systemsignale begrenzt bleiben.

Im Bild 1.1 ist eine Übersicht zur nichtlinearen adaptiven Regelung dargestellt. Hierbei wurden die fünf entwickelten Reglerstrukturen entsprechend eingeordnet. Eine solche Einteilung ist sehr schwierig und nicht eindeutig durchzuführen. Bei der hier gewählten Darstellung wurden die Ansätze mit universellen Approximatoren getrennt von den modellbasierten Methoden zur nichtlinearen adaptiven Regelung aufgeführt, da die ersten kein Strukturwissen benötigen und die zweiten eine bekannte Struktur voraussetzen. Wie man sieht, vereint die ANNNA-Regelung beide Ansätze indem ANN nur dort benutzt werden, wo kein Wissen vorliegt. Dies führt, wie im Weiteren gezeigt, zu handhabbaren Netzstrukturen.

Diese Arbeit ist in sieben Kapitel gegliedert. Nach der Einführung und der





**Bild 1.1:** Übersicht zur nichtlinearen adaptiven Regelung

---

Problemstellung im Kapitel 1 werden im Kapitel 2 die aktuellen Verfahren zur nichtlinearen adaptiven Regelung und zur Neuroregelung kurz beschrieben und deren Vor- und Nachteile diskutiert. Die Verfahren zur Regelung von Hammerstein-Systemen und deren Entwurfsschritte werden im Kapitel 3 vorgestellt. Es wird eine nicht duale und eine duale Version der Reglerstruktur beschrieben. Kapitel 4 behandelt den Reglerentwurf für Wiener-Systeme. Für die Parameteridentifikation wird zum einen das RPE-Verfahren und zum anderen ein LIP-Verfahren eingeführt. Kapitel 5 befasst sich mit der erweiterten *Backstepping*-Reglerstruktur. Anhand einer Simulation wird gezeigt, dass dieses Verfahren deutlich flexibler als die Standard-*Backstepping*-Regelung ist. Im Kapitel 6 werden simulative und experimentelle Ergebnisse der vorgestellten Reglerstrukturen diskutiert und Kapitel 7 fasst die wichtigsten Ergebnisse dieser Arbeit zusammen.

## 2 Nichtlineare adaptive Regelung

In diesem Kapitel wird eine Einführung in die bekanntesten Ansätze zur adaptiven Regelung von nichtlinearen Systemen gegeben (Knohl, 1996). Ziel ist nicht die umfassende Beschreibung aller Verfahren, sondern vielmehr sollen die Entwurfsideen und die jeweiligen Vor- und Nachteile dargestellt werden. Basierend auf den Verfahren zur adaptiven Regelung von linearen Systemen wurden nichtlineare adaptive Reglerstrukturen hauptsächlich ab 1980 entwickelt. Die große Zahl der Methoden kann wie folgt eingeteilt werden:

- Adaptive Linearisierung durch Zustandsrückführung,
- Adaptive Linearisierung durch Ausgangsrückführung,
- Adaptive *Sliding-Mode*-Regelung,
- Regelung mit dem *Speed-Gradienten*-Verfahren,
- Regelung von Systemen mit statischen Nichtlinearitäten im System-Ein-/Ausgang und
- Regelung mit künstlichen neuronalen Netzen.

Bei dieser Klassifizierung wurden Ansätze, wie beispielsweise die Fuzzy-Regelung (Berstecher, 1998) nicht berücksichtigt, die im Weiteren nicht diskutiert und zum Verständnis der neuen Reglerstrukturen nicht benötigt werden.

## 2.1 Linearisierung durch Zustandsrückführung

Die Linearisierung eines Systems durch eine geeignete Rückführung der Zustandsgrößen ist eine bekannte Methode zur Regelung von nichtlinearen Systemen (Schwarz, 1991; Isidori, 1989). Eine Erweiterung dieses Verfahrens wird seit einigen Jahren für die adaptive Regelung von nichtlinearen Regelstrecken, deren Eigenschaften (Arbeitspunkt, Parameter, Umgebung) nicht konstant sind, erfolgreich benutzt. Zur Einführung in das als *Adaptive Backstepping* bezeichnete Verfahren soll zuerst der Grundgedanke der Linearisierung anhand eines einfachen Beispiels demonstriert werden. Als Beispiel möge das durch die Differenzialgleichung

$$\dot{x} = p \cos x - x^3 + u \quad (2.1)$$

beschriebene System dienen. Hierbei ist  $x$  der Zustand des Systems,  $u$  die Eingangsgröße und  $p$  ein zunächst bekannter Parameter. Dieses System kann mit Hilfe des Stellgesetzes

$$u = -(p \cos x - x^3) - x + u_L \quad (2.2)$$

um den Ursprung stabilisiert werden. Setzt man die Stellgröße aus Gl. (2.2) in Gl. (2.1) ein, so ergibt sich

$$\dot{x} = -x + u_L \quad (2.3)$$

für das geschlossene Gesamtsystem, welches nun ein lineares System erster Ordnung mit einer Polstelle bei minus eins ist. Das nichtlineare System wird also durch die Aufschaltung obiger Stellgröße in ein lineares überführt. Dieses *neue* System kann nun mit einem beliebigen konventionellen, linearen Regler geregelt werden, wobei  $u_L$  das Stellsignal dieses Reglers darstellt. Kann man in diesem Fall das Stellgesetz direkt durch Hinschauen ermitteln, wird es doch bei komplexeren Systemen wesentlich schwieriger. Eine Vorgehensweise zur systematischen Berechnung des Stellgesetzes für komplizierte nichtlineare Systeme wird, z.B. in (Isidori, 1989) oder in (Nijmeijer und van der Schaft, 1990), beschrieben. Der gesamte Regelkreis besteht somit aus zwei Teilen. Zum einen aus der nichtlinearen dynamischen Rückkopplung, die das System linearisiert,

und zum anderen einem übergeordneten linearen Regler, mit dem die dynamischen Eigenschaften des geschlossenen Systems eingestellt werden können.

Dieser Ansatz enthält zwei Schwächen. Obwohl bei dem hier behandelten Beispiel der Term  $x^3$  für die Stabilisierung des Systems um den Ursprung hilfreich wäre, wird er genau wie  $\cos x$  neutralisiert. Es werden also bei dem Vorgehen nach Gl. (2.2) auch eventuell *nützliche* Nichtlinearitäten aus dem Systemverhalten entfernt. Das andere Problem tritt bei Parameterveränderungen  $\Delta p$  auf, durch die sich das Systemverhalten entscheidend ändern kann. Für das obige Beispiel ergibt sich das geänderte Systemverhalten zu

$$\dot{x} = \Delta p \cos x - x. \quad (2.4)$$

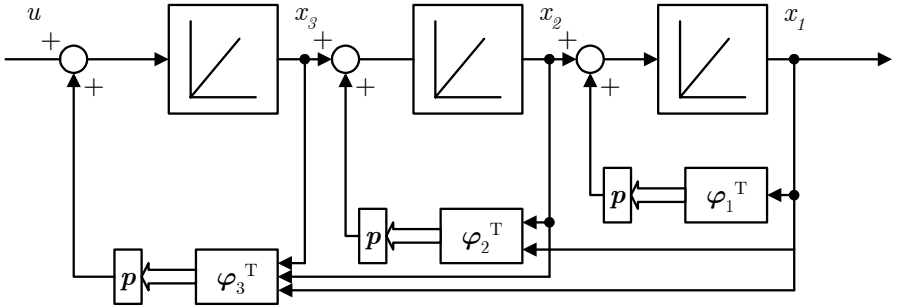
Hierbei entfernt sich die Ruhelage des geschlossenen Systems mit wachsendem  $\Delta p$  immer weiter vom Ursprung. Diese Änderungen können bei komplexeren Anlagen im Extremfall bis zur Instabilität des Systems führen. Es ist also notwendig, das Regelsystem um eine Parameteranpassung zu erweitern.

Im Allgemeinen ist dies aber nicht möglich. Jedoch kann für Systeme in *Parametric-Strict-Feedback-Form* (PSF) mit Hilfe des *Backstepping*-Algorithmus sowohl das Stellgesetz als auch die Adaptionvorschriften für die Reglerparameter ermittelt werden. Die Zustandsraumdarstellung eines PSF-Systems ist gegeben durch

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1)\mathbf{p} \\ \dot{x}_2 &= x_3 + \varphi_2^T(x_1, x_2)\mathbf{p} \\ &\vdots \\ \dot{x}_{n-1} &= x_n + \varphi_{n-1}^T(x_1, \dots, x_{n-1})\mathbf{p} \\ \dot{x}_n &= \beta(\mathbf{x})u + \varphi_n^T(\mathbf{x})\mathbf{p} \end{aligned} \quad (2.5)$$

mit  $\beta(\mathbf{x}) \neq 0, \forall \mathbf{x} \in \mathbb{R}^n$ . Die Motivation für den Namen dieser Systemstruktur kann direkt dem Diagramm im Bild 2.1 entnommen werden. Bis auf die Verbindungen der Integratoren laufen alle Signale rückwärts, sind also zurückgekoppelt.

Eine Regelstrecke, die mit Hilfe des Gleichungssystems (2.5) beschreibbar ist, kann mit dem *Backstepping*-Algorithmus (Kokotović et al., 1992) stabili-



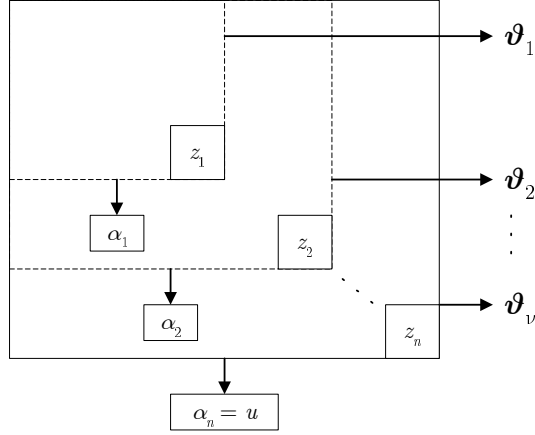
**Bild 2.1:** Blockdiagramm eines *Parametric-Strict-Feedback* Systems dritter Ordnung ( $\beta(\mathbf{x}) = 1$ )

sirt werden. Hierbei werden sowohl Stellgesetz als auch das Adaptionsgesetz rekursiv berechnet. Dieses Verfahren garantiert die globale asymptotische Stabilität aller Zustandsgrößen und aller Parameterschätzungen.

Die Anzahl der Entwurfsschritte ist gleich der Ordnung des Systems. Bei jedem Schritt wird eine neue Zustandsvariable  $z_i$ , eine Stabilisierungsfunktion  $\alpha_i$  und ein Parametervektor  $\boldsymbol{\vartheta}_i$  generiert. Dieses Vorgehen erfordert, für ein System  $n$ -ter Ordnung mit  $n_p$  unbekannten Parametern, insgesamt  $nn_p$  Parametervektoren und ist somit stark überparametriert. Möglichkeiten zur Vermeidung dieser Überparametrisierung werden von den gleichen Autoren beschrieben (Krstić et al., 1995).

Das Schema des *Recursive-Backstepping*-Algorithmus ist im Bild 2.2 dargestellt. Der Algorithmus startet in der linken, oberen Ecke, und es wird zunächst mit der direkten Methode von Ljapunow eine Regelung zur Stabilisierung des Zustandes  $z_1$  entworfen. Unter Berücksichtigung der Stabilisierungsfunktion  $\alpha_1$  wird dann der nächste Zustand  $z_2$  stabilisiert. Wieder erhält man einen Schätzvektor  $\boldsymbol{\vartheta}_2$  und eine Stabilisierungsfunktion  $\alpha_2$ , die im folgenden Schritt verwendet wird. Für alle Zustandsgrößen fortgeführt, gelangt man schließlich nach  $n$  Entwurfsschritten zum Stellgesetz  $u$  rechte, untere Ecke im Bild 2.2.

Zusammengefasst ergibt sich die folgende rekursive Rechenvorschrift eines

Bild 2.2: Schema des *Recursive-Backstepping*-Algorithmus

adaptiven Reglers für eine nichtlineare Regelstrecke in PSF-Form:

$$u = \frac{1}{\beta(\mathbf{x})} \alpha_n(\mathbf{x}, \varphi_1, \dots, \varphi_n) \quad (2.6)$$

$$\dot{\vartheta}_i = \mathbf{\Gamma} \left( \varphi_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} \varphi_j \right) z_i, \quad i = 1, \dots, n, \quad (2.7)$$

mit  $\beta(\mathbf{x}) \neq 0 \quad \forall \mathbf{x} \in \mathbb{R}$ . Hierbei ist  $\mathbf{\Gamma} = \mathbf{\Gamma}^T > 0$  eine frei zu wählende Matrix, welche die Größe der Adaptionsschritte bestimmt.

Die neuen Zustände  $z_i$  und die Stabilisierungsfunktionen  $\alpha_i, i = 1, 2, \dots, n$  sind über die Gleichungen

$$z_i = x_i - \alpha_{i-1}(x_1, \dots, x_{i-1}, \vartheta_1, \dots, \vartheta_{i-1}) \quad (2.8)$$

$$\begin{aligned} \alpha_i = & -c_i z_i - z_{i-1} - \left( \varphi_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} \varphi_j \right)^T \vartheta_i \\ & + \sum_{j=1}^{i-1} \left[ \frac{\partial \alpha_{i-1}}{\partial x_j} x_{j+1} + \frac{\partial \alpha_{i-1}}{\partial \vartheta_j} \mathbf{\Gamma} \left( \varphi_j - \sum_{k=1}^{j-1} \frac{\partial \alpha_{j-1}}{\partial x_k} \varphi_k \right) z_j \right] \end{aligned} \quad (2.9)$$

mit  $z_0 \equiv \alpha_0 \equiv 0$  definiert. Der durch die Gln. (2.6), (2.7), (2.8) und (2.9) beschriebene Regler stabilisiert die in den Gln. (2.5) dargestellte Regelstrecke um den Ursprung und garantiert zusätzlich die Beschränktheit der Zustandsgrößen  $x_1, x_2, \dots, x_n$  und der Parameterschätzvektoren  $\vartheta_1, \vartheta_2, \dots, \vartheta_n$ . Stimmen die geschätzten Parameter mit den tatsächlichen Parametern überein, kann das resultierende Gesamtsystem mit den neuen Zustandsgrößen  $z$  in der Matrixform

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \vdots \\ \dot{z}_{n-1} \\ \dot{z}_n \end{bmatrix} = \begin{bmatrix} -c_1 & 1 & 0 & \dots & 0 \\ -1 & -c_2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & -c_{n-1} & 1 \\ 0 & \dots & 0 & -1 & -c_n \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} \quad (2.10)$$

geschrieben werden. Dieses System ist für  $c_i > 0, \forall i$  stabil und es kann gezeigt werden, dass die Zustandsgrößen begrenzt bleiben und für  $t \rightarrow \infty$  gegen null gehen.

Der hier vorgestellte Algorithmus ist einfach auf Systeme mit mehreren Eingängen erweiterbar, wenn das System in der *multi-input* PSF-Beschreibung vorliegt. Das Problem der Überparametrisierung kann mit dem Einsatz der als *Tuning-Function-Design* eingeführten Methode (Krstić et al., 1995) gelöst werden. Bei diesem Ansatz ist die Anzahl der geschätzten Parameter gleich der Zahl der unbekannten Parameter.

Zusammengefasst steht mit diesem Verfahren ein leistungsfähiger Regelalgorithmus zur Verfügung, mit dem nichtlineare Systeme linearisiert und um den Ursprung stabilisiert werden können. Hierbei werden eventuelle Parameterschwankungen und -unsicherheiten ausgeglichen.

## 2.2 Linearisierung durch Ausgangsrückführung

Für den Entwurf des im Abschnitt 2.1 beschriebenen adaptiven Reglers wurde die Messbarkeit aller Zustandsgrößen vorausgesetzt. Dies ist in der Praxis meistens nicht realisierbar. Weit realistischer ist es, dass von einer Anlage nur



die Eingangssignale, die Regelgrößen und höchstens einige wenige Zustandsgrößen gemessen werden können. Für diesen Fall wurde ein Algorithmus zur Synthese eines adaptiven Reglers vorgestellt, der nur Informationen über die Eingangssignale und Regelgrößen benötigt. Bei diesem Algorithmus werden die unbekannten Zustandsgrößen mit Hilfe eines Beobachters geschätzt und diese Zustände für den Reglerentwurf mit dem *Backstepping*-Algorithmus verwendet. Es ist natürlich möglich, Zustände, die direkt verfügbar sind, mit in die Berechnung aufzunehmen. Dabei würde sich eine Kombination des Verfahrens aus Abschnitt 2.1 und dem im Folgenden beschriebenen Verfahren ergeben. Um den Grundgedanken dieses Verfahrens (Kanellakopoulos et al., 1993, 1990, 1991c) zu verdeutlichen, wird auf diese Kombination nicht weiter eingegangen. Bei der Verwendung von Beobachtern in Regelkreisen für lineare Systeme wird gefordert, dass der Beobachtungsfehler schnell abfällt und dann vernachlässigbar klein bleibt. Das Vertrauen auf das *certainty-equivalence*(CE)-Prinzip bei nichtlinearen Systemen kann zur Instabilität des Regelkreises führen. Aus diesem Grund darf der Beobachtungsfehler bei diesem Entwurf nicht vernachlässigt werden. Er wird als externe Störung berücksichtigt. Durch Hinzufügen eines speziellen Dämpfungsterms (*nonlinear damping*) in das Regelgesetz kann dann das Verschwinden des Fehlers erzwungen werden. Eine Klasse von nichtlinearen Regelstrecken, für die ein solcher Beobachterentwurf möglich ist, sind die Parametric-Output-Feedback-Systeme, auch als POF-Systeme bezeichnet, welche die folgende Darstellung aufweisen:

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \varphi_{0,1}(y) + \sum_{j=1}^{n_p} p_j \varphi_{j,1}(y) \\
 \dot{x}_2 &= x_3 + \varphi_{0,2}(y) + \sum_{j=1}^{n_p} p_j \varphi_{j,2}(y) \\
 &\vdots \\
 \dot{x}_{\rho-1} &= x_{\rho} + \varphi_{0,\rho-1}(y) + \sum_{j=1}^{n_p} p_j \varphi_{j,\rho-1}(y)
 \end{aligned} \tag{2.11}$$

$$\begin{aligned}
\dot{x}_\rho &= x_{\rho+1} + \varphi_{0,\rho}(y) + \sum_{j=1}^{n_p} p_j \varphi_{j,\rho}(y) + b_m \beta(y) u \\
&\vdots \\
\dot{x}_n &= \varphi_{0,n}(y) + \sum_{j=1}^{n_p} p_j \varphi_{j,n}(y) + b_0 \beta(y) u \\
y &= x_1.
\end{aligned}$$

Die Parameter  $p_j, j = 1, \dots, n_p$  und  $b_i, i = 0, \dots, m$  sind unbekannt oder unsicher. Charakteristisch bei der Systembeschreibung nach den Gln. (2.11) ist die lineare Abhängigkeit der Zustandsgrößen  $x_i$  untereinander und, dass die Nichtlinearitäten  $\varphi_{i,j}$  ausschließlich eine Funktion des Ausgangssignals  $y$  sind.

Ist die Regelstrecke durch eine POF-Form beschrieben, so kann nun unter den Voraussetzungen

1. das Vorzeichen von  $b_m$  ist bekannt,
2. das Polynom  $B(s) = b_m s^m + \dots + b_1 s + b_0$  ist ein Hurwitz-Polynom,
3.  $\beta(y) \neq 0 \quad \forall y \in \mathbb{R}$  und
4. der Sollwert  $w$  und seine ersten  $\rho$  Ableitungen sind bekannt, endlich und zusätzlich ist die  $\rho$ -te Ableitung stückweise stetig,

eine adaptive Regelung angegeben werden, welche die Signale des geschlossenen Systems begrenzt und die Regelabweichung

$$e = w - y \tag{2.12}$$

für  $t \rightarrow \infty$  verschwinden lässt. Für den Entwurf wird zunächst der Beobachter der ungemessenen Zustandsgrößen benötigt. Hierfür schreibt man Gl.(2.11) in Vektorform

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \boldsymbol{\varphi}_0(y) + \sum_{j=1}^{n_p} p_j \boldsymbol{\varphi}_j(y) + \mathbf{b}\beta(y)u \tag{2.13}$$

$$y = \mathbf{c}^T \mathbf{x} \tag{2.14}$$

mit

$$\mathbf{A} = \begin{bmatrix} 0 & & \\ \vdots & \mathbf{I}_{(n-1) \times (n-1)} & \\ 0 & \dots & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{0}_{(\rho-1) \times 1} \\ b_m \\ \vdots \\ b_0 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1) \times 1} \end{bmatrix} \quad (2.15)$$

und

$$\boldsymbol{\varphi}_j = \begin{bmatrix} \varphi_{j,1} \\ \varphi_{j,2} \\ \vdots \\ \varphi_{j,n} \end{bmatrix}, \quad 0 \leq j \leq n_p. \quad (2.16)$$

Der Zustandsvektor  $\mathbf{x}$  dieses Systems wird mit den drei Hilfsgrößen  $\boldsymbol{\xi}_0, \boldsymbol{\xi}_1$  und  $\mathbf{v}_j$  geschätzt. Diese Größen sind durch die drei Filterbeziehungen

$$\begin{aligned} \dot{\boldsymbol{\xi}}_0 &= \mathbf{A}_0 \boldsymbol{\xi}_0 + \mathbf{k}y + \boldsymbol{\varphi}_0(y) \\ \dot{\boldsymbol{\xi}}_j &= \mathbf{A}_0 \boldsymbol{\xi}_j + \boldsymbol{\varphi}_j(y), & 1 \leq j \leq n_p \\ \dot{\mathbf{v}}_j &= \mathbf{A}_0 \mathbf{v}_j + \mathbf{e}_{n-j} \beta(y) & 1 \leq j \leq m \end{aligned} \quad (2.17)$$

definiert. Mit dem Vektor  $\mathbf{e}_i$  als  $i$ -tem Einheitsvektor und dem Verstärkungsvektor  $\mathbf{k}$ , der so gewählt wird, dass die Eigenwerte der Matrix  $\mathbf{A}_0 = \mathbf{A} - \mathbf{k}\mathbf{c}^T$  nur in der linken  $s$ -Halbebene liegen. Subtrahiert man Gl (2.17) von Gl. (2.13), so folgt daraus das Fehlersystem

$$\dot{\boldsymbol{\varepsilon}} = \mathbf{A}_0 \boldsymbol{\varepsilon} \quad (2.18)$$

mit

$$\boldsymbol{\varepsilon} \triangleq \mathbf{x} - \left( \boldsymbol{\xi}_0 + \sum_{j=1}^{n_p} p_j \boldsymbol{\xi}_j + \sum_{j=1}^m b_j \mathbf{v}_j \right). \quad (2.19)$$

Da die Matrix  $\mathbf{A}_0$  stabil gewählt wurde, konvergiert das Fehlersystem exponentiell gegen null. Das bedeutet, dass das System

$$\boldsymbol{\xi}_0 + \sum_{j=1}^{n_p} p_j \boldsymbol{\xi}_j + \sum_{j=1}^m b_j \mathbf{v}_j \quad (2.20)$$

eine Schätzung für den Zustandsvektor  $\mathbf{x}$  darstellt. Ähnlich wie im Abschnitt 2.1 kann nun unter Berücksichtigung der Beobachtungssignale nach Gl. (2.20) ein adaptiver Regler mit dem rekursiven *Backstepping*-Verfahren (Krstić et al., 1995; Krstić und Kokotović, 1993; Kanellakopoulos et al., 1991b) entworfen werden. Da die Rechnung und der Beweis der Stabilität des Reglersystems sehr komplex sind, werden nur die Ergebnisse aufgeführt und auf deren Herleitung an dieser Stelle verzichtet. Das nichtlineare System nach den Gln. (2.11) kann bei Erfüllung der vier Annahmen von Seite 17 und mit Hilfe des Beobachters nach den Gln. (2.17) mit dem folgenden Stellgesetz stabilisiert werden:

$$u = \frac{1}{\beta(y)} \left[ \alpha_\rho - v_{m,\rho+1} + \vartheta_{1,1} w^{(\rho)} \right], \quad (2.21)$$

$$\dot{\mathbf{z}}_1 = \text{sgn}(b_m) \mathbf{\Gamma} \left[ \boldsymbol{\omega}_1(y, \bar{\boldsymbol{\xi}}^{(2)}, \bar{\mathbf{v}}^{(2)}, \bar{\mathbf{w}}^{(1)}) - \dot{w} \mathbf{e}_1 \right] z_1, \quad (2.22)$$

$$\dot{\mathbf{z}}_2 = \mathbf{\Gamma} \left[ \boldsymbol{\omega}_2(y, \bar{\boldsymbol{\xi}}^{(2)}, \bar{\mathbf{v}}^{(2)}, \bar{\boldsymbol{\vartheta}}^{(2)}, \bar{\mathbf{w}}^{(1)}) + z_1 \mathbf{e}_{n_p+m+1} \right] z_2 \quad \text{und} \quad (2.23)$$

$$\dot{\mathbf{z}}_i = \mathbf{\Gamma} \boldsymbol{\omega}_i(y, \bar{\boldsymbol{\xi}}^{(i)}, \bar{\mathbf{v}}^{(i)}, \bar{\boldsymbol{\vartheta}}^{(i-1)}, \bar{\mathbf{w}}^{(i-1)}) z_i, \quad i = 3, \dots, \rho. \quad (2.24)$$

Hierbei ist  $\mathbf{e}_i$  wieder der Einheitsvektor im  $\mathbb{R}^{n_p+m+1}$  und  $z_i, \boldsymbol{\omega}_i, \alpha_i, i = 1, \dots, \rho$  sind mit den Entwurfparametern  $c_i > 0, d_i > 0, i = 1, \dots, \rho$  und der Verstärkungsmatrix  $\mathbf{\Gamma} > 0$  durch die folgenden Gleichungen gegeben:

$$z_1 = y - w, \quad (2.25)$$

$$z_i = v_{m,i} - \alpha_{i-1} \left( y, \bar{\boldsymbol{\xi}}^{(i)}, \bar{\mathbf{v}}^{(i)}, \bar{\boldsymbol{\vartheta}}^{(i)}, \bar{\mathbf{w}}^{(i-1)} \right) - \vartheta_{1,1} w^{(i)}, \quad i = 2, \dots, \rho, \quad (2.26)$$

$$\alpha_1 = -\boldsymbol{\vartheta}_1^T \boldsymbol{\omega}_1, \quad (2.27)$$

$$\begin{aligned} \alpha_2 = & -c_2 z_2 - \vartheta_{2,n_p+m+1} z_1 - d_2 \left( \frac{\partial \alpha_1}{\partial y} \right)^2 z_2 + \frac{\partial \alpha_1}{\partial y} [\xi_{0,2} + \varphi_{0,1}(y)] - \boldsymbol{\vartheta}_2^T \boldsymbol{\omega}_2 \\ & + k_2 v_{m,1} + \frac{\partial \alpha_1}{\partial \xi_0} [\mathbf{A}_0 \boldsymbol{\xi}_0 + \mathbf{k} y + \varphi_0(y)] + \sum_{j=1}^{n_p} \frac{\partial \alpha_1}{\partial \boldsymbol{\xi}_j} [\mathbf{A}_0 \boldsymbol{\xi}_j + \varphi_j(y)] \quad (2.28) \\ & + \sum_{j=0}^m \frac{\partial \alpha_1}{\partial \mathbf{v}_j} \mathbf{A}_0 \mathbf{v}_j + \left( \frac{\partial \alpha_1}{\partial \boldsymbol{\vartheta}_1} + \dot{w} \mathbf{e}_1^T \right) \text{sgn}(b_m) \mathbf{\Gamma} [\boldsymbol{\omega}_1 - \dot{w} \mathbf{e}_1] z_1 + \frac{\partial \alpha_1}{\partial w} \dot{w}, \end{aligned}$$

$$\begin{aligned}
\alpha_i = & -c_i z_i - z_{i-1} - d_i \left( \frac{\partial \alpha_{i-1}}{\partial y} \right)^2 z_i + \frac{\partial \alpha_{i-1}}{\partial y} [\xi_{0,2} + \varphi_{0,1}(y)] - \boldsymbol{\vartheta}_i^T \boldsymbol{\omega}_i \\
& + k_i v_{m,1} + \frac{\partial \alpha_{i-1}}{\partial \xi_0} [\mathbf{A}_0 \boldsymbol{\xi}_0 + \mathbf{k}y + \boldsymbol{\varphi}_0(y)] + \sum_{j=1}^{n_p} \frac{\partial \alpha_{i-1}}{\partial \xi_j} [\mathbf{A}_0 \boldsymbol{\xi}_j + \boldsymbol{\varphi}_j(y)] \\
& + \sum_{j=0}^m \frac{\partial \alpha_{i-1}}{\partial \mathbf{v}_j} \mathbf{A}_0 \mathbf{v}_j + \left( \frac{\partial \alpha_{i-1}}{\partial \boldsymbol{\vartheta}_1} + w^{(i-1)} \mathbf{e}_1^T \right) \text{sgn}(b_m) \boldsymbol{\Gamma} [\boldsymbol{\omega}_1 - \dot{w} \mathbf{e}_1] z_1 \\
& + \frac{\partial \alpha_{i-1}}{\partial \boldsymbol{\vartheta}_2} \boldsymbol{\Gamma} [\boldsymbol{\omega}_2 + z_1 \mathbf{e}_{n_p+m+1}] z_2 + \sum_{j=3}^{i-1} \frac{\partial \alpha_{i-1}}{\partial \boldsymbol{\vartheta}_j} \boldsymbol{\Gamma} \boldsymbol{\omega}_j z_j + \sum_{j=1}^{i-2} \frac{\partial \alpha_{i-1}}{\partial w^{(j)}} w^{(j+1)}, \\
& i = 3, \dots, \rho.
\end{aligned} \tag{2.29}$$

Die Hilfsgrößen  $\boldsymbol{\omega}_i, \bar{\boldsymbol{\xi}}^{(i)}, \bar{\mathbf{v}}^{(i)}, \bar{\boldsymbol{\vartheta}}^{(i)}, \bar{\mathbf{w}}^{(i)}$  sind Signalvektoren, die aus den Beobachter- und Systemsignalen wie folgt zusammengesetzt werden:

$$\begin{aligned}
\boldsymbol{\omega}_1^T = & [c_1 z_1 + d_1 z_1 + \xi_{0,2} + \varphi_{0,1}, \varphi_{1,1} + \xi_{1,2}, \dots, \varphi_{n_p,1} + \xi_{n_p,2}, \\
& v_{0,2}, \dots, v_{m-1,2}], \tag{2.30}
\end{aligned}$$

$$\boldsymbol{\omega}_i^T = -\frac{\partial \alpha_{i-1}}{\partial y} [\varphi_{1,1} + \xi_{1,2}, \dots, \varphi_{n_p,1} + \xi_{n_p,2}, v_{0,2}, \dots, v_{m,2}], \quad i = 2, \dots, \rho, \tag{2.31}$$

$$\bar{\boldsymbol{\xi}}^{(i)} = [\xi_{0,1}, \dots, \xi_{0,i}, \dots, \xi_{n_p,1}, \dots, \xi_{n_p,i}], \quad i = 1, \dots, \rho - 1, \tag{2.32}$$

$$\bar{\mathbf{v}}^{(i)} = [v_{0,1}, \dots, v_{0,i}, \dots, v_{m-1,1}, \dots, v_{m-1,i}, v_{m,1}, \dots, v_{m,i-1}], \quad i = 1, \dots, \rho, \tag{2.33}$$

$$\bar{\boldsymbol{\vartheta}}^{(i)} = [\vartheta_1^T, \dots, \vartheta_i^T], \quad i = 1 \dots, \rho, \tag{2.34}$$

$$\bar{\mathbf{w}}^{(i)} = [w, \dot{w}, \dots, w^{(i)}], \quad i = 1, \dots, \rho. \tag{2.35}$$

Der Nachteil des hier beschriebenen Reglers ist die starke Überparametrisierung. Für  $n_p + m + 1$  unbekannte Größen werden insgesamt  $(n_p + m + 1) \times \rho$  Schätzungen berechnet. Zusätzlich werden für den Beobachter weitere drei

Hilfssignale benötigt. Dies macht deutlich, dass der Rechenaufwand für diese Regelung sehr hoch ist. Genau hier setzen viele Verbesserungsvorschläge, wie beispielsweise die  $\kappa$ -*Adaptive* Regelung (Kanellakopoulos et al., 1993), an. Mit dieser gelingt es, die Überparametrisierung vollständig zu vermeiden. Eine andere Möglichkeit zur Verbesserung des Regelalgorithmus bietet der Beobachterentwurf, der unabhängig vom Reglerentwurf gemacht werden kann. Hier könnte man z.B. Beobachter mit besserem Konvergenzverhalten verwenden.

Häufig werden Regelstreckenmodelle aus dem vorgegebenen Aufbau und den einzelnen physikalischen Eigenschaften eines Systems hergeleitet. Die Koordinaten des Modells sind somit vorgegeben und man wird in der Regel keine Systembeschreibung unmittelbar nach den Gln. (2.11) erhalten. Es ist aber möglich, Regelstrecken in der allgemeinen Zustandsraumdarstellung

$$\dot{\mathbf{x}} = \underbrace{\mathbf{f}_0(\mathbf{x}) + \sum_{i=1}^{n_p} \mathbf{p}_i \mathbf{f}_i(\mathbf{x})}_{\mathbf{f}(\mathbf{x})} + \underbrace{\left[ \mathbf{g}_0 + \sum_{i=1}^{n_p} \mathbf{p}_i \mathbf{g}_i(\mathbf{x}) \right]}_{\mathbf{g}(\mathbf{x})} u \quad (2.36)$$

unter bestimmten Bedingungen mit Hilfe einer Koordinatentransformation in die POF-Form zu transformieren. Hierbei sind  $\mathbf{f}_i(\mathbf{x})$  und  $\mathbf{g}_i(\mathbf{x})$  glatte Vektorfelder in einer Umgebung des Ursprungs  $\mathbf{x}$  mit  $\mathbf{f}_i(\mathbf{0}) = \mathbf{0}$  und  $\mathbf{g}_i(\mathbf{0}) \neq \mathbf{0}$ ,  $0 \leq i \leq n_p$  und  $u$  ist die Stellgröße. Glatte Vektorfelder sind Felder, bei denen alle partiellen Ableitungen beliebiger Ordnung existieren. Für die PSF-Form aus Abschnitt 2.1 kann dies ebenfalls gezeigt werden (Krstić et al., 1995)

Die in den Abschnitten 2.1 und 2.2 beschriebenen Reglerstrukturen bieten die Möglichkeit, nichtlineare Systeme mit Parameterunsicherheiten einem Sollwert  $w$  nachzuführen. Hierfür werden entweder die Zustände oder die Ausgangssignale der Regelstrecke benötigt. Die *Adaptive Backstepping*-Methode ist ein mächtiges Syntheseverfahren zum Entwurf nichtlinearer adaptiver Regler, mit denen der geschlossene Regelkreis global asymptotisch stabilisierbar ist. Nachteile dieser Methode, die eine Implementierung an einer realen Regelstrecke wesentlich erschweren, sind:

- Der sehr hohe Entwurfs- und Rechenaufwand bei der Reglersynthese.
- Das System muss linear in den unbekannten Parametern sein.

- Die nichtlinearen Vektorfunktionen  $\varphi_i$  müssen bekannt und konstant sein, damit der Beweis der Stabilität geführt werden kann. Dies ist aber gerade bei realen Anlagen nicht gegeben.
- Im Allgemeinen sind die Parameterschätzungen nicht erwartungstreu.
- Für die Stabilität des Regelkreises wird unbegrenzte Stellenergie angenommen.
- Für die große Zahl von Reglerparametern  $c_i, d_i$  und  $\mathbf{F}$  existiert keine Systematik zur Einstellung.

## 2.3 Adaptive *Sliding-Mode*-Regelung

Die *Sliding-Mode*-Regelung ist ein etabliertes Verfahren zur robusten Regelung. Die Grundlagen dieser Methode wurden zunächst für Eingrößensysteme erarbeitet (Emeljanov, 1969; Utkin, 1992) und dann für Mehrgrößensysteme erweitert (Slotine und Li, 1991). Nichtlineare Regelstrecken mit begrenzten Parameterschwankungen und/oder Modellunsicherheiten können mit diesem Ansatz effizient geregelt werden. Mit wachsender Parameterunsicherheit/Modellunsicherheit wird aber die erzielbare Regelgüte immer geringer. Genau an dieser Grenze ist es sinnvoll, die konventionelle *Sliding-Mode*-Regelung um eine Parameteradaption zu erweitern.

Das Konzept der *Sliding-Mode*-Regelung beruht auf dem Gedanken, dass es einfacher ist, mehrere Systeme niedriger Ordnung als ein komplexes System hoher Ordnung zu regeln. Der Grundgedanke dieser Methode soll anhand eines nichtlinearen SISO<sup>1</sup>-Systems mit der Zustandsraumdarstellung

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{b}(\mathbf{x})u, \\ y &= x_n\end{aligned}\tag{2.37}$$

verdeutlicht werden. Hier ist  $y$  das Ausgangssignal,  $u$  das Stellsignal,  $\mathbf{x}$  der Zustandsvektor und  $\mathbf{f}(\mathbf{x})$  und  $\mathbf{b}(\mathbf{x})$  sind im Allgemeinen nichtlineare Funktionen. Als Ziel einer Regelung soll nun erreicht werden, dass der Zustandsvektor

---

<sup>1</sup>Single-Input Single-Output

$\mathbf{x}$  einem vorgegebenen Zielvektor  $\mathbf{x}_d$  über eine bestimmte Zeitspanne folgt. Es soll also der Fehlervektor

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_d = [e, \dot{e}, \dots, e^{(n-1)}]^T \quad (2.38)$$

für  $t \rightarrow \infty$  zu null werden. Um diese Aufgabe zu lösen, wird eine Gleitfläche oder auch *Sliding*-Fläche im  $\mathbb{R}^n$  definiert, die beispielsweise durch die Funktion

$$s(e, t) = \left( \frac{d}{dt} + \lambda \right)^{n-1} e = \left[ \sum_{\nu=0}^{n-1} \binom{n-1}{\nu} \lambda^{n-1-\nu} \frac{d^\nu}{dt^\nu} \right] e \quad (2.39)$$

mit  $s(e, t) = 0$  und  $\lambda > 0$  beschrieben wird. Der Klammerausdruck in Gl.(2.39) kann mit Hilfe der Binomialkoeffizienten ausgewertet werden. Es ist aber zu beachten, dass nicht das Ergebnis der Differenziation potenziert wird, sondern hier ist die  $n$ -te Ableitung einzusetzen. Für ein System zweiter Ordnung ergibt sich beispielsweise

$$s = \dot{e} + \lambda e \quad (2.40)$$

oder für  $n = 3$

$$s = \ddot{e} + 2\lambda\dot{e} + \lambda^2 e. \quad (2.41)$$

Die Gl. (2.39) ist eine lineare Differenzialgleichung, deren Lösung für  $s = 0$

$$e(t) = \left( \sum_{\nu=0}^{n-2} c_\nu t^\nu \right) e^{-\lambda t} \quad (2.42)$$

mit  $t > 0$  ist. Man sieht, dass gilt

$$e_\infty = \lim_{t \rightarrow \infty} e(t) = 0. \quad (2.43)$$

Mit  $e_\infty$  verschwinden auch sämtliche zeitliche Ableitungen des Fehlers  $e(t)$ , so dass

$$\mathbf{e}_\infty = \mathbf{0} \quad (2.44)$$

resultiert. Der Wert von  $s$  ist somit ein direktes Maß für die Regelabweichung  $e$  und deren  $n$  Ableitungen und es folgt aus der Beschränktheit von  $s$  die



Beschränktheit von  $e$ . Das Problem der Folgeregelung  $n$ -ter Ordnung ist damit auf ein Stabilisierungsproblem erster Ordnung reduziert worden. Wählt man nun die Stellgröße  $u$  so, dass die Stabilitätsbedingung entsprechend Ljapunow

$$\dot{V}(s, t) = \frac{1}{2} \frac{\partial}{\partial t} s^2 \leq -\kappa |s| \quad (2.45)$$

erfüllt ist, erreicht das Regelsystem, selbst mit  $\mathbf{x}(0) \neq \mathbf{x}_d(0)$ , nach einer endlichen Zeit die Gleitfläche und der Fehler wird zu null (Slotine und Li, 1991). Hierbei ist  $\kappa$  eine positive Konstante. Interpretiert man  $s^2$  als Maß für den Abstand des Systems von der Gleitfläche, so erzwingt Gl. (2.45) ein stetiges Abnehmen des Abstands. Ist die Fläche einmal erreicht, verbleibt das System auf ihr. Die Dynamik des geschlossenen Regelkreises im Gleitzustand ist dann durch die Differenzialgleichung gemäß Gl. (2.39) definiert und somit unabhängig von der Dynamik der Regelstrecke. Dadurch wird das Regelsystem weder von Parameterschwankungen noch von Modellierungsfehlern beeinflusst. Dem Entwurf der Schaltfunktion  $s(e, t)$ , sie kann sowohl linear als auch nicht-linear sein, kommt große Bedeutung zu. Sie legt das dynamische Verhalten des geschlossenen Regelsystems im Gleitzustand fest, wobei der Existenzbeweis des Gleitzustands für beliebige Schaltfunktionen sehr komplex werden kann.

Nach der Bestimmung der Schaltfunktion wird der eigentliche Regler entworfen. Bei der Methode der äquivalenten Regelung teilt sich dieser Entwurf in zwei Schritte auf, und das Stellgesetz ergibt sich zu

$$u = u_e + u_n. \quad (2.46)$$

Im Englischen wird  $u_e$  als *equivalent control*, also dem äquivalenten Stellgesetz und  $u_n$  als *non-continuous control*, der unstetige Teil des Stellgesetzes, bezeichnet. Befindet sich das Regelsystem zum Zeitpunkt  $t = 0$  auf der Gleitfläche, und sind die Systemparameter konstant, dann gilt  $s(0) = 0$ , und es muss der kontinuierliche Teil des Stellgesetzes  $u_e$  ermittelt werden, der das System auf der Gleitfläche hält. Dies ist gleichbedeutend/äquivalent mit der Forderung

$$\dot{s} = 0. \quad (2.47)$$

Für den Fall von Parameterunsicherheiten muss das System zunächst auf die Gleitfläche gebracht werden. Hierzu erweitert man den Regler um ein Zwei-

punktglied, das bei jedem Seitenwechsel des Regelsystems bezüglich der Gleitfläche das Vorzeichen ändert. Das erweiterte Stellgesetz kann dann wie folgt geschrieben werden:

$$u = u_e + k \operatorname{sgn}(s). \quad (2.48)$$

Hierbei ist  $\operatorname{sgn}$  die Signumfunktion:

$$\operatorname{sgn}(s) = \begin{cases} +1 & s > 0 \\ -1 & s < 0 \end{cases} \quad (2.49)$$

und  $k$  eine Konstante, die mit steigender Parameterunsicherheit immer größer gewählt werden muss. Hieraus ergeben sich immer größere Stellwerte  $u$ , die einen unnötig hohen Energieaufwand bedeuten. Sind die Schwankungen zu stark, ist schließlich die *Sliding-Mode*-Regelung nicht mehr sinnvoll einzusetzen. Ein grundsätzliches Problem der *Sliding-Mode*-Regelung ist das Schalten des Reglers mit hoher Frequenz (*Rattern* oder *chattering*) in der Nähe der Gleitfläche. Dieses Phänomen macht den Einsatz des unmodifizierten *Sliding-Mode*-Reglers in vielen Bereichen unmöglich, in denen Stellglieder nicht sehr häufig umgeschaltet werden dürfen oder können.

In der Literatur findet man verschiedene Ansätze zur Erweiterung des *Sliding-Mode*-Verfahrens, mit denen es dann auch für die Regelung von Systemen mit relativ großen Parameterschwankungen geeignet ist.

Beispielsweise beschreiben Llanes-Santiago et al. (1993) für kontinuierliche, nichtlineare Systeme der Form

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \boldsymbol{\Theta}) + \mathbf{g}(\mathbf{x}, \boldsymbol{\Theta})u, \\ y &= h(\mathbf{x}, \boldsymbol{\Theta}), \end{aligned} \quad (2.50)$$

eine adaptive Erweiterung des *Sliding-Mode*-Reglers. Beim Entwurf dieses Reglers werden die in der Gleitfläche  $s = 0$  auftretenden Parameter  $\boldsymbol{\Theta}$  durch ihre Schätzwerte  $\hat{\boldsymbol{\Theta}}$  ersetzt. Das damit auf der Basis von Gl. (2.45) entwickelte Stellgesetz ist nun ebenfalls nur noch von den Schätzungen und nicht mehr von den tatsächlichen Parametern abhängig. Um die geschätzten Parameter zu adaptieren, bilden die Autoren mit dem Parameterfehler  $\boldsymbol{\Theta} - \hat{\boldsymbol{\Theta}}$  die Abweichung

der Gleitfläche  $s - \hat{s}$ . Mit Hilfe einer geeigneten Ljapunow-Funktion zur Stabilisierung des Fehlersystems um den Ursprung kann dann das Adaptionsgesetz für die Parameter angegeben werden.

Für den Entwurf eines adaptiven *Sliding-Mode*-Reglers für nichtlineare, diskrete Systeme wählen Won und Hedrick (1995) ein ganz anderes Verfahren. Sie nehmen anstatt einer einzigen Gleitfläche für das gesamte System mehrere kleinere Flächen. Durch das Einführen von virtuellen Hilfsregelgrößen wird jede einzelne *Sliding*-Funktion nacheinander stabilisiert und ein Adaptionsgesetz für die Parameterschätzungen berechnet.

Die adaptive *Sliding-Mode*-Regelung ist ein interessanter Ansatz zur Regelung von nichtlinearen Systemen mit großen Parameterunsicherheiten. Dies wird erreicht, indem der an sich schon robuste Ansatz durch eine Parameterschätzung erweitert wird. Ein weiterer Vorteil ist die, im Vergleich zu den in den Abschnitten 2.1 und 2.2 vorgestellten Verfahren, relativ einfache mathematische Beschreibung der *Sliding-Mode*-Regelung.

## 2.4 Regelung mit dem *Speed-Gradienten-Verfahren*

Das *Speed-Gradienten*(SG)-Verfahren wurde von Krasovskii (1976) als Optimierungsalgorithmus vorgestellt und später von Fradkov (1979) zur adaptiven Regelung eingesetzt. Der Ansatz ist — wie bereits Anfangs erwähnt — hauptsächlich von russischsprachigen Autoren entwickelt worden und findet in der westlichen (englischsprachigen) Welt kaum Beachtung. Anders als bei den bisher vorgestellten adaptiven Reglern ist dieser Ansatz zunächst sehr allgemein gehalten und kann somit an nahezu beliebige Systemdarstellungen angepasst werden. Hierbei können spezielle Eigenschaften der verschiedenen Darstellungen beim Entwurf des Reglers berücksichtigt werden.

Betrachtet wird eine nichtlineare, zeitvariante Regelstrecke in Zustandsraumdarstellung

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \boldsymbol{\eta}, t), \quad (2.51)$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}, \boldsymbol{\eta}, t). \quad (2.52)$$

Bei diesem System ist  $\mathbf{x} \in \mathbb{R}^n$  der Zustandsvektor,  $\mathbf{y} \in \mathbb{R}^r$  der Ausgangsvektor,  $\boldsymbol{\eta} \in \mathbb{R}^m$  der Parametervektor und  $\mathbf{f}(\mathbf{x}, \boldsymbol{\eta}, t)$  eine glatte Vektorfunktion. Für den von außen beeinflussbaren Parametervektor wurde in Gl. (2.51) der Buchstabe  $\boldsymbol{\eta}$  gewählt, weil er neben dem in der Regelungstechnik üblicherweise verwendeten Stellvektor  $\mathbf{u}$  auch noch weitere Systemparameter enthalten kann.

Ziel der Regelung dieser Regelstrecke mit dem *Speed-Gradienten-Verfahren* ist die Minimierung eines Gütefunktionalis der Form

$$I(\mathbf{x}, \mathbf{y}, \boldsymbol{\eta}, t) \geq 0 \quad \forall t > 0. \quad (2.53)$$

In der Wahl des Gütefunktionalis ist man zunächst frei. Es können also spezielle Wünsche und Eigenschaften, wie z.B. Energieaufwand, Geschwindigkeit oder Stellgrößenbegrenzungen, in die Gütefunktion integriert werden. Typische Gütefunktionale, die sich für die verschiedensten Aufgabenstellungen eignen, sind z.B.

1. das allgemeine Integralkriterium

$$I = \int_0^t L(\mathbf{x}, \mathbf{y}, \boldsymbol{\eta}, \tau) d\tau \stackrel{!}{=} \text{Min} \quad \text{oder}$$

2. das energieoptimale Integralkriterium

$$I = \int_0^t \mathbf{u}^T(\tau) \mathbf{R}(\tau) \mathbf{u}(\tau) d\tau \stackrel{!}{=} \text{Min},$$

wobei  $\mathbf{R}(t)$  eine symmetrische, positiv definite Matrix ist.

Weitere Funktionale sind in (Unbehauen, 1988) vorgeschlagen.

Die Aufgabe des Entwurfs besteht nun in der Bestimmung des Adaptionsgesetzes

$$\dot{\boldsymbol{\eta}} = \mathbf{f}(\mathbf{x}, \mathbf{y}, \boldsymbol{\eta}, t) \quad (2.54)$$

so, dass das geschlossene System, beschrieben durch die Gln. (2.51) und (2.54), das gewählte Gütekriterium möglichst gut erfüllt und Parameterschwankungen

und äußere Störungen ausgeregelt werden. Wie schon erwähnt, enthält der Vektor  $\boldsymbol{\eta}$  alle von außen veränderbaren Systemgrößen. Dies können zum einen die Stellgröße  $\boldsymbol{u}$  aber auch andere Regler- oder Systemparameter sein. Es kann nun gezeigt werden (Fradkov, 1979), dass mit der Vorschrift

$$\dot{\boldsymbol{\eta}} = -\boldsymbol{\Gamma} \nabla_{\boldsymbol{\eta}} \dot{I}, \quad (2.55)$$

die geforderten Eigenschaften des geschlossenen Systems erreicht werden. Mit der in Gl. (2.55) dargestellten, einfachsten Form des SG-Verfahrens lässt sich auch der Name und die Philosophie dieser Methode erklären.

Die verstellbaren Systemparameter werden proportional zum negativen Gradienten der zeitlichen Änderung des Gütefunktional d.h. der Gütefunktionalgeschwindigkeit geändert.

Der Hintergrund dieser Philosophie ist intuitiv leicht zu erklären. Um das gewählte Gütefunktional zu minimieren, sollte  $\boldsymbol{\eta}$  in Richtung des negativen Gradienten gewählt werden. Da das Gütefunktional nicht zwingend von  $\boldsymbol{\eta}$  abhängig ist, kann versucht werden, das Ansteigen von  $I$ , also  $\dot{I}$ , mit Hilfe von  $\boldsymbol{\eta}$  zu verringern. Dies kann eventuell zu  $\dot{I} < 0$  führen, woraus sich eine Verringerung von  $I$  ergibt.

Der SG-Algorithmus kann in verschiedenen Formen beschrieben werden. In der differenziellen Form ergibt sich das SG Gesetz zu

$$\frac{d}{dt} [\boldsymbol{\eta} + \boldsymbol{\psi}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\eta}, t)] = -\boldsymbol{\Gamma} \nabla_{\boldsymbol{\eta}} \dot{I} \quad (2.56)$$

oder in integraler Form zu

$$\boldsymbol{\eta} = -\boldsymbol{\psi}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\eta}, t) - \int_0^t \boldsymbol{\Gamma} \nabla_{\boldsymbol{\eta}} \dot{I} d\tau. \quad (2.57)$$

Hierbei ist  $\boldsymbol{\psi}$  eine Vektorfunktion, welche die Bedingung  $\boldsymbol{\psi}^T \nabla_{\boldsymbol{\eta}} \dot{I} \geq 0$  erfüllen muss. Für die Anfangsbedingungen gilt

$$\boldsymbol{\eta}(0) = \mathbf{0} \quad \text{und} \quad \boldsymbol{\psi}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\eta}, 0) = \mathbf{0}. \quad (2.58)$$

Der Stabilitätsnachweis sowohl des einfachen als auch des erweiterten SG-Gesetzes kann leicht mit Hilfe einer geeigneten Ljapunow-Funktion geführt werden.

Um die Verwendung des SG-Algorithmus zu vertiefen, soll das folgende einfache Beispiel dienen, bei dem die lineare SISO-Regelstrecke, beschrieben durch die Zustandsraumdarstellung

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}u \\ y &= \mathbf{c}^T \mathbf{x},\end{aligned}\tag{2.59}$$

mit einem Eingang  $u$ , einem Ausgang  $y$  und dem Zustandsvektor  $\mathbf{x} \in \mathbb{R}^n$  dem Sollwert  $w$  nachgeführt werden soll. Hierzu wählt man beispielsweise das Gütefunktional

$$I = \frac{1}{2}e^2\tag{2.60}$$

mit  $e = w - y$ . Da  $u$  die einzig beeinflussbare Größe ist, wird die Substitution  $u \rightarrow \eta$  durchgeführt und der Geschwindigkeitsgradient von Gl. 2.56 ergibt sich mit

$$\dot{I} = e\dot{e} = e(\dot{w} - \dot{y}) = e(\dot{w} - \mathbf{c}^T(\mathbf{A}\mathbf{x} + \mathbf{b}\eta))\tag{2.61}$$

zu

$$\nabla_{\eta} \dot{I} = -e\mathbf{c}^T \mathbf{b}.\tag{2.62}$$

Unter der Annahme, dass für das Skalarprodukt

$$\mathbf{c}^T \mathbf{b} \neq 0\tag{2.63}$$

gilt und mit den Definitionen  $\psi = k_p \frac{\nabla_{\eta} \dot{I}}{\mathbf{c}^T \mathbf{b}} = -k_p e$  und  $k_i = \Gamma \mathbf{c}^T \mathbf{b}$  gelangt man zur differenziellen Darstellung des Stellgesetzes

$$\frac{d}{dt}(\eta - k_p e) = k_i e.\tag{2.64}$$

Durch Integration und Rücksubstitution  $\eta \rightarrow u$  erhält man schließlich den bekannten PI-Regler

$$u(t) = k_p e + k_i \int_0^t e(\tau) dt.\tag{2.65}$$

Dieses Beispiel zeigt die Anwendung des SG-Verfahrens zur Ermittlung eines Stellgesetzes für ein lineares System. Durch die allgemeine Definition des Verfahrens ist es möglich, nach der gleichen Vorgehensweise beliebige von außen zugängliche Parameter zu optimieren, beispielsweise bei der adaptiven Regelung eines Roboterarms, bei dem Parameterschwankungen auftreten (Fradkov und Stotsky, 1992). Sowohl das Stellgesetz als auch die Parameteradaption werden mit Hilfe des SG-Algorithmus berechnet. Eine weitere Anwendung des SG-Verfahrens ist die Transformation eines nichtlinearen Systems in ein passives und damit ein BIBO-stabiles<sup>2</sup> System (Seron et al., 1995). Passiv ist ein System mit dem Eingangsvektor  $\mathbf{u} \in \mathbb{R}^r$ , dem Ausgangsvektor  $\mathbf{y} \in \mathbb{R}^m$  und dem Zustandsvektor  $\mathbf{x} \in \mathbb{R}^n$ , wenn eine reelle Funktion  $V(\mathbf{x})$  mit  $V(0)$  existiert, welche die Ungleichung

$$\int_0^t \mathbf{u}^T(\tau) \mathbf{y}(\tau) d\tau \geq V(\mathbf{x}(t)) - V(\mathbf{x}_0) \quad (2.66)$$

erfüllt.

Ein Problem des SG-Verfahrens ist die starke Empfindlichkeit gegenüber Störungen und unmodellierten Parameterunsicherheiten. Schon kleine Störungen können ein unbegrenztes Anwachsen der kontrollierbaren Größen  $\boldsymbol{\eta}$  und damit die Instabilität des Gesamtsystems hervorrufen. Um diese Gefahr zu begrenzen, wurde das erweiterte Gütefunktional

$$I_g = I + \frac{\lambda}{2} \int_0^t \boldsymbol{\eta}^T \boldsymbol{\eta} d\tau \quad (2.67)$$

vorgeschlagen (Fradkov, 1979), mit dem sich das Adaptionsgesetz

$$\dot{\boldsymbol{\eta}} = -\boldsymbol{\Gamma}(\nabla_{\boldsymbol{\eta}} \dot{I} + \lambda \boldsymbol{\eta}) \quad (2.68)$$

ergibt. Diese Erweiterung kann je nach Betrachtung als negative Rückkopplung der variablen Parameter oder auch als Gewichtung des Energieaufwands gedeutet werden. Der offensichtliche Nachteil dieser Methode ist der Geschwindigkeitsverlust der Regelung. Hier scheint aber eine größere Robustheit des Gesamtsystems gegenüber Störungen die wichtigere Eigenschaft zu sein.

---

<sup>2</sup>Bounded-Input Bounded-Output

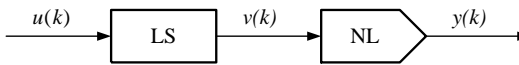
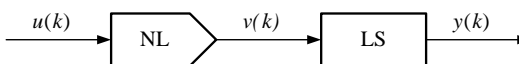
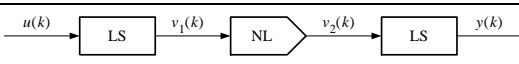
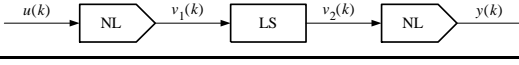
Gleichung (2.63) ist nur erfüllbar, falls der Polüberschuss der Regelstrecke null oder eins ist (Fradkov et al., 1999). Schon für ein  $PT_2S$ -System kann mit dem SG-Verfahren kein Regelgesetz gefunden werden (Mao, 1997). Für nicht-lineare Regelstrecken folgt daraus, dass dieses Verfahren nur für Systeme mit einem relativen Grad bis eins anwendbar ist. Zusätzlich muss bei Regelstrecken mit unbekannten Parametern die *extended-matching*-Bedingung erfüllt sein, bei der sich zwischen Stellgröße und Parameter maximal ein Integrator befinden darf.

Bis auf die zuvor erwähnten Einschränkungen zeigt sich, dass das SG-Verfahren sehr flexibel an die verschiedensten Systemdarstellungen und Aufgabenstellungen angepasst werden kann. Häufig lassen sich existierende Regelungen für spezielle Problemstellungen auf die gleiche Weise mit Hilfe des SG Verfahrens berechnen. Sie sind also im Prinzip Spezialfälle des SG-Verfahrens (Fradkov, 1979).

## **2.5 Regelung von Systemen mit statischen Nichtlinearitäten**

Bei den bisher beschriebenen adaptiven Regelungen handelte es sich um relativ allgemeine Ansätze, die für eine große Klasse von Systemen geeignet sind. Im Folgenden sollen Verfahren zur adaptiven Regelung von speziellen Systemen in Blockstruktur, die aus einem linearen Teilsystem und einer statischen Nichtlinearität bestehen, beschrieben werden. Unter der Voraussetzung, dass ein gegebenes Problem mit Hilfe dieser speziellen Blockstrukturen beschrieben werden kann, bieten diese Ansätze den Vorteil der relativ einfachen Parametridentifikation. Typisch für diese Klasse von Systemen ist das Hammerstein- und das Wiener-System. Beim Hammerstein-System ist der statische, nichtlineare Block vor dem Eingang eines linearen Systems angeordnet. Diese Konfiguration kommt beispielsweise zustande, wenn ein Stellglied eine große Totzone oder starke Haftreibung aufweist. Weitere unstetige Kennlinien, die bei Regelungssystemen häufig auftreten, sind die Getriebelose, die Hysterese oder eine stückweise lineare Funktion. Werden die beiden Blöcke, also lineares Teilsystem



Name	Blockstruktur
Wiener-System	
Hammerstein-System	
Wiener-Hammerstein-S.	
Hammerstein-Wiener-S.	

**Tabelle 2.1:** Nichtlineare blockorientierte Modellstrukturen

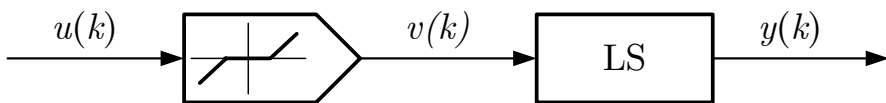
und statische Nichtlinearität, vertauscht, ergibt sich das Wiener-System. Diese Konfiguration tritt beispielsweise bei der pH-Wertregelung auf. Hierbei kann das Regelsystem durch ein lineares System zweiter Ordnung gefolgt von einer statischen Kennlinie, der Titrationskurve, beschrieben werden. Durch weitere Kombinationen von linearen Systemen und nichtlinearen Kennlinien ergeben sich die in Tabelle 2.1 zusammengefassten Blockstrukturen. Hierbei bedeutet LS dynamisches, lineares System und NL statische, nichtlineare Kennlinie. Die dargestellten Systeme können im Zeitbereich sowohl diskret, wie in Tabelle 2.1 geschehen, als auch kontinuierlich beschrieben werden.

Für die angegebenen Blockstrukturen existieren bereits eine Reihe von Ansätzen zur adaptiven Regelung, bei denen verschiedene nichtlineare Kennlinien, wie beispielsweise eine Totzone (Xiong und Unbehauen, 1991b; Tao und Kokotović, 1992), einer Getriebelose (Tao und Kokotović, 1993; Tao, 1995) oder einer stückweise stetigen Kennlinie (Kung und Womack, 1984; Xiong und Unbehauen, 1991a; Pajunen, 1992), betrachtet werden.

Das prinzipielle Vorgehen bei der Reglersynthese ist bei diesen Verfahren sehr ähnlich. Zunächst wird die Kenntnis der statischen, nichtlinearen Kennlinie vorausgesetzt. Um den störenden Einfluss dieser Nichtlinearität zu kompensieren, wird ein nichtlinearer Block mit der inversen Kennlinie derselben im Regelkreis in einer Reihenschaltung hinzugefügt. Für den Fall der vollständigen Kompensation der wirklichen Nichtlinearität, ist das sich ergebende Regelsys-

tem dann linear, und es kann ein linearer Standardregler entworfen werden. Sind sowohl die Parameter der statischen Nichtlinearität als auch die des linearen Teilsystems der Regelstrecke unbekannt, muss noch eine Parameteradaption des Reglers durchgeführt werden.

Um dieses Prinzip zu verdeutlichen, wird im Folgenden als *Beispiel* eine adaptive Regelung für ein Hammerstein-System mit Totzone besprochen. Im Bild 2.3 ist das Blockschaltbild des betrachteten Systems dargestellt. Zur Vereinfachung wird angenommen, dass das lineare Teilsystem vollständig bekannt ist. Die Parameter der Totzone sollen unbekannt sein oder mit der Zeit variieren.



**Bild 2.3:** Hammerstein-System mit Totzone

Der Reglerentwurf teilt sich in zwei Schritte auf. Im ersten Schritt entwirft man einen Regler für das lineare Teilsystem, wobei der Einfluss der Nichtlinearität unberücksichtigt bleibt. Dann wird im zweiten Schritt die Inverse der Nichtlinearität entworfen, die im Fall der idealen Übereinstimmung alle nichtlinearen Effekte aufhebt. Da aber die Parameter der Totzone unbekannt sind, muss noch ein Adaptionsgesetz für die Parameter der Inversen entworfen werden.

Angenommen, die Regelstrecke habe die Darstellung

$$Y(s) = G(s)V(s), \quad v(t) = D(u(t)), \quad G(s) = k_p \frac{B(s)}{A(s)}, \quad (2.69)$$

und für das lineare System gelte

1.  $G(s)$  ist minimalphasig,  $B(s)$  und  $A(s)$  sind normierte Polynome,
2. der relative Grad  $n^*$  von  $G(s)$  ist bekannt,
3. die obere Grenze  $\bar{n}$  des Nenners  $A(s)$  ist bekannt und

4. das Vorzeichen der Verstärkung  $k_p$  ist bekannt.

Dann ist das Ziel der Regelung, den Fehler  $e(t) = y_m(t) - y(t)$  mit

$$Y_m(s) = G_m(s)W(s) \quad (2.70)$$

für  $t \rightarrow \infty$  auszuregeln. In Gl. (2.70) bezeichnet  $w(t)$  die Führungsgröße der Regelung und  $y_m(t)$  das Ausgangssignal eines stabilen linearen Modellvergleichs-Systems mit der einfachen Übertragungsfunktion

$$G_m(s) = \frac{1}{A_m(s)}. \quad (2.71)$$

$A_m(s)$  ist ein Polynom der Ordnung  $n^*$ , dessen Nullstellen in der linken komplexen  $s$ -Halbebene liegen. Unter der Vernachlässigung der Totzone kann die Regelung beispielsweise mit dem Stellgesetz

$$U_d(s) = \boldsymbol{\Theta}_1^T \frac{\mathbf{a}(s)}{\Lambda(s)} U_d(s) + \boldsymbol{\Theta}_2^T \frac{\mathbf{a}(s)}{\Lambda(s)} Y(s) + \Theta_3 W(s) \quad (2.72)$$

aufgebaut werden (Tao und Tian, 1995).  $u_d(t)$  ist das Signal am Eingang der Inversen,  $\Lambda(s)$  ist ein frei wählbares, normiertes Polynom der Ordnung  $n$ ,  $\mathbf{a}(s) = [1, s, \dots, s^{n-1}]^T$  und die Entwurfsparameter in Form der Parametervektoren  $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2 \in \mathbb{R}^n$  und  $\Theta_3 \in \mathbb{R}$  müssen die Beziehungen

$$\boldsymbol{\Theta}_1^T \mathbf{a}(s) A(s) + \boldsymbol{\Theta}_2^T \mathbf{a}(s) k_p B(s) = \Lambda(s) (A(s) - B(s) A_m(s)) \quad (2.73)$$

und

$$\Theta_3 = \frac{1}{k_p} \quad (2.74)$$

erfüllen. Im nächsten Schritt muss die Inverse  $D^{-1}(u_d(t))$  der Nichtlinearität  $D(u(t))$  berechnet werden. Für die im vorliegenden Beispiel angenommene Totzone des nichtlinearen Systems und deren Inverse ergeben sich die im Bild 2.4 und 2.5 dargestellten Kennlinien, die mathematisch wie folgt beschrieben werden können:

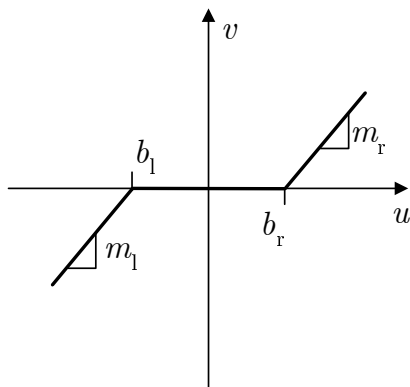


Bild 2.4: Totzone

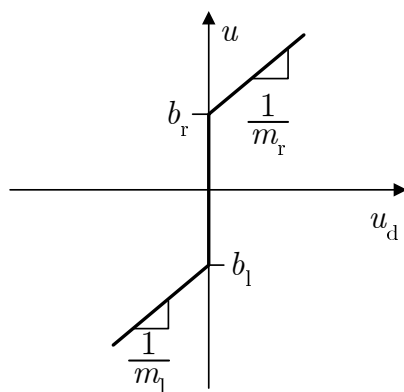


Bild 2.5: Inverse Totzone

- Die Totzone

$$v(t) = \begin{cases} m_r(u(t) - b_r) & \text{für } u(t) \geq b_r \\ 0 & \text{für } b_r > u(t) > b_l \\ m_l(u(t) - b_l) & \text{für } b_l \geq u(t) \end{cases} \quad (2.75)$$

- Die Inverse der Totzone

$$u(t) = \begin{cases} \frac{1}{m_r}(u_d(t) + m_r b_r) & \text{für } u_d(t) \geq 0 \\ \frac{1}{m_l}(u_d(t) + m_l b_l) & \text{für } 0 > u_d(t), \end{cases} \quad (2.76)$$

wobei  $m_l$  die Steigung der linken Geraden und  $b_l$  der zugehörige Achsenabschnitt ist sowie  $m_r$  und  $b_r$  die entsprechenden Parameter für die rechte Seite sind. Stimmen die Parameter der Totzone mit denen der Inversen überein, ist das Produkt  $D^{-1} \times D$  gleich eins. Da die Totzone Eingangssignale im Bereich  $b_r > u(t) > b_l$  in den Ursprung abbildet, existiert keine rechtsseitige Inverse. Aus diesem Grund kann die angegebene Inverse nicht zur Regelung eines Systems mit der Totzone im Ausgang verwendet werden kann. Die Struktur der zuvor beschriebenen Regelung ist im Bild 2.6 dargestellt.

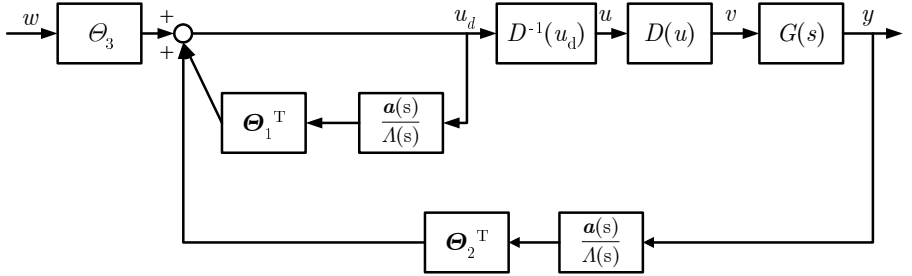


Bild 2.6: Adaptive Regelungsstruktur mit inverser Totzone

Für den Fall, dass Parameterunsicherheiten existieren, schlagen Tao und Kokotović (1994) mit der Annahme  $m_r = m_l = m$  das Adaptionsgesetz

$$\dot{\Theta}_0(t) = -\frac{\mathbf{\Gamma}\zeta(t)\varepsilon(t)}{1 + \zeta(t)^T\zeta(t) + \xi^2(t)} - \mathbf{\Gamma}\sigma(\Theta_0(t))\Theta_0(t) \quad (2.77)$$

für den Parametervektor  $\Theta_0(t) = [m, mb_r, mb_l]$ , mit den unbekannten Parametern der Totzone, vor. Hierbei gilt  $\mathbf{\Gamma}^T = \mathbf{\Gamma} > 0$ ,

$$\varepsilon(t) = e(t) - \xi(t), \quad (2.78)$$

$$\xi(t) = \Theta_0^T(t)\zeta(t) - \mathcal{L}^{-1} \left\{ H(s)\mathcal{L} \left\{ \Theta_0^T(t)\omega_0(t) \right\} \right\}, \quad (2.79)$$

$$\zeta(s) = H(s)\omega_0(s), \quad (2.80)$$

$$\sigma(\Theta_0(t)) = \begin{cases} 0 & \text{wenn } \|\Theta_0(t)\| < M_0, \\ \sigma_0 \left( \frac{\|\Theta_0(t)\|}{M_0} - 1 \right) & \text{wenn } M_0 \leq \|\Theta_0(t)\| < 2M_0, \\ \sigma_0 & \text{wenn } \|\Theta_0(t)\| \geq 2M_0 \end{cases} \quad (2.81)$$

und

$$H(s) = \Theta_3^{-1}G_m(s) \left( 1 - \frac{\Theta_1^T a(s)}{\Lambda(s)} \right). \quad (2.82)$$

In diesem Adaptionsgesetz ist  $\omega_0(t)$  ein Signalvektor mit

$$\omega_0(t) = \begin{pmatrix} u(t) \\ \chi \\ 1 - \chi \end{pmatrix}, \quad \chi(t) = \begin{cases} 1 & \text{für } u_d(t) \geq 0 \\ 0 & \text{sonst} \end{cases}. \quad (2.83)$$

$\sigma_0$  ist ein freier Entwurfsparameter und  $M_0$  ist eine Abschätzung des maximalen Betrags von  $\Theta_0$ , so dass  $\|\Theta_0\| < M_0$  gilt.

Diese Regelungsstruktur garantiert auch bei starken Parameterschwankungen das Verschwinden des Fehlers  $e(t)$  für  $t \rightarrow \infty$  und die Beschränktheit aller Systemsignale. Für den Fall  $m_l \neq m_r \neq m$  existiert dieser Beweis zwar noch nicht, G. Tao und P. K. Kokotović stellen aber in ihren Simulationen auch hierfür die Konvergenz des Fehlers fest.

Die vorgestellte Regelungsstruktur kann prinzipiell auf alle Systeme mit statischen Nichtlinearitäten angewandt werden, für die eine mathematische Beschreibung der Inversen existiert. Sind die Parameter des linearen Teilsystems ebenfalls unbekannt, wird anstelle des festen Reglers ein adaptiver Ansatz zur Einstellung von  $\Theta_1$  und  $\Theta_2$  gewählt.

Zusammengefasst können mit diesem Verfahren lineare Systeme mit Nichtlinearitäten im Ein- und/oder Ausgang und unbekannten Parameter geregelt werden. Hierbei teilt sich der Entwurf in zwei Schritte auf, die unabhängig voneinander ausgeführt werden können. Sowohl für den Regler als auch bei der Adaption der Parameter der Inversen können noch andere als die hier vorgestellten Algorithmen verwendet werden.

Der gemeinsame Nachteil dieser Ansätze ist, dass sie nur für eine bestimmte statische Nichtlinearität funktionieren, welche exakt bekannt sein muss und sich mit der Zeit nicht ändern darf. In der Realität sind diese Annahmen häufig nicht erfüllt. Hier entsteht die Nichtlinearität durch Überlagerung von verschiedenen Effekten, die sich in ihrer Gesamtheit nur sehr schwierig beschreiben lassen. Außerdem kann sich durch Umwelteinflüsse, wie beispielsweise Temperaturschwankung oder Alterung, die Charakteristik der Nichtlinearität wesentlich verändern. Ein weiteres Problem ist, dass zur *online*-Identifikation schnelle Verfahren, wie beispielsweise der rekursive Least-Squares-Schätzung (RLS) benutzt werden. Hierdurch ergibt sich als weitere Beschränkung, dass

die Nichtlinearität bezüglich der unbekannten Parameter linear beschreibbar sein muss.

## 2.6 Reglerstrukturen mit künstlichen neuronalen Netzen

Die in den vorhergehenden Abschnitten vorgestellten Regler basieren auf parametrischen Modellen, bei denen die Kenntnis über die Struktur des Systems vorausgesetzt wird und nur die Parameter unbekannt sind. Eine Alternative zu diesen Modellen besteht in der Modellierung einer Regelstrecke durch künstliche neuronale Netze, im englischen Sprachraum auch als *artificial neural networks* (ANN) bezeichnet, die allgemeine Approximatoren darstellen (Girosi und Poggio, 1990). Diese Eigenschaft sichert, dass durch ANN jede nichtlineare Funktion, welche die Bolzano-Weierstrass Eigenschaft (Bronstein et al., 1999) erfüllt, beliebig genau approximiert werden kann. Es liegt deshalb nahe, ANN bei der Regelung von nichtlinearen Systemen zu verwenden. Aus diesem Ansatz wurden Reglerstrukturen entwickelt, bei denen im Regelkreis das Regelstreckenmodell und/oder dessen Inverse benötigt wird. Basierend auf Regelverfahren, die zunächst für lineare Systeme vorgeschlagen wurden, sind dies beispielsweise

- das Kopieren eines existierenden Reglers,
- die adaptive Regelung mit Vergleichsmodell und
- die Regelung mit internem Prozessmodell.

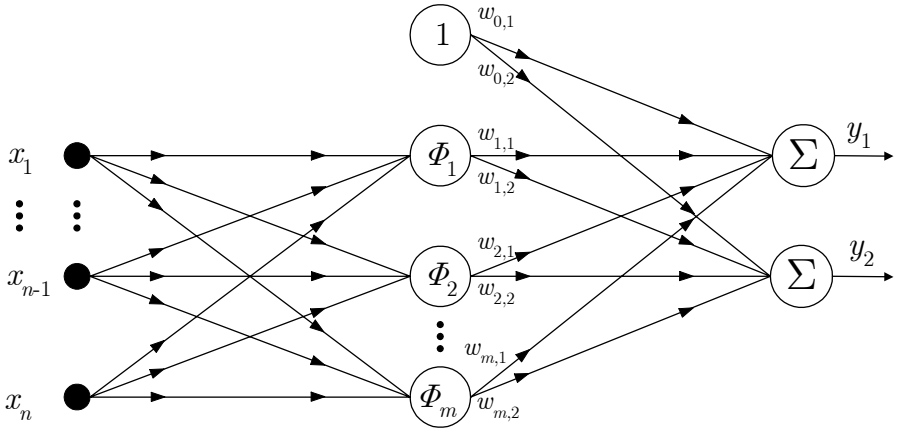
Die Grundgedanken dieser Verfahren werden im Folgenden in komprimierter Form vorgestellt (vgl. Ng (1997)). Bei diesen Verfahren werden ANN zur Modellierung eines nichtlinearen, dynamischen Prozesses eingesetzt. Dabei muss die zeitliche Ordnung des Systems bekannt sein. Zusätzlich muss bei den Reglerstrukturen, welche die Prozessinverse enthalten, die Nulldynamik des Prozesses stabil sein. Dieser Nachweis ist rechenintensiv und kann bei unbekannten Systemen im Allgemeinen nicht geführt werden.

Ein wichtiger Aspekt beim Entwurf von Reglern mit ANN ist die Auswahl der Netzstruktur. Sie beeinflusst die Wahl des Lernalgorithmus zur Optimierung der Netze, die erzielbare Regelgüte, die Rechenzeit und die Geschwindigkeit der Parameterkonvergenz. Aus diesem Grund befasst sich ein großer Teil der Forschung im Bereich der ANN mit dieser Fragestellung, weshalb die Anzahl der Strukturen und Strukturvariationen sehr groß ist. Es muss leider festgestellt werden, dass bis heute kein Verfahren existiert, mit dem ein systematischer Entwurf bezüglich der Netztopologie und der Neuronenanzahl in vertretbarer Zeit möglich wäre. Die bedeutendste und meist angewandte Grundstruktur ist das mehrschichtige Perzeptronnetz (engl. *Multilayer Perceptron (MLP)*). In Verbindung mit dem bekannten *Backpropagation*-Lernalgorithmus wird diese Struktur sowohl zur Identifikation als auch zur Regelung von nicht-linearen Systemen eingesetzt (Hagan und Demuth, 1999). MLP-Netze sind nichtlinear in den unbekannten Parametern (Gewichten), was bei der Optimierung zu langen Trainingszeiten führt. Ein ähnliches Problem tritt bei der Verwendung von ANN mit interner Dynamik auf (Knohl, 1994b). Neben der nichtlinearen Parametrisierung muss zusätzlich die Stabilität gesichert werden. Besonders die nichtlineare Parametrisierung ist bei der adaptiven Regelung unerwünscht, weshalb sich die Radial-Basis-Funktionen-Netze in diesem Bereich immer stärker durchsetzen. Nach Festlegung der Formparameter und der Neuronenzahl sind diese Netze linear in den Parametern und können somit zuverlässig und rasch optimiert werden.

### 2.6.1 Das Radial-Basis-Funktionen(RBF)-Netz

Die RBF-Netze sind eine relativ neue Klasse von ANN, die zur Identifikation von nichtlinearen Systemen benutzt werden. In zahlreichen Untersuchungen (Chen und Billings, 1992; Pottmann und Seborg, 1992; Knohl, 1994a) wurde gezeigt, dass sie diese Aufgabe hervorragend lösen. Ihre Struktur, im Bild 2.7 dargestellt, ist sehr einfach. Sie bestehen aus einer Eingangsschicht, die den Vektor  $\mathbf{x} = [x_1 x_2 \dots x_n]^T$  der Eingangssignale  $x_i, i = 1, \dots, n$  puffert, einer versteckten Schicht, welche die Neuronen mit dem Vektor  $\Phi$  der RBF-Aktivierungsfunktionen  $\Phi_j, j = 1, \dots, m$  enthält und einer Ausgangsschicht, in





**Bild 2.7:** Das Radial-Basis-Funktionen(RBF)-Netz für  $n$  Eingänge und zwei Ausgänge

der die gewichteten Aktivierungen summiert werden.

Die mathematische Ein-Ausgangsbeschreibung eines RBF-Netzes ist durch die Beziehung

$$y_l(k) = w_{0,l} + \sum_{j=1}^m w_{j,l} \Phi_j(\|\mathbf{x}(k) - \mathbf{c}_j\|) \quad (2.84)$$

gegeben. Hierbei ist  $y_l(k)$  das  $l$ -te Ausgangssignal,  $\mathbf{x}(k)$  der Eingangsvektor,  $w_{j,l}, j = 0, \dots, m$  sind die Netzgewichte und  $\mathbf{c}_j$  sind die Zentren der RB-Funktionen. Mit  $\|\mathbf{x}(k) - \mathbf{c}_j\| = r$  als Abstandsmaß eines Eingangsvektors von einem Zentrum ist  $\Phi_j(r)$  die RB-Funktion. In der Literatur werden häufig die folgenden Funktionen vorgeschlagen:

- Gauß-Funktion

$$\Phi(r) = \frac{1}{\sqrt{2\pi\mathfrak{w}}} \exp\left(-\frac{r^2}{2\mathfrak{w}^2}\right), \quad \mathfrak{w} > 0, \quad (2.85)$$

- Reziprok-Multiquadratische-Funktion

$$\Phi(r) = \frac{1}{\sqrt{(r^2 + \mathfrak{w})}}, \quad \mathfrak{w} > 0, \quad (2.86)$$

- Multiquadratische-Funktion

$$\Phi(r) = \sqrt{(r^2 + \mathfrak{w})}, \quad \mathfrak{w} \geq 0, \quad (2.87)$$

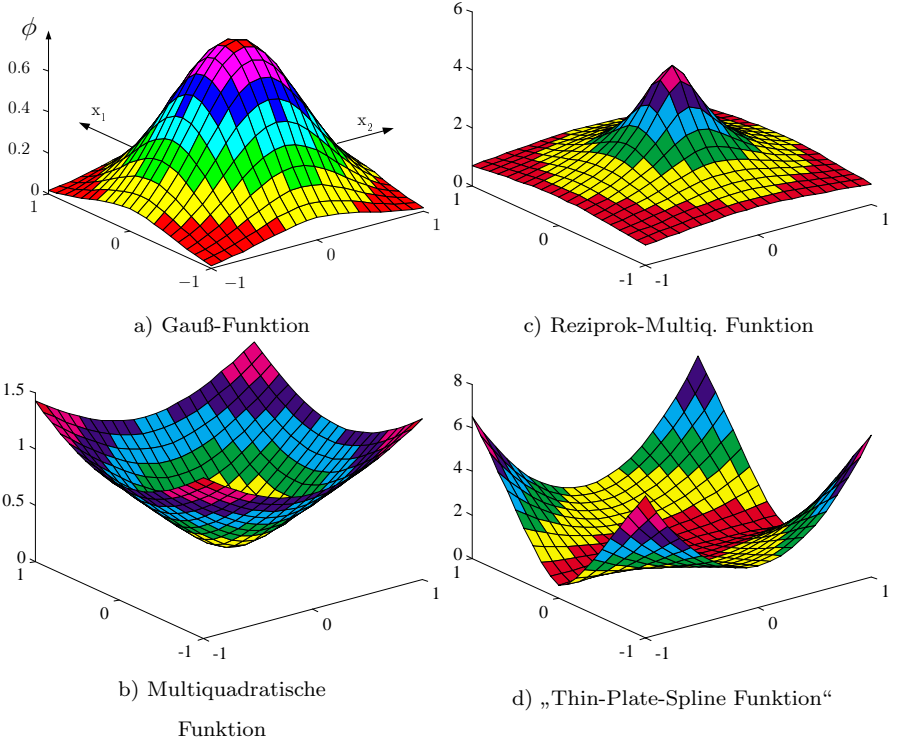
- „Thin-Plate-Spline“-Funktion

$$\Phi(r) = r^2 \log(r), \quad (2.88)$$

wobei  $\mathfrak{w}$  ein Formparameter ist, der die spezielle Form der Funktion festlegt. Im Bild 2.8 sind für  $n = 2$  typische Verläufe  $\Phi = \Phi(x_1, x_2)$  dargestellt. Erwähnenswert ist, dass ein Neuron mit einer der oberen Aktivierungsfunktionen (Teilbilder a) und c)) nur für einen lokalen Eingangsbereich aktiv ist. Die unteren Aktivierungsfunktionen (Teilbilder b) und d)) hingegen beeinflussen den Ausgang des Netzes im gesamten Eingangsraum. Im Allgemeinen kann ein Polynom auf der rechten Seite von Gl. (2.84) addiert werden. Mit dieser Erweiterung können unerwünschte Oszillationen zwischen den einzelnen Zentren vermieden werden.

Für eine gegebene Menge von  $N$  Eingangsvektoren  $\mathbf{x}(k) \in \mathbb{R}^n, k = 1, \dots, N$  und den zugehörigen Ausgangswerten  $y_l(k) \in \mathbb{R}, k = 1, \dots, N$  kann gezeigt werden, dass ein Gewichtsvektor  $\mathbf{w}$  existiert, der die Summe der Fehlerquadrate

$$I = \sum_{k=1}^N \left( y_l(k) - (w_{0,l} + \sum_{j=1}^m w_{j,l} \Phi_j(\|\mathbf{x}(k) - \mathbf{c}_j\|)) \right)^2 \quad (2.89)$$



**Bild 2.8:** Typische Radial-Basis-Funktionen im  $\mathbb{R}^2$

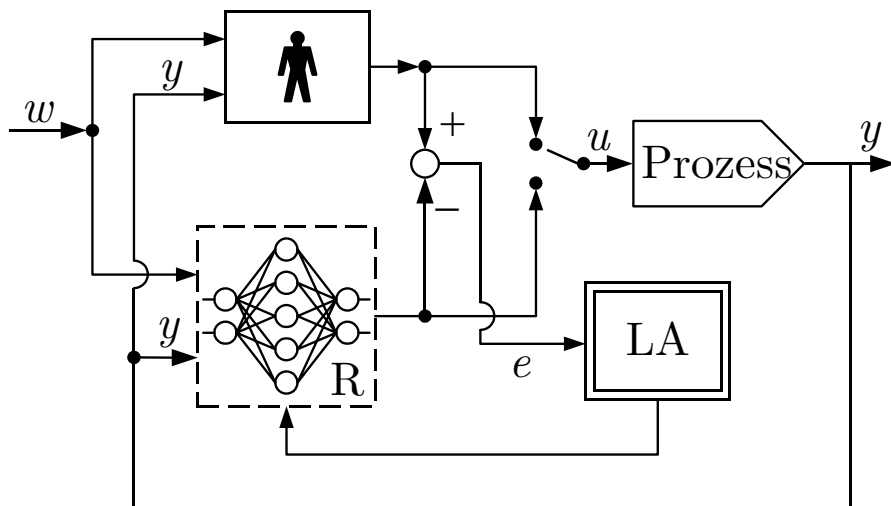
minimiert (Broomhead und Lowe, 1988). Wie aus dieser Darstellung ersichtlich, können mit RBF-Netzen Vektorfunktionen mit mehreren Veränderlichen approximiert werden. Sollen skalare Funktionen, die nur von *einer* Veränderlichen abhängig sind, approximiert werden, hat der Eingangsraum des Netzes die Dimension eins. Ausgehend von Gl. (2.89) und der Annahme, dass der Fehler beliebig reduzierbar ist, kann dann das RBF-Netz entsprechend

$$y(k) = w_0 x(k) + \sum_{j=1}^m w_j \Phi_j(|x(k) - \mathbf{c}_j|) \quad (2.90)$$

mit einem zusätzlichen linearen Term für diese Aufgabe verwendet werden.

### 2.6.2 Kopieren eines existierenden Reglers

Ein existierender Regler, beispielsweise in Form des Verhalten eines Prozessbedieners, lässt sich mit Hilfe eines ANN kopieren. Bild 2.9 zeigt, wie das Ver-



**Bild 2.9:** Nachbildung eines Reglers durch ein ANN

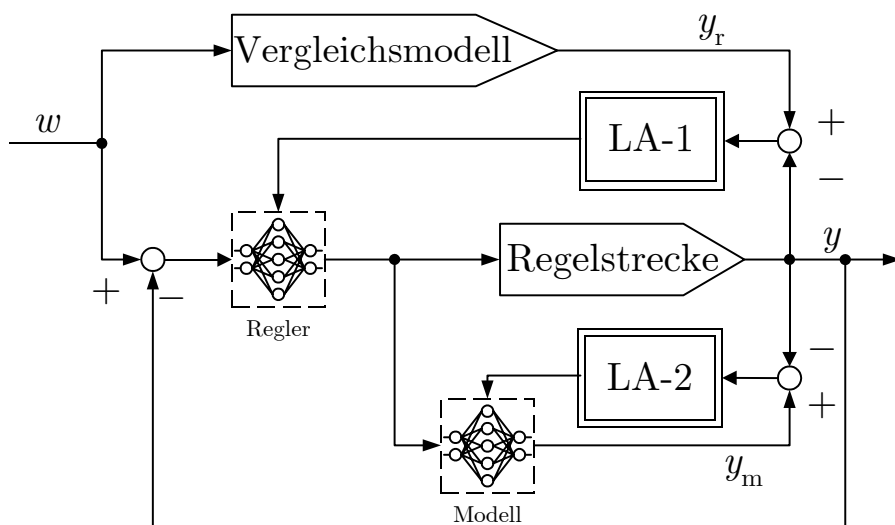
halten des Prozessbedieners auf ein neuronales Netz abgebildet werden kann. Parallel zum Bediener ist das Netz  $R$  geschaltet, welchem die gleichen Informationen  $w$  und  $y$  zugeführt werden. Zunächst wird der Prozess manuell geregelt und das Ausgangssignal des ANN mit dem vom Bediener erzeugten Stellsignal  $u$  verglichen. Die Abweichung wird vom Lernalgorithmus  $LA$  ausgewertet, wobei die Parameter des ANN so optimiert werden, dass der Fehler  $e$  minimal wird. Nach erfolgreichem Training hat der Regler  $R$  das Verhalten des Bedieners gelernt, und mit dem Schalter in der unteren Position übernimmt nun das ANN die Funktion des Reglers.

Es ergibt sich zunächst keine Verbesserung der Regelung, aber im Fall eines kopierten Bedieners ist das Netz beliebig reproduzierbar und unabhängig von störenden, äußeren Einflüssen. Weiterhin kann das Netz aufbauend auf dem

kopierten Verhalten verbessert werden. Ein Problem dieser Strategie ist, dass das ANN alle Informationen benötigt, die ein Bediener zur Regelung auswertet. Diese sind aber häufig nicht einmal dem Bediener bewusst.

### 2.6.3 Adaptive Regelung mit Vergleichsmodell

Dieses Regelsystem enthält einen Grundregelkreis bestehend aus Regelstrecke



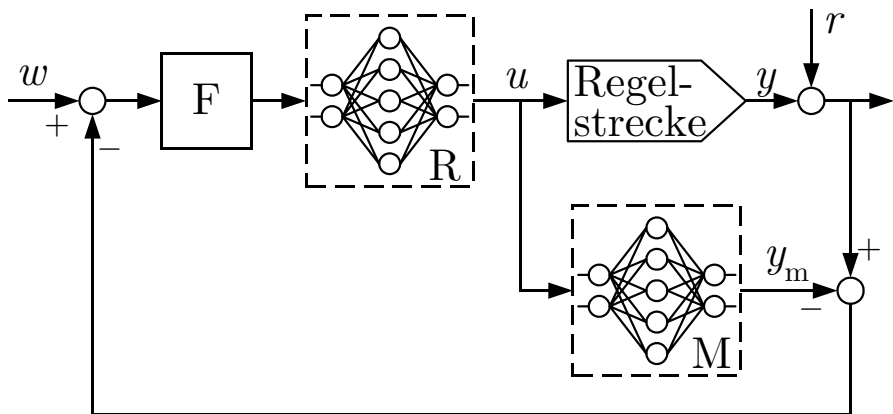
**Bild 2.10:** Indirekte adaptive Regelung mit Vergleichsmodell

und Regler. Das gewünschte Übertragungsverhalten des geschlossenen Regelkreises wird mit einem Vergleichsmodell vorgegeben. Um dies zu erreichen, wird bei nichtlinearen Prozessen der Einsatz von ANN vorgeschlagen (Narendra und Parthasarathy, 1990). Die Struktur dieses Reglers ist im Bild 2.10 dargestellt. Hierbei wird das Referenzsignal  $y_r$  mit dem Ausgangssignal  $y$  verglichen und der Fehler dem Lernalgorithmus LA-1 zugeführt. Der Lernalgorithmus trainiert das Netz im Regler R so, dass das Fehlersignal verschwindend klein wird, d.h. der Grundregelkreis folgt dem Vergleichsmodell. Da das Fehler-signal am Ausgang der Regelstrecke gebildet wird, liegt die unbekannte Regel-

strecke zwischen dem Netz und dem Fehlersignal. Der Fehler kann somit nicht zum Training zurückgeführt werden. Eine realisierbare Lösung bietet die im Bild 2.10 dargestellte Parallelschaltung eines zweiten ANN zur Regelstrecke, das deren Übertragungsverhalten mit Hilfe des Lernalgorithmus LA-2 lernt. Über dieses Netz kann dann der Fehler zum Netz im Regler geführt werden.

### 2.6.4 Adaptive Regelung mit internem Prozessmodell

Diese im Bild 2.11 dargestellte Regelstrategie, die auch unter der englischen



**Bild 2.11:** Regelung mit internem Prozessmodell

Bezeichnung *Internal Model Control* bekannt ist, wurde ebenfalls ursprünglich für lineare Regelstrecken entwickelt. Im Fall von nichtlinearen Systemen wird ein ANN parallel zur Regelstrecke geschaltet (Hunt et al., 1992). Ist das ANN perfekt trainiert, so werden nur Störungen zurückgekoppelt, ohne Störungen liegt eine Steuerung vor. Der ANN-Regler  $R$  kompensiert in diesem Fall die Regelstreckendynamik und enthält daher in seinem Übertragungsverhalten das inverse Regelstreckenmodell. Vorausgesetzt, dass der Regler  $R$  ebenfalls perfekt trainiert ist, wird das Verhalten des geschlossenen Systems allein durch das frei wählbare Filter  $F$  bestimmt. Diese Regelung ist nur für stabile Regelstrecken mit stabiler Nulldynamik geeignet, was eine starke Einschränkung bedeutet.

## 2.7 Zusammenfassung

Für die nichtlineare adaptive Regelung nichtlinearer Regelstrecken wurden Regelungsstrukturen vorgestellt, die sich in der Anwendbarkeit, Komplexität und den zugrundeliegenden Modellen unterscheiden.

Bei den ersten beiden Verfahren (Abschnitte 2.1 und 2.2) wird der adaptive Regler mit Hilfe der *Backstepping*-Methode entworfen. Hierbei wird bei jedem Entwicklungsschritt eine Zustandsgröße stabilisiert und ein Adaptionsgesetz für die unsicheren Parameter entwickelt. Damit diese Entwurfsmethode angewendet werden kann, muss die Regelstrecke entweder in der PSF- oder in der POF-Form beschrieben vorliegen. Für den Fall, dass die Regelstrecke in keiner der beiden Formen beschrieben ist, wurden allgemeine Transformationsbedingungen angegeben. Sind diese Bedingungen erfüllt, kann das System immer in eine der beiden Darstellungsformen transformiert werden. Aufgrund der komplexen Reglerstruktur kann dieses Verfahren nur für Anlagen niedriger Ordnung in Echtzeit eingesetzt werden. Desweiteren existiert keine Systematik, wie die zahlreichen Reglerparameter einzustellen sind. Durch den Entwurf bedingt, erhält man ein global asymptotisch stabiles Regelsystem. Hierbei wird aber implizit vorausgesetzt, dass beliebig viel Stellenergie zur Verfügung steht. Die ist aber bei realen Anlagen gerade nicht gegeben, was eine weitere starke Einschränkung dieser Methoden bedeutet.

Abschnitt 2.3 beschreibt die Grundsätze der adaptiven *Sliding-Mode*-Regelung. Es wird gezeigt, wie diese Methode ausgehend von der konventionellen *Sliding-Mode*-Regelung um eine Parameteradaption erweitert werden kann. Zur Vermeidung des in der Praxis störenden Ratterns wurden in neuester Zeit Erweiterungen vorgeschlagen (Utkin et al., 1999).

Das vierte und älteste Verfahren ist das *Speed-Gradienten*-Verfahren (Abschnitt 2.4). Bevor es für die Regelung von nichtlinearen Systemen entdeckt wurde, ist dieses Verfahren zur Lösung von ganz allgemeinen Optimierungsproblemen verwendet worden. Durch seinen relativ allgemeinen Ansatz ist es prinzipiell zur Regelung von Systemen in einer beliebigen Beschreibungsform geeignet. Es kann gezeigt werden, dass viele Speziallösungen adaptiver Rege-

lungen aus dem *Speed-Gradienten*-Verfahren hergeleitet werden können.

Im Abschnitt 2.5 wurde ein Verfahren zur Regelung von linearen Systemen mit statischen Nichtlinearitäten am Ein- oder Ausgang vorgestellt. Typische statische Nichtlinearitäten sind z. B. die Totzone oder die Getriebelose, deren beschreibende Funktionen an einigen Stellen Unstetigkeiten besitzen. Bei dieser Regelung wurde die Kenntnis der Parameter des linearen Systems vorausgesetzt. Es ist aber ebenfalls möglich, den Regler um eine Adaption für diese Parameter zu erweitern. Bei dem dargestellten Verfahren ist es nur möglich, eine Reglersynthese durchzuführen, wenn die Struktur der Nichtlinearität bekannt ist.

Abschnitt 2.6 enthält eine Übersicht der gängigen Verfahren zur adaptiven Regelung mit Hilfe von ANN. Der große Vorteil bei diesen Verfahren, der häufig hervorgehoben wird, ist, dass im Prinzip kein Wissen über die zu regelnde Anlage benötigt wird. Dies stimmt aber nur eingeschränkt. Um das Netz vernünftig konfigurieren zu können, müssen Neuronen im Eingangsraum sinnvoll verteilt werden. Wie schon zuvor erwähnt, gibt es hierfür aber keine systematische Lösung. Aus diesem Grund ist es notwendig, den Eingangsraum möglichst klein zu halten und somit existierendes Wissen im Regler zu berücksichtigen. Dies ist einer der großen Vorteile der im Folgenden beschriebenen Reglerstrukturen für Hammerstein- und Wiener-Systeme.



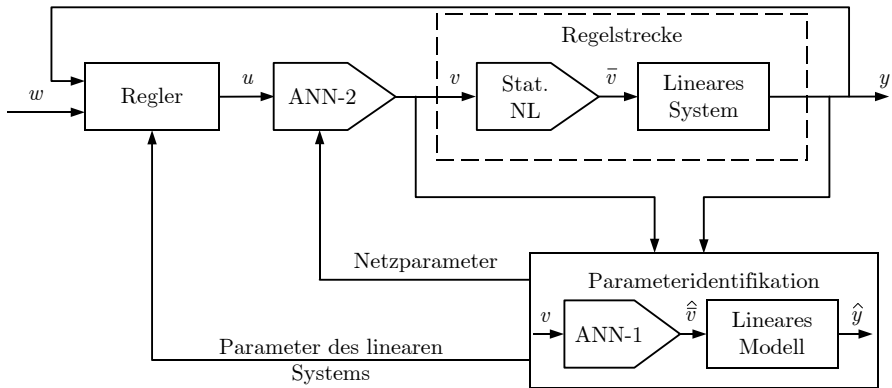
# 3 Adaptive Regelung von Hammerstein-Systemen

Die Beschreibung nichtlinearer Regelstrecken mit Hilfe von Hammerstein-Modellen ist für viele Anwendungen sehr realistisch, da dieses Verhalten häufig in technischen Anlagen auftritt (Knohl et al., 2000b) oder durch Eigenschaften der Stellglieder der Regelstrecke hinzugefügt wird. Hydraulische Ventile besitzen beispielsweise immer eine Totzone (Knohl und Unbehauen, 2000a). Weitere statische Eingangsnichtlinearitäten können durch Reibung oder Verstärkerkennlinien hervorgerufen werden. Da sich dieses Verhalten durch Alterung oder Änderung der Umgebung mit der Zeit ändern kann, eignen sich besonders adaptive Regler für derartige Regelstrecken. In (Šlemić und Lewis, 2000) wird ein Verfahren zur Regelung von Hammerstein-Systemen vorgeschlagen, bei dem die Eingangsnichtlinearität durch ein ANN gelernt wird, das lineare Teilsystem aber als bekannt und konstant angenommen wird. Eine weitere Struktur verwendet eine Kombination aus linearem Modell und MLP-ANN (Al-Duwaish et al., 1995). Dabei werden die Parameter des linearen Systems mit der RLS-Schätzung ermittelt und für das Training des ANN wird der BP-Lernalgorithmus verwendet. Es zeigt sich hierbei, dass die Konvergenz des BP-Lernalgorithmus stark von der Wahl der Lernrate abhängt (Al-Duwaish und Karim, 1997), weshalb seine Verwendung zur adaptiven Regelung schwierig ist. Im Folgenden werden zwei adaptive Regelungsverfahren mit ANN vorgestellt, bei denen alle Parameter mit der RLS-Schätzung bestimmt werden (Knohl und Unbehauen, 1998, 2000c). Um Parameterunsicherheit während der Startpha-

se der Adaption zu berücksichtigen, wurde beim zweiten Verfahren ein dualer Ansatz verwendet.

### 3.1 Indirekte adaptive Regelung

Die HANNNA-Regelungsstruktur ist im Bild 3.1 für den Fall eines Eingrößen (SISO)-Systems dargestellt, wobei die Regelstrecke als Serienschaltung einer unbekannten statischen Nichtlinearität und eines linearen Teilsystems beschrieben wird.



**Bild 3.1:** Strukturbild der HANNNA-Regelung

ben wird. Die messbaren Ein- und Ausgangssignale ( $v$  und  $y$ ) der Regelstrecke werden für die Parameteridentifikation verwendet. In diesem Block wird als Regelstreckenmodell eine Serienschaltung eines neuronalen Netzes (ANN-1) mit einem linearen Modell angesetzt. Als Ergebnis der Identifikation erhält man getrennt voneinander die Parameter des linearen Modells und die Gewichte des neuronalen Netzes. Das lineare Modell kann nun zum Entwurf eines Standardreglers, z.B. eines LQ-Reglers, verwendet werden. Mit Hilfe des neuronalen Netzes ANN-1 muss nun noch ANN-2 so trainiert werden, dass es die Eingangsnichtlinearität möglichst gut kompensiert. Dies kann auf verschiedene Weise

durchgeführt werden. Eine naheliegende Methode scheint dazu der Ansatz von

$$v = f_{\text{ANN-2}}(u) = f_{\text{ANN-1}}^{-1}(u) \quad (3.1)$$

zu sein, wobei  $f_{\text{ANN-1}}(u)$  die Ein-Ausgangsbeschreibung des Netzes ANN-1 und  $u$  das Stellsignal des linearen Reglers ist. Dieser Ansatz führt aber in einer Echtzeitumgebung zu zeitkritischen Problemen und zur numerischen Instabilität (Lewis et al., 1999). Eine wichtige Voraussetzung für eine *online*-Identifikation ist ein Algorithmus, bei dem die Anzahl an Rechenoperationen von der Zeit unabhängig ist (Ljung und Söderström, 1986). Dies ist aber bei der direkten numerischen Lösung von Gl. (3.1) nicht der Fall. Speziell in der Anfangsphase des Trainings von ANN-1, bei der sich durch Approximationsfehler mehrdeutige Kennlinien des Netzes ergeben können, scheitert dieses Vorgehen. Es ist deshalb für den praktischen Einsatz ungeeignet. Eine andere Möglichkeit wäre, ein analytisch invertierbares Netz zu wählen. Auch dies ist nur sehr eingeschränkt möglich. Deshalb wurde ein gänzlich anderes Vorgehen gewählt. Berechnet man über den gesamten Wertebereich des Eingangssignals  $v$  die zugehörigen Ausgangswerte  $\hat{v}$  mit dem trainierten Netz ANN-1 und vertauscht dann die Ein- und Ausgangswerte, so kann der neue Datensatz gleichzeitig zum Erlernen der inversen Nichtlinearität für das zweite Netz ANN-2 benutzt werden. Ein großer Vorteil bei diesem Vorgehen ist, dass auch bei nicht invertierbaren Kennlinien eine, im Sinne eines wählbaren Gütefunktional, optimale Lösung gefunden wird. Diese Eigenschaft wird im Abschnitt 6.4 ausführlich diskutiert.

Generell kann der Typ des neuronalen Netzes für ANN-1 und ANN-2 frei gewählt werden. Wegen ihrer interessanten Eigenschaften werden in dieser Arbeit Radial-Basis-Funktionen(RBF)-Netze verwendet. Zum einen ist allgemein bekannt, dass sie ausgezeichnete Funktionsapproximatoren sind (Chen und Billings, 1992). Zum anderen sind sie linear in den Netzgewichten, was die Verwendung des etablierten RLS-Schätzung zum Training erlaubt. Dieser Algorithmus sichert die schnelle und zuverlässige Konvergenz der Parameterschätzung.

Zusammengefasst lässt sich der Reglerentwurf in die folgenden Schritte unterteilen:

1. Auswahl des Netztyps von ANN-1 und ANN-2 sowie Festlegung der Startkonfiguration:
  - Anzahl der Neuronen
  - Neuronenzentren
  - Formparameter
2. Festlegung der Ordnung des linearen Modells.
3. Auswahl des Algorithmus für die Parameterschätzung.
4. Wahl eines adäquaten Reglers für das lineare System.
5. Berechnung des Stellgesetzes.
6. Training von ANN-1 und ANN-2.

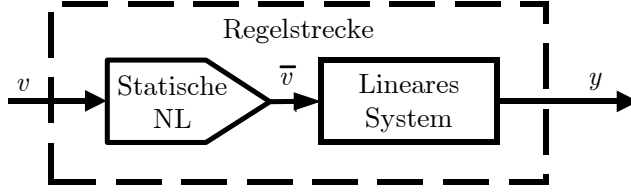
Mit Neuronenzentren sind hier die Zentren der Basisfunktionen gemeint, und der Formparameter ist ein freier Parameter zur Beeinflussung des Verlaufs der Basisfunktion (Abschnitt 2.6.1). Ein wesentlicher Vorteil dieser Struktur ist die Modularität. Die Art des ANN, der Regler für das lineare Teilsystem und der Parameterschätzalgorithmus können, mit nur geringen Einschränkungen, frei gewählt werden.

### 3.1.1 Der Reglerentwurf

Zum Entwurf des HANNNA-Reglers wird die Regelstrecke durch die im Bild 3.2 dargestellte Blockstruktur beschrieben. Mit der Annahme, dass das lineare Teilsystem nicht sprungfähig ist, ist seine Ein-Ausgangsfunktion durch

$$y(k) = - \sum_{i=1}^n a_i y(k-i) + \sum_{i=1}^n b_i \bar{v}(k-i) \quad (3.2)$$

gegeben. Hierbei ist  $\bar{v} = f(v(k))$  das Ausgangssignal der statischen Nichtlinearität. Da im Allgemeinen der Verlauf der Nichtlinearität unbekannt ist, wird im Folgenden ein RBF-Netz zu ihrer Approximation eingesetzt. Die Struktur und



**Bild 3.2:** Blockstruktur der Regelstrecke als Hammerstein-System

die Eigenschaften der RBF-Netze wurden schon im Abschnitt 2.6.1 beschrieben. Mit Gl. (2.90) kann für die Ausgangsgröße des nichtlinearen Teilsystems

$$\bar{v}(k) = w_0 v(k) + \sum_{j=1}^m w_j \Phi_j(|v(k) - \mathbf{c}_j|) \quad (3.3)$$

geschrieben werden. Wird Gl. (3.3) in Gl. (3.2) eingesetzt, erhält man

$$\begin{aligned} y(k) = & - \sum_{i=1}^n a_i y(k-i) + \sum_{i=1}^n b_i w_0 v(k-i) \\ & + \sum_{i=1}^n \sum_{j=1}^m b_i w_j \Phi_j(|v(k-i) - \mathbf{c}_j|). \end{aligned} \quad (3.4)$$

Zur vereinfachten Darstellung werden die Hilfsparameter

$$\beta_{ij} = b_i w_j \quad (3.5)$$

und die Hilfssignale

$$\begin{aligned} x_0(k) &= v(k) \\ x_j(k) &= \Phi_j(|v(k) - \mathbf{c}_j|), \quad j = 1, \dots, m \end{aligned} \quad (3.6)$$

eingeführt. Mit den Definitionen der Gln. (3.5) und (3.6) geht Gl. (3.4) über in

$$y(k) = - \sum_{i=1}^n a_i y(k-i) + \sum_{i=1}^n \sum_{j=0}^m \beta_{ij} x_j(k-i). \quad (3.7)$$

### 3.1.1.1 Parameteridentifikation

Die Beziehung gemäß (3.7) ist linear in den unbekannten Parametern und nach kurzer Umformung und Einführung des Gleichungsfehlers  $\varepsilon(k)$  lässt sie sich durch

$$y(k) = \mathbf{m}^T(k)\mathbf{p} + \varepsilon(k) \quad (3.8)$$

mit dem Datenvektor

$$\mathbf{m}^T(k) = [-y(k-1) \dots -y(k-n), \\ x_0(k-1) \dots x_0(k-n) \dots x_m(k-1) \dots x_m(k-n)], \quad (3.9)$$

und dem Parametervektor

$$\mathbf{p} = [a_1 \dots a_n, \beta_{10} \dots \beta_{n0} \dots \beta_{1m} \dots \beta_{nm}]^T \quad (3.10)$$

beschreiben. Für den Gleichungsfehler  $\varepsilon(k)$  wird angenommen, dass er durch gleichverteiltes weißes Rauschen mit dem Mittelwert null und einer Streuung  $\sigma_\varepsilon$  beschrieben werden kann. Zur Schätzung der Parameter wird dann der RLS-Algorithmus mit konstanter Spur (Mosca, 1995) verwendet:

$$\hat{\mathbf{p}}(k) = \hat{\mathbf{p}}(k-1) + \mathbf{q}(k)\hat{\varepsilon}(k), \quad (3.11)$$

$$\mathbf{q}(k) = \frac{\mathbf{P}(k-1)\mathbf{m}(k)}{1 + \mathbf{m}^T(k)\mathbf{P}(k-1)\mathbf{m}(k)}, \quad (3.12)$$

$$\hat{\varepsilon}(k) = y(k) - \mathbf{m}^T(k)\hat{\mathbf{p}}(k-1), \quad (3.13)$$

$$\mathbf{P}(k) = \frac{1}{\lambda(k)}[\mathbf{P}(k-1) - \mathbf{q}(k)\mathbf{m}^T(k)\mathbf{P}(k-1)] \quad (3.14)$$

und

$$\lambda(k) = 1 - \frac{\mathbf{m}^T(k)\mathbf{P}^2(k-1)\mathbf{m}(k)}{\text{tr}(\mathbf{P}(0))(1 + \mathbf{m}^T(k)\mathbf{P}(k-1)\mathbf{m}(k))}, \quad (3.15)$$

wobei die Schätzwerte des Gleichungsfehlers  $\varepsilon(k)$  und der Parameter  $\mathbf{p}(k)$  zur Zeit  $t_k$  mit  $\hat{\varepsilon}(k)$  bzw.  $\hat{\mathbf{p}}(k)$  bezeichnet werden. Der Vektor  $\mathbf{q}(k)$  ist der Kalman-sche Verstärkungsfaktor und  $\lambda(k)$  ist ein Gewichtungsfaktor, der so gewählt ist, dass die Spur der Kovarianzmatrix der Parameterschätzungen  $\mathbf{P}(k)$  konstant bleibt. Mit dieser Einstellung lässt sich das „Einschlafen“ der Schätzung

verhindern, damit der Algorithmus auch zur Identifikation von Regelstrecken, deren Parameter sich im Vergleich zur Eigendynamik nur langsam ändern, eingesetzt werden kann. Für weitere Einstellungen von  $\lambda(k)$  existieren eine Reihe von Vorschlägen (Unbehauen, 1988).

Im nächsten Schritt müssen aus den  $n(m+2)$  Schätzparametern die  $2n$  Parameter des linearen Teilsystems und die  $m$  Netzgewichte ermittelt werden. Mit

$$w_0 = 1 \quad (3.16)$$

und Gl. (3.5) können die geschätzten Parameter  $\hat{a}_i$  und  $\hat{b}_i$  des zugehörigen Modells der Regelstrecke direkt aus dem Schätzvektor  $\hat{\mathbf{p}}(k)$  abgelesen werden:

$$\hat{a}_i = \hat{p}_i, \quad (3.17)$$

und

$$\hat{b}_i = \hat{p}_{n+i} = \hat{\beta}_{i0}, \quad i = 1, \dots, n. \quad (3.18)$$

Aus Gl. (3.3) ist ersichtlich, dass durch die Wahl von  $w_0 = 1$  das Netz in der Startphase der Adaption ( $w_i = 0, i = 1, \dots, m$ ) eine Gerade mit der Steigung eins darstellt. Für die Bestimmung der Netzgewichte  $\hat{w}_j$  ergibt sich das überbestimmte Gleichungssystem

$$\begin{aligned} \hat{\beta}_{1j} &= \hat{b}_1 \hat{w}_j \\ \hat{\beta}_{2j} &= \hat{b}_2 \hat{w}_j \\ &\vdots \\ \hat{\beta}_{nj} &= \hat{b}_n \hat{w}_j. \end{aligned} \quad (3.19)$$

Die Netzgewichte können beispielsweise durch die Minimierung der Summe der Fehlerquadrate

$$\sum_{i=1}^n \sum_{j=1}^m (\hat{\beta}_{ij} - \hat{b}_i \hat{w}_j)^2 \quad (3.20)$$

ermittelt werden. Die Lösung ist gegeben durch

$$\hat{w}_j = \frac{\sum_{i=1}^n \hat{\beta}_{ij} \hat{b}_i}{\sum_{i=1}^n \hat{b}_i^2}, \quad j = 1, \dots, m. \quad (3.21)$$

### 3.1.1.2 Entwurf des LQ-optimalen Reglers

Die geschätzten Parameter des linearen Teilsystems, beschrieben durch die diskrete Übertragungsfunktion  $B(z^{-1})/A(z^{-1})$  mit

$$A(z^{-1}) = 1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2} + \dots + \hat{a}_n z^{-n} \quad (3.22)$$

und

$$B(z^{-1}) = \hat{b}_1 z^{-1} + \hat{b}_2 z^{-2} + \dots + \hat{b}_n z^{-n}, \quad (3.23)$$

werden im Folgenden zum Entwurf eines LQ-Reglers verwendet, der das Funktional

$$J = E\{(w(k) - y(k))^2 + \rho u^2(k)\}, \quad (3.24)$$

minimiert, d.h. die Varianz des Stellsignals und der Regelabweichung. Der Faktor  $\rho$  nimmt eine Bewertung des Stellsignals vor. Die Minimierung von Gl. (3.24) wird, im Falle einer SISO-Regelstrecke, mit dem Stellgesetz (Åström und Wittenmark, 1996)

$$U(z) = \frac{t_o}{R(z)} W(z) - \frac{S(z)}{R(z)} Y(z) \quad (3.25)$$

erreicht, wobei durch  $t_o$  ein Reglerparameter,  $S(z)$  und  $R(z)$  Reglerpolynome sowie durch  $W(z)$ ,  $U(z)$  und  $Y(z)$  der z-transformierte Wert von Sollwert, Stellgröße und Ausgangsgröße beschrieben werden. Mittels Spektralfaktorisierung entsprechend

$$P(z)P(z^{-1}) = \rho(z-1)A(z)A(z^{-1})(z^{-1}-1) + B(z)B(z^{-1}) \quad (3.26)$$

erhält man dann das charakteristische Polynom  $P(z)$  des geschlossenen Regelkreises. Zur Vermeidung einer bleibenden Regelabweichung, hervorgerufen



durch eventuelle Ungenauigkeiten in den Koeffizienten der Regelstreckenpolynome  $A(z^{-1})$  und  $B(z^{-1})$  oder Modellfehler, wurde in Gl. (3.26) der Faktor  $1 - z^{-1}$  eingeführt. Dieser bewirkt, dass der Regler integrales Verhalten zeigt. Die Reglerpolynome  $S(z)$  und  $R(z) = R^*(z)(1 - z^{-1})$  erhält man durch Lösung der Diophantischen Gleichung

$$A(z)(1 - z^{-1})R^*(z) + B(z)S(z) = P(z) \quad (3.27)$$

und  $t_o$  wird zu

$$t_o = \frac{P(1)}{B(1)} \quad (3.28)$$

gewählt, womit die statische Verstärkung des Regelkreises ohne Integrator zu eins gesetzt wird.

Der dargestellte Regler kann auch für Systeme mit Totzeit erweitert werden. Hierbei wird das  $B(z^{-1})$ -Polynom entsprechend der Totzeit erweitert. Zusätzlich muss dies im Identifikationsblock berücksichtigt werden. Ebenfalls mögliche Erweiterungen, wie beispielsweise das Hinzufügen einer Störübertragungsfunktion oder eines erweiterten Beobachterpolynoms  $T(z^{-1})$ , sollen hier nicht betrachtet werden, da sie zum Verständnis des vorgestellten Regelungskonzeptes nichts beitragen.

### 3.1.1.3 Training von ANN-2

Um den Entwurf des HANNNA-Reglers zu vervollständigen, muss das Training des zweiten Netzes ANN-2 (siehe Bild 3.1) näher betrachtet werden. Unter der Voraussetzung, dass die Eingangsnichtlinearität invertierbar ist, kann mit dem trainierten Netz ANN-1 ein Ein-Ausgangsdatensatz über den interessierenden Wertebereich erzeugt werden:

$$[\mathbf{u}_{\text{ANN-1}}(N) \mid \mathbf{y}_{\text{ANN-1}}(N) = \mathbf{f}_{\text{ANN-1}}(\mathbf{u}_{\text{ANN-1}}(N))] \quad (3.29)$$

mit

$$\mathbf{u}_{\text{ANN-1}}(N) = [u_{\text{ANN-1}}(1) \dots u_{\text{ANN-1}}(N)]^T, \quad (3.30)$$

$$\mathbf{y}_{\text{ANN-1}}(N) = [y_{\text{ANN-1}}(1) \dots y_{\text{ANN-1}}(N)]^T. \quad (3.31)$$

Die Werte von  $u_{\text{ANN-1}}$  werden hierbei so gewählt, dass der gesamte Stellbereich der Regelstrecke abgedeckt ist. Nach Vertauschung der Ein- und Ausgangswerte werden diese Daten für das Training von ANN-2 benutzt. Da das RBF-Netz linear in den Gewichten ist, lässt es sich mit

$$\mathbf{y}_{\text{ANN-2}}(N) - \mathbf{u}_{\text{ANN-2}}(N) = \mathbf{M}(N)\mathbf{p} + \boldsymbol{\varepsilon}(N), \quad (3.32)$$

beschreiben, wobei

$$\mathbf{M}(N) = \begin{pmatrix} \Phi_1(|u_{\text{ANN-2}}(1) - \mathbf{c}_1|) & \dots & \Phi_m(|u_{\text{ANN-2}}(1) - \mathbf{c}_m|) \\ \vdots & \vdots & \vdots \\ \Phi_1(|u_{\text{ANN-2}}(N) - \mathbf{c}_1|) & \dots & \Phi_m(|u_{\text{ANN-2}}(N) - \mathbf{c}_m|) \end{pmatrix} \quad (3.33)$$

und

$$\mathbf{p} = [w_1 \dots w_m]^T \quad (3.34)$$

ist. Mit der Annahme, dass der Gleichungsfehler  $\boldsymbol{\varepsilon}(N)$  durch weißes Rauschen mit dem Mittelwert null und der Varianz  $\sigma_\varepsilon^2$  beschreibbar ist, können die optimalen Netzgewichte im Sinne der kleinsten Gleichungsfehlerquadrate mit der direkten analytischen Lösung

$$\hat{\mathbf{p}} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T (\mathbf{y}_{\text{ANN-2}} - \mathbf{u}_{\text{ANN-2}}). \quad (3.35)$$

berechnet werden. Falls das direkte LSE-Verfahren in einer Echtzeitimplementierung zu zeitlichen Problemen führt, können selbstverständlich schnellere Verfahren, wie beispielsweise das RLS-Verfahren, verwendet werden. Mit der Wahl des Trainingsalgorithmus ist der Reglerentwurf abgeschlossen.

### 3.1.2 Die Wahl der Netzparameter

Beim Entwurf des HANNNA-Reglers wurde angenommen, dass RBF-Netze linear in den Netzgewichten sind. Diese Annahme stimmt nur, wenn sowohl die Zentren  $\mathbf{c}$  als auch die Formparameter  $\mathbf{w}$  der Basisfunktionen zuvor gewählt wurden. Deren optimale Festlegung und die Anzahl der Neuronen sind noch

ein ungelöstes Problem. Je größer die Dimension des Eingangsraums ist, desto komplexer wird diese Aufgabe. Bei der hier gegebenen Problemstellung, der Approximation einer statischen, skalaren, nichtlinearen Funktion, hat jedoch der Eingangsraum des Netzes die Dimension eins. Dies führt zu folgendem einfachen Vorgehen bei der Festlegung der Netzparameter und der Neuronenzahl:

1. Da die Größe des Eingangsraums bekannt ist, können die Neuronen in gleichen Abständen in diesem verteilt werden.
2. Die Genauigkeit der Approximation kann durch Hinzufügen von Neuronen erhöht werden. Hierbei ist auf den steigenden Rechenaufwand zu achten.
3. Der Formparameter wird grob unter Berücksichtigung der Neuronenzahl und der Größe des Eingangsraums eingestellt. Die Feineinstellung wird während der ersten Simulation vorgenommen.
4. Die genaue Einstellung des Formparameters hat hierbei einen geringeren Einfluss auf die Genauigkeit des Netzes als die Anzahl und die Zentrenlage der Neuronen.

Wie schon erwähnt, ist dies keinesfalls die optimale Lösung. Die Simulationsergebnisse zeigen aber, dass man mit diesem Vorgehen rasch zu einem brauchbaren Ergebnis kommt.

### 3.1.3 Stabilitätsbetrachtung

Die Stabilität ist bei adaptiven Regel-Systemen neben Parametervariationen der Regelstrecke gleichzeitig auch von Veränderungen der Reglerparameter abhängig. Für die Analyse der Stabilität derartiger nichtlinearer Systeme existieren leistungsfähige Vorgehensweisen und Verfahren wie beispielsweise

- die Betrachtung der lokalen Stabilität durch Linearisierung,
- der Beweis der Stabilität mit der Ljapunow-Theorie oder
- die Anwendung der Hyperstabilitäts-Theorie.

Diese Verfahren sind mathematisch sehr aufwendig und liefern nur eingeschränkt das gewünschte Ergebnis. Der Beweis der lokalen Stabilität liefert keine Aussage über das globale Systemverhalten. Ein wesentliches Problem bei der Ljapunow-Theorie ist das Auffinden einer geeigneten Ljapunow-Funktion. Aus den genannten Gründen wird auf einen ausführlichen Stabilitätsbeweis verzichtet und eine heuristische Stabilitätsbetrachtung gegeben. Mit den Annahmen

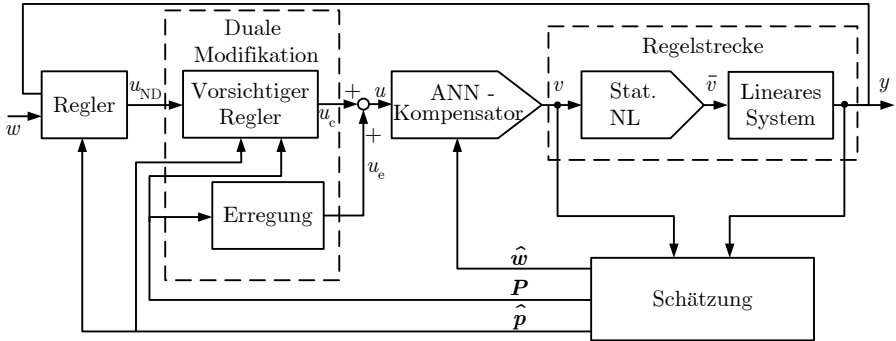
1. die Ordnung der Polynome  $A(z^{-1})$  und  $B(z^{-1})$  ist bekannt,
2. die Polynome  $A(z^{-1})$  und  $B(z^{-1})$  sind teilerfremd und
3. die Parameterschätzungen bleiben beschränkt,

kann die Beschränktheit des Stellsignales  $u(k)$  und des Ausgangssignales  $y(k)$  des vorgestellten Regelkreises ohne die Nichtlinearitäten gezeigt werden (Goodwin und Sin, 1984). Im Fall der vollständigen Kompensation wäre damit auch der gesamte Kreis stabil. Bei nicht vollständiger Kompensation ergibt sich ein Störsignal am Eingang des linearen Regelstreckenteils. Wird der Regelkreis instabil, kann dies somit nur durch das Störsignal hervorgerufen werden. Da aber dieses Signal gerade den Approximationsfehler von ANN-2 beschreibt, kann dieser durch Änderung der Neuronenzahl und Variation der Formparameter beliebig verringert werden. Es sollte somit immer möglich sein, einen stabilen geschlossenen Regelkreis zu erhalten. In jedem Fall erhält man ein kleineres Störsignal als ohne Kompensation.

## 3.2 Duale Adaptive Regelung

Beim Entwurf adaptiver Regelungssysteme, speziell solche mit neuronalen Netzen, besitzt man besonders in der Anfangsphase der Adaption keine oder nur geringe Kenntnis über die tatsächlichen Parameter der Regelstrecke. Deshalb ist in diesem Moment der Parameterschätzfehler besonders groß und somit die Parameterschätzung unsicher. Mit einem Regler, der auf der Basis dieser unsicheren Schätzungen entworfen ist, wird man im Allgemeinen kein zufriedenstellendes Regelverhalten erzielen. Um dieses Problem zu lösen, bietet sich

die duale Regelung an (Åström und Wittenmark, 1971). Für den Fall einer unsicheren Schätzung wird im Folgenden der im Abschnitt 3.1 vorgestellte Regler um einen dualen Anteil erweitert (Knohl et al., 2000a).



**Bild 3.3:** Strukturbild der D-HANNNA-Regelung

Der erweiterte Regelkreis ist im Bild 3.3 dargestellt. Wie zuvor handelt es sich hier um ein indirektes Verfahren, bei dem zunächst die Modellparameter geschätzt werden. Basierend auf diesen Parametern wird dann ein linearer Regler und ein ANN-Kompensator für die statische Eingangsnichtlinearität berechnet. Neu ist, dass dem Block Schätzung neben den Parameterschätzwerten  $\hat{\mathbf{w}}$  und  $\hat{\mathbf{p}}$  auch die Kovarianzmatrix  $\mathbf{P}$  entnommen wird. Sie wird dem Block *Duale Modifikation* zugeführt, der sich zwischen dem Ausgangssignal  $u_{\text{ND}}$  des nicht dualen Reglers und dem ANN-Kompensator befindet. In dieser Stufe wird mit Hilfe der Matrix  $\mathbf{P}$  die Parameterunsicherheit bestimmt und entsprechend wird ein vorsichtiges (*cautious*) Stellsignal  $u_c$  und eine Erregung (*excitation*)  $u_e$  zur Verbesserung der Parameterkonvergenz berechnet. Das Prinzip des hier verwendeten *bicriterial approach* (Filatov und Unbehauen, 2000) lässt sich wie folgt zusammenfassen:

Ist die Schätzung unsicher, wird ein vorsichtiges Stellsignal und eine geeignete Erregung zur Verbesserung der Konvergenz der Parameterschätzung ausgegeben. Ist die Parameterschätzung zuverlässig,

wird die Erregung abgestellt und das Stellsignal  $u_{\text{ND}}$  ausgegeben.

Im Einzelnen setzt sich der Reglerentwurf aus folgenden Schritten zusammen:

1. Auswahl des Netztyps von ANN-1 und ANN-2 sowie Festlegung der Startkonfiguration.
2. Festlegung der Ordnung des linearen Modells.
3. Auswahl des Algorithmus für die Parameterschätzung.
4. Berechnung des nicht-dualen Stellsignals  $u_{\text{ND}}$ .
5. Berechnung des vorsichtigen Stellsignals  $u_c$  und der Erregung  $u_e$ .
6. Training der Netze ANN-1 und ANN-2.

Der Entwurf des D-HANNNA-Reglers unterscheidet sich wesentlich nur im Punkt 5 von dem des HANNNA-Reglers. Aus diesem Grund werden im Folgenden nur die jeweiligen Unterschiede herausgearbeitet und notwendige Definitionen eingeführt.

### 3.2.1 Der Reglerentwurf

Ausgehend von einem RBF-Netz zur Darstellung der statischen Nichtlinearität, wird die Regelstrecke als Hammerstein-System gemäß Gl. (3.7)

$$y(k) = - \sum_{i=1}^n a_i y(k-i) + \sum_{i=1}^n \sum_{j=0}^m \beta_{ij} x_j(k-i), \quad (3.36)$$

angesetzt. Fasst man die unbekannten Parameter zusammen, liefert dies bei Verwendung der Definitionen (3.5), (3.6) und (3.16) die Fehlergleichung:

$$y(k+1) = \mathbf{m}^T(k) \mathbf{p} + \varepsilon(k+1), \quad (3.37)$$

mit dem Parametervektor

$$\mathbf{p}^T = [b_1 \dots b_n, a_1 \dots a_n | \beta_{11} \dots \beta_{n1} \dots \beta_{1m} \dots \beta_{nm}]$$

$$= [\mathbf{p}_1^T \mid \mathbf{p}_2^T] = [b_1 \mid \mathbf{p}_0^T \mid \mathbf{p}_2^T] \quad (3.38)$$

und dem Daten- oder Regressionsvektor

$$\begin{aligned} \mathbf{m}^T(k) &= [u(k) \dots u(k-n+1), -y(k) \dots -y(k-n+1) \mid \\ &\quad x_1(k) \dots x_1(k-n+1) \dots x_m(k) \dots x_m(k-n+1)] \\ &= [\mathbf{m}_1^T(k) \mid \mathbf{m}_2^T(k)] = [u(k) \mid \mathbf{m}_0^T(k) \mid \mathbf{m}_2^T(k)]. \end{aligned} \quad (3.39)$$

Die Parameter des linearen Teilsystems sind hierbei im Vektor  $\mathbf{p}_1$  und die des Netzes im Vektor  $\mathbf{p}_2$  zusammengefasst. Eine entsprechende Aufteilung ist ebenfalls beim Regressionsvektor zu finden. Bei Verwendung der RLS-Schätzung erhält man nun

$$\hat{\mathbf{p}}(k) = E\{\mathbf{p}(k) \mid \mathfrak{S}_k\} \quad (3.40)$$

als Erwartungswert von  $\mathbf{p}$  und die Kovarianzmatrix der Parameterschätzung

$$\mathbf{P}(k) = E\{(\mathbf{p}(k) - \hat{\mathbf{p}}(k))(\mathbf{p} - \hat{\mathbf{p}}(k))^T \mid \mathfrak{S}_k\} \quad (3.41)$$

mit

$$\mathfrak{S}_k = [y(0), y(k), \dots, u(0), u(k-1)], \mathfrak{S}_0 = [y(0)] \quad (3.42)$$

als Menge aller Ein- und Ausgangssignale der Regelstrecke zum Zeitpunkt  $t_k$ .

Die Kovarianzmatrix  $\mathbf{P}(k)$  kann in der Form

$$\mathbf{P}(k) = \begin{bmatrix} \mathbf{P}_1(k) & \mathbf{P}_{12}^T(k) \\ \mathbf{P}_{12}(k) & \mathbf{P}_2(k) \end{bmatrix} \quad (3.43)$$

geschrieben werden. Aus Gl. (3.43) geht hervor, dass  $\mathbf{P}_1(k)$  der Kovarianzmatrix der Parameter des linearen Teilsystems und  $\mathbf{P}_2(k)$  die Kovarianzmatrix der Netzwichte darstellt. Mit diesen Definitionen kann nun die duale Modifikation des Reglers durchgeführt werden.

### 3.2.1.1 Duale Modifikation

Dem HANNNA-Regler aus Abschnitt 3.1 wird der *bicriterial approach* hinzugefügt. Transformiert man das Stellgesetz aus Gl. (3.25) in den Zeitbereich,

so ist das *nominelle* Stellsignal, welches durch einen Regler basierend auf bekannten Regelstreckenparametern erzeugt würde, gegeben durch

$$u_n(k) = t_o w(k) - \mathbf{p}_3^T(k) \mathbf{m}_3(k) \quad (3.44)$$

mit

$$\mathbf{p}_3^T(k) = [r_1(k) \dots r_{n-1}(k), s_0(k) \dots s_{n-1}(k)] \quad (3.45)$$

und

$$\mathbf{m}_3^T(k) = [u(k-1) \dots u(k-n+1), y(k) \dots y(k-n+1)]. \quad (3.46)$$

Entsprechend lässt sich das nominelle Ausgangssignal der Regelstrecke durch

$$y_n(k+1) = b_1 u_n(k) + \mathbf{p}_0^T(k) \mathbf{m}_0(k) \quad (3.47)$$

beschreiben.

Um das Regelverhalten zu verbessern, werden die beiden Gütefunktionen

$$J_k^e = -E\{(y(k+1) - \hat{\mathbf{p}}_1^T \mathbf{m}_1(k))^2 \mid \mathfrak{F}_k\} \quad (3.48)$$

und

$$J_k^c = E\{(y_n(k+1) - y(k+1))^2 \mid \mathfrak{F}_k\} \quad (3.49)$$

eingeführt, wobei das Ausgangssignal der Regelstrecke durch

$$y(k+1) = \mathbf{p}_1^T(k) \mathbf{m}_1(k) \quad (3.50)$$

gegeben ist. Die Gütefunktion, definiert durch Gl. (3.48), gewichtet den Prädiktionsfehler. Durch Minimierung dieser Funktion wird er vergrößert, wodurch eine bessere Parameterkonvergenz erreicht wird. Sie erzeugt somit das Erregungssignal  $u_e(k)$ . Die durch Gl. (3.49) beschriebene Gütefunktion gewichtet den Abstand des Systemausgangs  $y(k)$  vom nominellen Ausgang  $y_n(k)$  oder, da dieser unbekannt ist, vom geschätzten nominellen Ausgang

$$\hat{y}_n(k+1) = \hat{b}_1 u_n(k) + \hat{\mathbf{p}}_0^T(k) \mathbf{m}_0(k). \quad (3.51)$$



Das endgültige Stellsignal des dualen Reglers ergibt sich aus der Minimierung der Gütefunktionen gemäß der Gln. (3.48) und (3.49)

$$u(k) = \arg \min_{u(k) \in \Omega_k} \{J_k^e\}, \quad (3.52)$$

mit den Randbedingungen

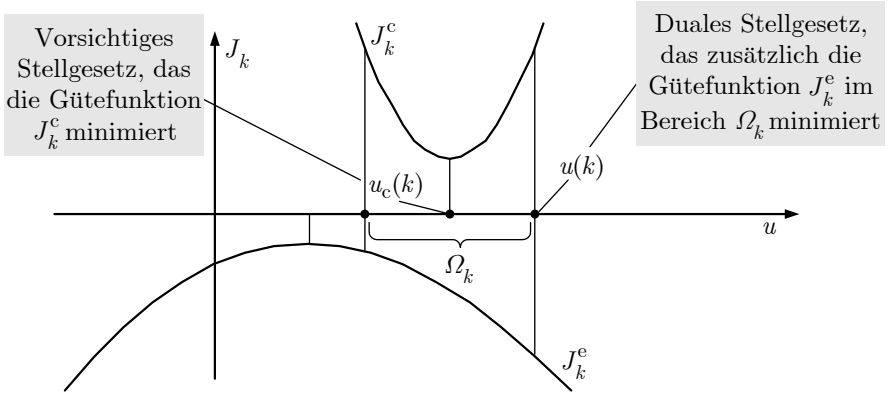
$$\Omega_k = [u_c(k) - \theta(k); u_c(k) + \theta(k)], \quad (3.53)$$

$$\theta(k) = h_k\{\mathbf{P}(k)\}, \quad (3.54)$$

wobei  $h_k(\cdot)$  eine differenzierbare positive Funktion ist, für die

$$\lim_{k \rightarrow \infty} \theta(k) = 0, \text{ wenn } \text{tr}(\mathbf{P}(k)) = 0 \quad (3.55)$$

gilt. Häufig wird  $h_k(\mathbf{P}(k)) = \eta \text{tr}(\mathbf{P}(k))$  mit  $\eta > 0$  gewählt. Der Faktor  $\eta$  beschreibt dann die Verstärkung des Erregungssignals. Die grafische Lösung dieses Problems ist im Bild 3.4 dargestellt. Zur numerischen Lösung wird



**Bild 3.4:** Grafische Darstellung der Optimierung des Stellsignals  $u(k)$  nach dem des *bicriterial approach*

Gl. (3.51) in Gl. (3.49) eingesetzt, wobei  $y_n(k)$  durch  $\hat{y}_n(k)$  und  $u_n(k)$  durch  $u_{\text{ND}}(k)$  ersetzt wird. Es ergibt sich

$$J_k^c = \text{E} \left\{ \left( \hat{y}_n(k+1) - y(k+1) \right)^2 \mid \mathfrak{F}_k \right\}$$

$$\begin{aligned}
&= \mathbb{E} \left\{ \left( \hat{b}_1(k) u_{\text{ND}}(k) + \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) - y(k+1) \right)^2 \mid \mathfrak{S}_k \right\} \\
&= \mathbb{E} \left\{ \hat{b}_1^2(k) u_{\text{ND}}^2(k) + \mathbf{m}_0^{\text{T}}(k) \hat{\mathbf{p}}_0(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) + y^2(k+1) \right. \\
&\quad \left. + 2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) - 2\hat{b}_1(k) u_{\text{ND}}(k) y(k+1) \right. \\
&\quad \left. - 2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) y(k+1) \mid \mathfrak{S}_k \right\} \\
&= \mathbb{E} \left\{ \hat{b}_1^2(k) u_{\text{ND}}^2(k) \mid \mathfrak{S}_k \right\} + \mathbb{E} \left\{ \mathbf{m}_0^{\text{T}}(k) \hat{\mathbf{p}}_0(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mid \mathfrak{S}_k \right\} \\
&\quad + \mathbb{E} \left\{ y^2(k+1) \mid \mathfrak{S}_k \right\} + \mathbb{E} \left\{ 2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mid \mathfrak{S}_k \right\} \\
&\quad + \mathbb{E} \left\{ -2\hat{b}_1(k) u_{\text{ND}}(k) y(k+1) \mid \mathfrak{S}_k \right\} \\
&\quad + \mathbb{E} \left\{ -2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) y(k+1) \mid \mathfrak{S}_k \right\}. \tag{3.56}
\end{aligned}$$

Die sich ergebenden Terme für die Erwartungswerte können unabhängig voneinander berechnet werden.

$$\mathbb{E} \left\{ \hat{b}_1^2(k) u_{\text{ND}}^2(k) \mid \mathfrak{S}_k \right\} = \hat{b}_1^2(k) u_{\text{ND}}^2(k) \tag{3.57}$$

$$\mathbb{E} \left\{ \mathbf{m}_0^{\text{T}}(k) \hat{\mathbf{p}}_0(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mid \mathfrak{S}_k \right\} = \mathbf{m}_0^{\text{T}}(k) \hat{\mathbf{p}}_0(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \tag{3.58}$$

$$\mathbb{E} \left\{ 2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mid \mathfrak{S}_k \right\} = 2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \tag{3.59}$$

$$\begin{aligned}
\mathbb{E} \left\{ -2\hat{b}_1(k) u_{\text{ND}}(k) y(k+1) \mid \mathfrak{S}_k \right\} &= -2\hat{b}_1(k) u_{\text{ND}}(k) \mathbb{E} \{ y(k+1) \} \\
&= -2\hat{b}_1(k) u_{\text{ND}}(k) \mathbb{E} \{ \mathbf{p}_1^{\text{T}}(k) \mathbf{m}_1(k) \} \\
&= -2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_1^{\text{T}}(k) \mathbf{m}_1(k) \\
&= -2\hat{b}_1(k) u_{\text{ND}}(k) [\hat{b}_1 \mid \hat{\mathbf{p}}_0^{\text{T}}(k)] \begin{bmatrix} u(k) \\ \mathbf{m}_0(k) \end{bmatrix} \\
&= -2\hat{b}_1^2(k) u_{\text{ND}}(k) u(k) - 2\hat{b}_1(k) u_{\text{ND}}(k) \hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \tag{3.60}
\end{aligned}$$

$$\begin{aligned}
\mathbb{E} \left\{ -2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) y(k+1) \mid \mathfrak{S}_k \right\} &= -2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mathbb{E} \{ y(k+1) \} \\
&= -2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \mathbb{E} \{ \mathbf{p}_1^{\text{T}}(k) \mathbf{m}_1(k) \} \\
&= -2\hat{\mathbf{p}}_0^{\text{T}}(k) \mathbf{m}_0(k) \hat{\mathbf{p}}_1^{\text{T}}(k) \mathbf{m}_1(k)
\end{aligned}$$

$$= -2\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k)\hat{b}_1(k)u(k) - 2\mathbf{m}_0^T(k)\hat{\mathbf{p}}_0(k)\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k) \quad (3.61)$$

$$\begin{aligned} \mathbb{E} \left\{ y^2(k+1) \mid \mathfrak{F}_k \right\} &= \text{var} \{ y(k+1) \} + \mathbb{E}^2 \{ y(k+1) \} \\ &= \mathbb{E} \left\{ \left( y(k+1) - \mathbb{E} \{ y(k+1) \} \right)^2 \right\} + \mathbb{E}^2 \{ y(k+1) \} \\ &= \mathbb{E} \left\{ \left( \mathbf{p}_1^T(k)\mathbf{m}_1(k) + \xi(k) - \hat{\mathbf{p}}_1^T(k)\mathbf{m}_1(k) \right) \right. \\ &\quad \left. \left( \mathbf{p}_1^T(k)\mathbf{m}_1(k) + \xi(k) - \hat{\mathbf{p}}_1^T(k)\mathbf{m}_1(k) \right) \right\} + \mathbb{E}^2 \{ y(k+1) \} \\ &= \mathbf{m}_1^T(k) \mathbb{E} \left\{ \left( \mathbf{p}_1^T(k) - \hat{\mathbf{p}}_1^T(k) \right)^T \left( \mathbf{p}_1^T(k) - \hat{\mathbf{p}}_1^T(k) \right) \right\} \mathbf{m}_1(k) + \\ &\quad \sigma_\xi^2 + \mathbb{E}^2 \{ y(k+1) \} \\ &= \mathbf{m}_1^T(k) \mathbf{P}_1(k) \mathbf{m}_1(k) + \sigma_\xi^2 + \mathbb{E}^2 \{ y(k+1) \} \\ &= [u(k) \mid \mathbf{m}_0^T(k)] \begin{bmatrix} p_{b_1}(k) & \mathbf{p}_{b_1\mathbf{p}_0}^T(k) \\ \mathbf{p}_{b_1\mathbf{p}_0}(k) & \mathbf{P}_{\mathbf{p}_0}(k) \end{bmatrix} \begin{bmatrix} u(k) \\ \mathbf{m}_0(k) \end{bmatrix} + \sigma_\xi^2 + \mathbb{E}^2 \{ y(k+1) \} \\ &= [p_{b_1}(k)u(k) + \mathbf{m}_0^T(k)\mathbf{p}_{b_1\mathbf{p}_0}(k) \mid \mathbf{p}_{b_1\mathbf{p}_0}^T(k)u(k) \\ &\quad + \mathbf{m}_0^T(k)\mathbf{P}_{\mathbf{p}_0}(k)] \begin{bmatrix} u(k) \\ \mathbf{m}_0(k) \end{bmatrix} + \sigma_\xi^2 + \mathbb{E}^2 \{ y(k+1) \} \\ &= p_{b_1}(k)u^2(k) + \mathbf{m}_0^T(k)\mathbf{p}_{b_1\mathbf{p}_0}(k)u(k) + \mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)u(k) \\ &\quad + \mathbf{m}_0^T(k)\mathbf{P}_{\mathbf{p}_0}(k)\mathbf{m}_0(k) + \sigma_\xi^2 + \mathbf{m}_1^T(k)\hat{\mathbf{p}}_1\hat{\mathbf{p}}_1^T\mathbf{m}_1(k) \end{aligned} \quad (3.62)$$

Das Signal  $\xi(k)$  beschreibt weißes Messrauschen mit der Varianz  $\sigma_\xi^2$  und dem Mittelwert null. Durch Einsetzen der Gln. (3.57), (3.58), (3.59), (3.60), (3.61), (3.62) in Gl. (3.56) erhält man

$$\begin{aligned} J_k^c &= \hat{b}_1^2(k)u_{\text{ND}}^2(k) + \mathbf{m}_0^T(k)\hat{\mathbf{p}}_0(k)\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k) + 2\hat{b}_1(k)u_{\text{ND}}(k)\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k) \\ &\quad - 2\hat{b}_1^2(k)u_{\text{ND}}(k)u(k) - 2\hat{b}_1(k)u_{\text{ND}}(k)\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k) \\ &\quad - 2\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k)\hat{b}_1(k)u(k) - 2\mathbf{m}_0^T(k)\hat{\mathbf{p}}_0(k)\hat{\mathbf{p}}_0^T(k)\mathbf{m}_0(k) \\ &\quad + p_{b_1}(k)u^2(k) + \mathbf{m}_0^T(k)\mathbf{p}_{b_1\mathbf{p}_0}(k)u(k) + \mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)u(k) \\ &\quad + \mathbf{m}_0^T(k)\mathbf{P}_{\mathbf{p}_0}(k)\mathbf{m}_0(k) + \sigma_\xi^2 + \mathbf{m}_1^T(k)\hat{\mathbf{p}}_1\hat{\mathbf{p}}_1^T\mathbf{m}_1(k) \end{aligned} \quad (3.63)$$

oder mit

$$\begin{aligned} \mathbf{m}_1^T(k) \hat{\mathbf{p}}_1 \hat{\mathbf{p}}_1^T \mathbf{m}_1(k) &= \hat{b}_1^2 u^2(k) + \hat{\mathbf{p}}_0^T(k) \mathbf{m}_0(k) \hat{b}_1(k) u(k) \\ &+ \mathbf{m}_0^T(k) \hat{\mathbf{p}}_0(k) \hat{b}_1(k) u(k) + \mathbf{m}_0^T(k) \hat{\mathbf{p}}_0(k) \hat{\mathbf{p}}_0^T(k) \mathbf{m}_0(k) \end{aligned} \quad (3.64)$$

ergibt sich schliesslich

$$\begin{aligned} J_k^c &= \hat{b}_1^2(k) u_{\text{ND}}^2(k) - 2\hat{b}_1^2(k) u_{\text{ND}}(k) \underline{u(k)} + p_{b_1} \underline{u^2(k)} + 2\mathbf{p}_{b_1 \mathbf{p}_0}^T(k) \mathbf{m}_0(k) \underline{u(k)} \\ &+ \hat{b}_1^2 \underline{u^2(k)} + \mathbf{m}_0^T(k) \mathbf{P}_{\mathbf{p}_0}(k) \mathbf{m}_0(k) + \sigma_\xi^2 \\ &= (p_{b_1} + \hat{b}_1^2) \underline{u^2(k)} + (2\mathbf{p}_{b_1 \mathbf{p}_0}^T(k) \mathbf{m}_0(k) - 2\hat{b}_1^2(k) u_{\text{ND}}(k)) \underline{u(k)} \\ &+ \hat{b}_1^2(k) u_{\text{ND}}^2(k) + \mathbf{m}_0^T(k) \mathbf{P}_{\mathbf{p}_0}(k) \mathbf{m}_0(k) + \sigma_\xi^2 \end{aligned} \quad (3.65)$$

als Gütefunktion. Die Struktur der Kovarianzmatrix  $\mathbf{P}_1(k)$  wurde dabei wie folgt angenommen

$$\mathbf{P}_1(k) = \begin{bmatrix} p_{b_1}(k) & \mathbf{p}_{b_1 \mathbf{p}_0}^T(k) \\ \mathbf{p}_{b_1 \mathbf{p}_0}(k) & \mathbf{P}_{\mathbf{p}_0}(k) \end{bmatrix}. \quad (3.66)$$

Das Minimum der Gütefunktion  $J_k^c$  (Gl. (3.65)) bezüglich des Stellsignals kann durch Nullsetzen der Ableitung

$$\left. \frac{\partial J_k^c}{\partial u_u} \right|_{u=u_c} = 2(p_{b_1} + \hat{b}_1^2) u(k) + 2(\mathbf{p}_{b_1 \mathbf{p}_0}^T(k) \mathbf{m}_0(k) - \hat{b}_1^2(k) u_{\text{ND}}(k)) \Big|_{u=u_c} = 0 \quad (3.67)$$

berechnet werden und man erhält

$$u_c(k) = \frac{\hat{b}_1^2(k) u_{\text{ND}}(k) - \mathbf{p}_{b_1 \mathbf{p}_0}^T(k) \mathbf{m}_0(k)}{\hat{b}_1^2(k) + p_{b_1}(k)}. \quad (3.68)$$

Das Minimum der Gütefunktion  $J_k^c$  liegt am Rand des Definitionsbereichs der Stellgröße (siehe Bild 3.4) und bei Berücksichtigung der in Gl. (3.53) eingeführten Randbedingung führt dies zu

$$u(k) = u_c(k) + \theta(k) \operatorname{sgn} \left\{ J_k^c(u_c(k) - \theta(k)) - J_k^c(u_c(k) + \theta(k)) \right\} = u_c(k) + u_e(k). \quad (3.69)$$

Zur Berechnung des Stellgesetzes (Gl. (3.69)) muss nun noch das Erregungssignal  $u_e(k)$  ermittelt werden. Wird Gl. (3.50) in die Gütefunktion  $J_k^c$  eingesetzt, erhält man

$$J_k^c \{ u(k) \} = -E \left\{ \left( \mathbf{p}_1^T(k) \mathbf{m}_1(k) + \xi(k) - \hat{\mathbf{p}}^T(k) \mathbf{m}_1(k) \right)^2 \mid \mathfrak{F}_k \right\}$$

$$\begin{aligned}
&= -\mathbb{E} \left\{ \left( (b_1(k) - \hat{b}_1(k))u(k) + (\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k) + \xi(k) \right)^2 \mid \mathfrak{S}_k \right\} \\
&= -\mathbb{E} \left\{ (b_1(k) - \hat{b}_1(k))^2 u^2(k) + \mathbf{m}_0^T(k) (\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k)) \right. \\
&\quad \cdot (\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k) + \xi^2(k) + 2(b_1(k) - \hat{b}_1(k))u(k) \\
&\quad \cdot (\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k) + 2(b_1(k) - \hat{b}_1(k))u(k)\xi(k) \\
&\quad \left. + 2(\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k)\xi(k) \mid \mathfrak{S}_k \right\} \\
&= -\mathbb{E} \left\{ (b_1(k) - \hat{b}_1(k))^2 u^2(k) \mid \mathfrak{S}_k \right\} \\
&\quad - \mathbb{E} \left\{ 2(b_1(k) - \hat{b}_1(k))u(k)(\mathbf{p}_0^T(k) - \hat{\mathbf{p}}_0^T(k))\mathbf{m}_0(k) \right\} \\
&\quad - \mathbb{E} \left\{ \mathbf{m}_0^T(k)(\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))(\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k) \right\} - \mathbb{E} \left\{ \xi^2(k) \right\} \\
&\quad - \mathbb{E} \left\{ 2(b_1(k) - \hat{b}_1(k))u(k)\xi(k) \right\} \\
&\quad - \mathbb{E} \left\{ 2(\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k)\xi(k) \mid \mathfrak{S}_k \right\} \\
&= -\mathbb{E} \left\{ (b_1(k) - \hat{b}_1(k))^2 u^2(k) \mid \mathfrak{S}_k \right\} \\
&\quad - \mathbb{E} \left\{ 2(b_1(k) - \hat{b}_1(k))u(k)(\mathbf{p}_0(k) - \hat{\mathbf{p}}_0(k))^T \mathbf{m}_0(k) \right\} + \bar{c}_1(k) \\
&= -p_{b_1}(k)u^2(k) - 2\mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)u(k) + \bar{c}_1(k) \tag{3.70}
\end{aligned}$$

In der Variablen  $\bar{c}_1(k)$  sind alle von  $u(k)$  unabhängigen Terme (unterstrichen) zusammengefasst, um die weitere Rechnung übersichtlicher zu halten. Durch Einsetzen der Intervallgrenzen für das Stellsignal ergibt sich nun

$$\begin{aligned}
&J_k^e \left\{ u_c(k) - \theta(k) \right\} - J_k^e \left\{ u(k) + \theta(k) \right\} = \\
&\quad -p_{b_1}(k)(u_c(k) - \theta(k))^2 - 2\mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)(u_c(k) - \theta(k)) + \bar{c}_1(k) \\
&\quad + p_{b_1}(k)(u_c(k) + \theta(k))^2 + 2\mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)(u_c(k) + \theta(k)) - \bar{c}_1(k) \\
&= 4p_{b_1}(k)u_c(k)\theta(k) + 4\mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)\theta(k) \\
&= 4\theta(k)(p_{b_1}(k)u_c(k) + \mathbf{p}_{b_1\mathbf{p}_0}^T(k)\mathbf{m}_0(k)). \tag{3.71}
\end{aligned}$$

Um das endgültige Stellgesetz des dualen Reglers zu erhalten, wird das Ergeb-

nis gemäß Gl. (3.71) in Gl. (3.69) eingesetzt und es gilt

$$u(k) = u_c(k) + \underbrace{\theta(k) \operatorname{sgn} \{p_{b_1}(k)u_c(k) + \mathbf{p}_{b_1 \mathbf{p}_0}^T \mathbf{m}_0(k)\}}_{u_e}. \quad (3.72)$$

Die vorsichtige Stellgröße  $u_c$  ergibt sich aus Gl. (3.68). Geht die Kovarianzmatrix während der Schätzung gegen null, vereinfacht sich das Stellgesetz zu  $u(k) = u_c(k)$  und gemäß Gl. (3.68) erhält man wieder den im Abschnitt 3.1.1.2 beschriebenen Regler.

### 3.2.1.2 Verbesserung des dualen Reglers

Bei der Berechnung des Stellsignals gemäß Gl. (3.72) wurde das Erregungssignal  $u_e(k)$  zum vorsichtigen Stellsignal  $u_c(k)$  addiert, um das Konvergenzverhalten der Parameterschätzung zu verbessern. Dies kann aber bei Störungen oder Modellierungsfehlern zu einer dauernden Verschlechterung des Regelverhaltens führen. Dieses Verhalten kann verbessert werden, indem  $u_e$  mit dem tiefpassgefilterten Betrag des Prädiktionsfehlers

$$\bar{E}(z)(1 - T_f z^{-1}) = (1 - T_f)z^{-1}|E(z)| \quad (3.73)$$

gewichtet wird.  $T_f$  ist ein Koeffizient, der das Zeitverhalten des Filters bestimmt und das verbesserte Stellgesetz ergibt sich zu

$$u(k) = u_c(k) + \theta(k) \operatorname{sgn} \{p_{b_1}(k)u_c(k) + \mathbf{p}_{b_1 \mathbf{p}_0}^T \mathbf{m}_0(k)\} \bar{e}(k). \quad (3.74)$$

Mit dieser Erweiterung sinkt die Amplitude des Erregungssignals bei fallendem Prädiktionsfehler. Des Weiteren kann für einfache Systeme das vorsichtige Stellsignal durch

$$u_c(k+1) = \frac{1}{1 + \lambda|e(k)|} \cdot u_{\text{ND}}(k+1) \quad (3.75)$$

beschrieben werden (Filatov und Unbehauen, 1999). Der Faktor  $\lambda$  gewichtet die Unsicherheit und muss speziell auf die jeweilige Regelstrecke abgestimmt werden.

Um den Reglerentwurf abzuschließen, muss noch das Kompensationsnetz ANN-2 entworfen werden. Das Vorgehen ist aber identisch mit dem im Abschnitt 3.1.1.3 beschriebenen Entwurf.

Die vorgestellten adaptiven Regelungsstrukturen für Hammerstein-Systeme sind flexibel und einfach zu implementieren. Durch die Verwendung von zwei künstlichen neuronalen Netzen zur *online* Identifikation und Kompensation der statischen Eingangsnichtlinearität wird kein Wissen über deren Verlauf beim Reglerentwurf benötigt. Ohne strukturelle Änderungen des Reglers kann dieser somit für Regelstrecken mit verschiedenen Eingangsnichtlinearitäten verwendet werden. Diese Eigenschaft wird im Kapitel 6 ausführlich untersucht. Die zusätzliche Einführung eines vorsichtigen Stellsignals und einer Erregung verbessert das Verhalten des geschlossenen Regelkreises besonders während der Anfangsadaption und bei Parameteränderungen.

## 4 Adaptive Regelung von Wiener-Systemen

Neben dem im Kapitel 3 beschriebenen Hammerstein-System ist das Wiener-System ein häufig verwendetes Modell zur Beschreibung nichtlinearer Systeme mittels Blockstrukturen. Wiener-Systeme bestehen aus einem linearen Übertragungssystem, dem ein statischer, nichtlinearer Block nachgeschaltet ist. Anders als bei Hammerstein-Systemen ist das Wiener-System nicht LIP. Dies erschwert die Parameteridentifikation wesentlich und dürfte der Hauptgrund für die relativ geringe Zahl der Arbeiten zu ihrer adaptiven Regelung sein. Ein Ansatz ist ein adaptiver Modellreferenzregler, bei dem die Nichtlinearität mit einer Spline-Funktion beschrieben wird (Pajunen, 1992; Zhu und Control, 1999). Ohne die Kenntnis über den Verlauf der Nichtlinearität erweist sich die Wahl der Stützstellen allerdings als schwierig und kritisch für diesen Ansatz. Zur Approximation der Ausgangsnichtlinearität werden in einem weiteren Ansatz MLP-ANN eingeführt (Al-Duwaish et al., 1995). Die Parameterbestimmung ist dabei in zwei Phasen aufgeteilt. In der ersten Phase wird ein Eingangssignal kleiner Amplitude aufgeschaltet, um so die Regelstrecke im gleichen Arbeitspunkt zu halten. In dieser Phase werden dann die Parameter des linearen Teilsystems ermittelt. Anschließend wird das Stellsignal vergrößert und mit Hilfe des BP-Algorithmus können die Netzgewichte trainiert werden. Während der Regelung bleiben die Parameter dann unverändert.

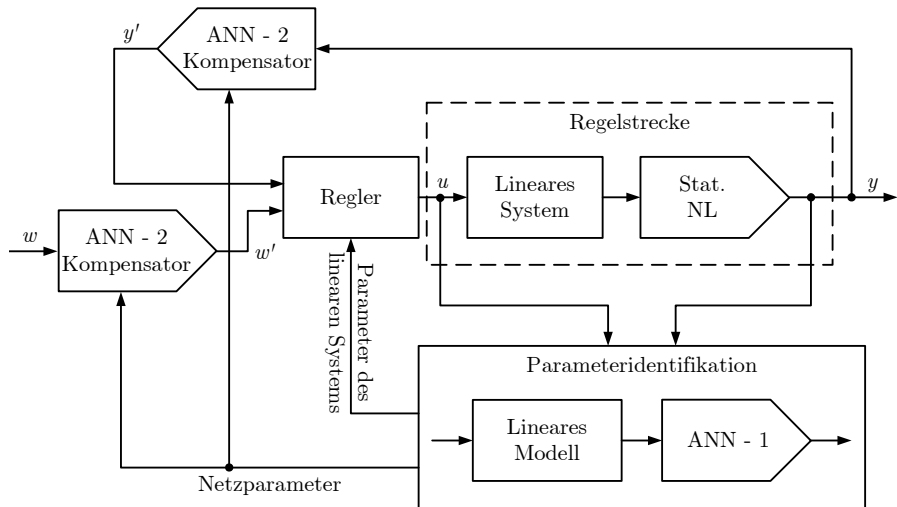
Im Folgenden werden zwei neue Realisierungen einer Regelungsstruktur mit ANN vorgestellt, bei denen sowohl die Parameter des linearen Systems als auch



die Gewichte der Netze *online* angepasst. Darauf basierend werden ein Regler für das lineare Teilsystem und eine Kompensation der Nichtlinearität berechnet. Da das Wiener-System nichtlinear in den Parametern ist, wurden die Parameter bei der ersten Realisierung mit Hilfe des RPE-Verfahrens bestimmt. Um Konvergenzprobleme des RPE-Verfahrens bei ungünstigen Signalen zu vermeiden, wurde bei der zweiten Realisierung das Identifikationsproblem durch direktes Einsetzen eines inversen ANN umformuliert. Man erhält so eine Darstellung der Regelstrecke die LIP ist, wodurch wieder der RLS-Algorithmus zur Parameterschätzung eingesetzt werden kann.

## 4.1 Die WANNNA-Reglerstruktur

Die auf einem Wiener-Modell beruhende ANNNA Reglerstruktur, im Weiteren als WANNNA-Regler bezeichnet, ist wie der HANNNA-Regler ein indirekter, adaptiver Regler bei dem die Identifikation der Regelstreckenparameter und der Reglerentwurf nacheinander durchgeführt werden. Die Regelungsstruktur ist im Bild 4.1 dargestellt. Dem Block Parameteridentifikation wird das Ein- und das Ausgangssignal der Regelstrecke zugeführt. Er liefert mit Hilfe eines geeigneten Schätzverfahrens getrennt die Parameter des linearen Teilsystems und die Gewichte des ANN-1. Das lineare Modell dient in der nächsten Stufe als Basis für den Entwurf eines linearen Reglers. Um die Ausgangsnichtlinearität zu kompensieren wird die Inverse von ANN-1 benötigt. Diese Aufgabe übernimmt ein zweites Netz (ANN-2), welches zweimal in dieser Struktur benötigt wird. Eine Kopie von ANN-2 wird der Regelstrecke nachgeschaltet und somit das Ausgangssignal  $y'$  erzeugt, welches dem Ausgangssignal des linearen Teilsystems entspricht. Mit der zweiten Kopie von ANN-2 wird der Sollwert transformiert. Das sich ergebende Signal  $w'$  ist der entsprechende Sollwert für den Ausgang des linearen Teilsystems. Die beiden transformierten Signale  $w'$  und  $y'$  werden dann dem Regler zugeführt, in dem im einfachsten Fall aus der Differenz  $w' - y'$  das Stellsignal  $u$  berechnet wird. Bei vollständiger Kompensation der statischen Nichtlinearität erhält man also ein lineares Gesamtsystem, welches mittels linearem Regler beherrschbar ist. Eine unvollständige Kompen-



**Bild 4.1:** Strukturbild der WANNNA-Regelung

sation kann als Messrauschen und zusätzlichem Sollwertrauschen interpretiert werden.

Zusammengefasst kann der Entwurf des WANNNA-Reglers mit den folgenden Schritten beschrieben werden:

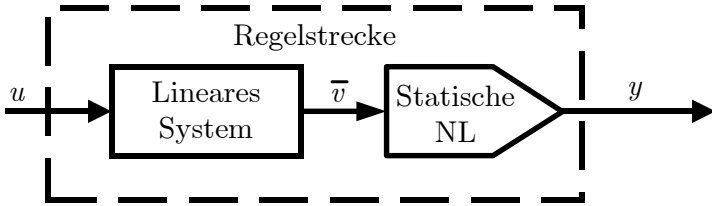
1. Auswahl des Netztyps von ANN-1 und ANN-2 sowie Festlegung der Startkonfiguration.
2. Festlegung der Ordnung des linearen Modells.
3. Auswahl des Algorithmus für die Parameterschätzung.
4. Festlegung des Lernalgorithmus von ANN-1 und ANN-2.
5. Ermittlung der transformierten Signale  $y'$  und  $w'$ .
6. Ermittlung des Stellsignals  $u$ .

Die beschriebene Regelungsstruktur ist modular und unterscheidet sich von der HANNNA-Regelung durch die Art der Kompensation der statischen Nicht-

linearität und der Parameterschätzung. Der Entwurf eines linearen Reglers, das Training des Kompensationsnetzes ANN-2 und die Netzkonfiguration wurden bereits in den Abschnitten 3.1.1.2, 3.1.1.3 und 3.1.2 beschrieben. Im Folgenden wird das Wiener-ANN-Modell definiert und anschließend werden zwei Methoden zur Schätzung der Parameter des linearen Teilsystems und der Netzgewichte beschrieben.

## 4.2 Das Wiener-ANN-Modell

Das Ein-Ausgangsverhalten des im Bild 4.2 dargestellten Wiener-Modells wird im Zeitbereich durch die Differenzengleichung



**Bild 4.2:** Wiener-Modell

$$\bar{v}(k) = \mathcal{J}^{-1} \left\{ \frac{B(z^{-1})}{A(z^{-1})} U(z) \right\} = - \sum_{i=1}^{n_{\bar{v}}} a_i \bar{v}(k-i) + \sum_{i=1}^{n_u} b_i u(k-i) \quad (4.1)$$

und die statische Ausgangskennlinie

$$y(k) = f(\bar{v}(k)) \quad (4.2)$$

beschrieben. Die Signale  $u(k)$ ,  $\bar{v}(k)$  und  $y(k)$  sind das Eingangs-, das nicht messbare interne und das Ausgangssignal der Regelstrecke. Die Polynome  $B(z^{-1})$  und  $A(z^{-1})$  sind bis auf die Ordnungen  $n_{\bar{v}}$  und  $n_u$  ebenfalls unbekannt und beschreiben das Verhalten des linearen Teilsystems. Der Operator  $\mathcal{J}^{-1}$  steht für die inverse z-Transformation. Bei den meisten Verfahren zur adaptiven Regelung von Wiener-Systemen wird eine bestimmte Struktur der

statischen Ausgangsnichtlinearität, wie beispielsweise eine Titrationskennlinie, eine Totzone oder eine Getriebelose vorausgesetzt und deren Parameter ermittelt. Diese Vereinfachung gilt in der Praxis nicht, weshalb bei dem vorgestellten Verfahren ein RBF-Netz als universeller Approximator zur Beschreibung der Nichtlinearität eingesetzt wird. Der Aufbau dieser Netze wurde bereits im Abschnitt 2.6.1 diskutiert. Eingesetzt in Gl. (4.2) ist das Ein-Ausgangsverhalten der Nichtlinearität dann durch

$$y(k) = w_0 \bar{v}(k) + \sum_{i=1}^m w_i \Phi_i(|\bar{v}(k) - \mathbf{c}_i|) \quad (4.3)$$

gegeben. Die Funktion  $\Phi_i(|\bar{v} - \mathbf{c}_i|)$  steht für eine frei wählbare radiale Basisfunktion. Die Netzgewichte  $w_i, i = 0, \dots, m$  sind die zu adaptierenden Parameter dieser Struktur. Der Einfachheit halber wird zunächst angenommen, dass die Zentren  $\mathbf{c}_i, i = 1, \dots, m$  und die Formparameter  $\mathbf{w}_i, i = 1, \dots, m$  zu Anfang der Regelung fest eingestellt werden. Bei dem beschriebenen System wurde bewusst auf die Einführung eines Störmodells verzichtet, da dies nicht zum Verständnis der vorgestellten adaptiven Regelungsstruktur beiträgt und die folgenden Berechnungen unnötig komplizieren würde. Prinzipiell kann aber bei Bedarf ein Störmodell berücksichtigt werden.

## 4.3 Parameteridentifikation mit dem RPE-Verfahren

Das mit den Gln. (4.1) und (4.3) beschriebene System ist nicht mehr linear in den unbekannten Parametern  $a_i, b_i$  und  $w_i$ . Deshalb ist es nicht möglich, das Identifikationsproblem mit dem LSE- oder RLS-Verfahren zu lösen, welche für lineare Systeme entwickelt wurden.

Das rekursive Prädiktionsfehlerverfahren (Ljung und Söderström, 1986) ist ein Verfahren, das zur Parameteridentifikation des Wiener-Modells verwendet werden kann (Kortmann und Unbehauen, 1986). Es basiert auf der Minimie-

nung der Summe der Prädiktionsfehlerquadrate

$$I_k(\mathbf{p}) = \frac{1}{2} \sum_i^k e^2(i, \mathbf{p}) \stackrel{!}{=} \text{Min}, \quad (4.4)$$

wobei der Prädiktionsfehler  $e(i)$  die Differenz zwischen dem prädizierten Ausgangssignal  $\hat{y}(i)$  und dem gemessenen Ausgangssignal  $y(i)$  ist. Die best-mögliche Prädiktion des Ausgangssignals zum Zeitpunkt  $t_{k+1}$  berechnet sich aus den Modellgleichungen (4.1) und (4.3), wobei für die Parameter deren aktuelle Schätzungen, zum Zeitpunkt  $t_k$ , eingesetzt werden. Bei Berücksichtigung einiger Näherungen (Ljung und Söderström, 1986) führt die Minimierung der Gütefunktion (Gl. (4.4)) zu einem rekursiven Schätzalgorithmus, der durch die vier Grundgleichungen

$$e(k) = y(k) - \hat{y}(k) \quad (4.5a)$$

$$\mathbf{q}(k) = \mathbf{P}(k-1)\mathbf{\Psi}(k) \left[ 1 + \mathbf{\Psi}^T(k)\mathbf{P}(k-1)\mathbf{\Psi}(k) \right]^{-1} \quad (4.5b)$$

$$\hat{\mathbf{p}}(k) = \hat{\mathbf{p}}(k-1) + \mathbf{q}(k)e(k) \quad (4.5c)$$

$$\mathbf{P}(k)\mathbf{q} = \frac{1}{\rho(k)} \left[ \mathbf{P}(k-1) - \mathbf{q}(k)\mathbf{\Psi}^T(k)\mathbf{P}(k-1) \right] \quad (4.5d)$$

beschrieben ist. Die Gln. (4.5) stimmen formal mit den Gleichungen des Standard-RLS-Algorithmus überein. Der Hauptunterschied besteht in der Bedeutung des Vektors  $\mathbf{q}(k)$  und der Matrix  $\mathbf{P}(k)$ . Bei deren Berechnung wird nun nicht ein Datenvektor verwendet, sondern der Gradientenvektor  $\mathbf{\Psi}(k)$ , der sich aus der Ableitung des prädizierten Ausgangssignals nach den zu schätzenden Parametern des Modells ergibt

$$\mathbf{\Psi}(k) = \frac{\partial \hat{y}(k)}{\partial \hat{\mathbf{p}}}. \quad (4.6)$$

Mit dem Gewichtungsfaktor  $\rho(k)$  kann die Konvergenzgeschwindigkeit variiert werden, was die Berücksichtigung von Regelstrecken mit zeitvarianten Parametern ermöglicht. Er kann sowohl konstant als auch zeitvariant sein.

Werden nun die unbekannten Parameter des Modells in dem Schätzvektor

$$\hat{\mathbf{p}}^T(k) = [\hat{w}_0 \dots \hat{w}_m, \hat{a}_1 \dots \hat{a}_{n_{\bar{v}}}, \hat{b}_1 \dots \hat{b}_{n_u}] \quad (4.7)$$

zusammengefasst, ergibt sich der Gradientenvektor zu:

$$\frac{\partial \hat{y}(k)}{\partial \hat{w}_0} = \bar{v}(k) \quad (4.8a)$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{w}_j} = \Phi_j(|\bar{v}(k) - \mathbf{c}_j|), \quad j = 1, \dots, m \quad (4.8b)$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{a}_j} = \left[ \hat{w}_0 + \sum_{i=1}^m \hat{w}_i \frac{\partial \Phi_i}{\partial |\bar{v}(k) - \mathbf{c}_i|} \frac{\partial |\bar{v}(k) - \mathbf{c}_i|}{\partial \bar{v}(k)} \right] \frac{\partial \bar{v}(k)}{\partial \hat{a}_j}, \quad j = 1, \dots, n_{\bar{v}} \quad (4.8c)$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{b}_j} = \left[ \hat{w}_0 + \sum_{i=1}^m \hat{w}_i \frac{\partial \Phi_i}{\partial |\bar{v}(k) - \mathbf{c}_i|} \frac{\partial |\bar{v}(k) - \mathbf{c}_i|}{\partial \bar{v}(k)} \right] \frac{\partial \bar{v}(k)}{\partial \hat{b}_j}, \quad j = 1, \dots, n_u \quad (4.8d)$$

mit

$$\frac{\partial \bar{v}(k)}{\partial \hat{a}_j} = -\bar{v}(k-j) - \sum_{i=1}^{n_{\bar{v}}} \hat{a}_i \frac{\partial \bar{v}(k-i)}{\partial \hat{a}_j}, \quad j = 1, \dots, n_{\bar{v}} \quad (4.9)$$

$$\frac{\partial \bar{v}(k)}{\partial \hat{b}_j} = u(k-j) - \sum_{i=1}^{n_{\bar{v}}} \hat{a}_i \frac{\partial \bar{v}(k-i)}{\partial \hat{b}_j}, \quad j = 1, \dots, n_u. \quad (4.10)$$

Die Berechnung des Gradientenvektors mit Hilfe der Gln. (4.9) und (4.10) erfordert einen enormen Rechen- und Speicheraufwand, der jedoch drastisch reduziert werden kann. Nach Substitution von  $k-l+1$  durch  $k$  in Gl. (4.9) erhält man für  $j=1$

$$\frac{\partial \bar{v}(k-l+1)}{\partial \hat{a}_1} = -\bar{v}(k-l) - \sum_{i=1}^{n_{\bar{v}}} \hat{a}_i \frac{\partial \bar{v}(k-l+1-i)}{\partial \hat{a}_1}, \quad l = 2, \dots, n_{\bar{v}}. \quad (4.11)$$

Werden Gl.(4.9) und Gl.(4.11) verglichen, folgt die Beziehung

$$\frac{\partial \bar{v}(k)}{\partial \hat{a}_l} = \frac{\partial \bar{v}(k-l+1)}{\partial \hat{a}_1}, \quad l = 2, \dots, n_{\bar{v}} \quad (4.12)$$

und ganz analog für die Ableitungen nach  $\hat{b}_l$

$$\frac{\partial \bar{v}(k)}{\partial \hat{b}_l} = \frac{\partial \bar{v}(k-l+1)}{\partial \hat{b}_1}, \quad l = 2, \dots, n_{\bar{v}}. \quad (4.13)$$

Bei Benutzung der Gln. (4.12) und (4.13) zur Berechnung des Gradientenvektors  $\Psi(k)$  werden die  $2n_{\bar{v}}$  Ableitungen des internen Signals  $\bar{v}(k)$  nach den Modellparametern durch eine einfache Zeitverschiebung der Ableitung des Signals  $\bar{v}(k)$  nach  $\hat{a}_1$  und  $\hat{b}_1$  ersetzt. Zusammen mit der partiellen Ableitung der Gaußfunktion als radiale Basisfunktion und bei Verwendung der Euklidischen Norm

$$\frac{\partial \Phi_i}{\partial |\bar{v}(k) - \mathbf{c}_i|} \frac{\partial |\bar{v}(k) - \mathbf{c}_i|}{\partial \bar{v}(k)} = -\frac{2}{\mathfrak{w}_i^2} (\bar{v}(k) - \mathbf{c}_i) \exp \left( -\frac{(\bar{v}(k) - \mathbf{c}_i)^2}{\mathfrak{w}_i^2} \right) \quad (4.14)$$

kann der Gradientenvektor wie folgt geschrieben werden:

$$\frac{\partial \hat{y}(k)}{\partial \hat{w}_0} = \bar{v}(k) \quad (4.15a)$$

$$\frac{\partial \hat{y}(k)}{\partial \hat{w}_j} = \exp \left( -\frac{(\bar{v}(k) - \mathbf{c}_j)^2}{\mathfrak{w}_j^2} \right), \quad j = 1, \dots, m \quad (4.15b)$$

$$\begin{aligned} \frac{\partial \hat{y}(k)}{\partial \hat{a}_j} &= \left[ w_0 + \sum_{i=1}^m -\frac{2w_i}{\mathfrak{w}_i^2} (\bar{v}(k) - \mathbf{c}_i) \exp \left( -\frac{(\bar{v}(k) - \mathbf{c}_i)^2}{\mathfrak{w}_i^2} \right) \right] \frac{\partial \bar{v}(k - j + 1)}{\partial \hat{a}_1}, \\ j &= 1, \dots, n_{\bar{v}} \end{aligned} \quad (4.15c)$$

$$\begin{aligned} \frac{\partial \hat{y}(k)}{\partial \hat{b}_j} &= \left[ w_0 + \sum_{i=1}^m -\frac{2w_i}{\mathfrak{w}_i^2} (\bar{v}(k) - \mathbf{c}_i) \exp \left( -\frac{(\bar{v}(k) - \mathbf{c}_i)^2}{\mathfrak{w}_i^2} \right) \right] \frac{\partial \bar{v}(k - j + 1)}{\partial \hat{b}_1}, \\ j &= 1, \dots, n_u. \end{aligned} \quad (4.15d)$$

Zur Berechnung des Gradienten und des geschätzten Ausgangssignals werden neben dem bekannten Eingangssignal auch das interne Signal  $\bar{v}(k)$  und die Parameter des Systems benötigt. Da deren wahren Werte jedoch unbekannt sind, werden ihre Schätzungen verwendet. Dieses Vorgehen entspricht dem Gewissheitsprinzip (*certainty equivalence*), bei dem eine ausreichend hohe Qualität der Schätzung angenommen wird. Die Konvergenz des RPE-Verfahrens für das Wiener-Modell wurde ausführlich diskutiert (Kortmann, 1988). Zusammengefasst ist sie gesichert, wenn die folgenden Bedingungen erfüllt sind:

1. Der Prädiktor muss stabil sein.

2. Der Prädiktor muss nach den Parametern des Modells differenzierbar sein.

Diese Konvergenzbedingungen sind erfüllt, wenn die Nullstellen des Polynoms  $A(z^{-1})$  im Einheitskreis liegen. Während der Adaption kann dies sichergestellt werden, indem nur ein neuer Parametervektor bestimmt wird, wenn das dazugehörige Modell stabil ist. Neben diesen Bedingungen sind weitere angegeben (Ljung und Söderström, 1986), die sich aber auf Eigenschaften wie beispielsweise die Signalqualität beziehen und unabhängig vom Modell sind.

## 4.4 Parameteridentifikation mit dem RLS-Verfahren

Das RPE-Verfahren zeigt in der Praxis Konvergenzprobleme bei der Identifikation im geschlossenen Regelkreis. Das kann hauptsächlich mit der ungenügenden Qualität der zur Schätzung benötigten Signale  $u(k)$  und  $y(k)$  begründet werden. Weder das Ein- noch das Ausgangssignal der Regelstrecke können direkt vorgegeben werden. Eine mögliche Lösung dieses Problems ist das direkte Einsetzen des Kompensationsnetzes ANN-2 in den Einschrittprädiktor. Man erhält mit diesem Vorgehen ein Schätzproblem, das LIP ist, weshalb beispielsweise der RLS-Algorithmus verwendet werden kann.

Das Modell des Wiener-Systems ist gemäß Abschnitt 4.2 durch die Differenzengleichung

$$\bar{v}(k) = - \sum_{i=1}^{n_{\bar{v}}} a_i \bar{v}(k-i) + \sum_{i=1}^{n_u} b_i u(k-i) \quad (4.16)$$

und die statische Nichtlinearität

$$y(k) = f(\bar{v}(k)) \quad (4.17)$$

gegeben. Aus Bild 4.1 geht hervor, dass zur Implementierung des WANNNA-Reglers nicht das ANN-1 sondern das Kompensationsnetz ANN-2 zur Beschreibung der inversen Nichtlinearität benötigt wird. Dieses ist mit Hilfe eines RBF-



ANN durch

$$\bar{v}(k) = f^{-1}(y(k)) = w_0 y(k) + \sum_{i=1}^m w_i \Phi_i(|y(k) - \mathbf{c}_i|) \quad (4.18)$$

beschrieben. Existiert die Inverse in der realen Regelstrecke nicht, schätzt das Netz eine Näherung, die im Sinne des gewählten Schätzalgorithmus optimal ist. Das Einsetzen von Gl. (4.18), mit  $w_0 = 1$ , in Gl. (4.16) führt zu

$$\begin{aligned} y(k) + \sum_{i=1}^m w_i \Phi_i(|y(k) - \mathbf{c}_i|) = & - \sum_{i=1}^{n_{\bar{v}}} a_i y(k-i) \\ & - \sum_{i=1}^{n_{\bar{v}}} \sum_{j=1}^m a_i w_j \Phi_j(|y(k-i) - \mathbf{c}_j|) + \sum_{i=1}^{n_u} b_i u(k-i) \end{aligned} \quad (4.19)$$

oder nach  $y(k)$  aufgelöst zu

$$\begin{aligned} y(k) = & - \sum_{i=1}^{n_{\bar{v}}} a_i y(k-i) + \sum_{i=1}^{n_u} b_i u(k-i) - \sum_{j=1}^m w_j \Phi_j(|y(k) - \mathbf{c}_j|) \\ & - \sum_{i=1}^{n_{\bar{v}}} \sum_{j=1}^m a_i w_j \Phi_j(|y(k-i) - \mathbf{c}_j|). \end{aligned} \quad (4.20)$$

Diese Modelldarstellung ist nur noch von den messbaren Signalen  $u(k)$  und  $y(k)$  abhängig und wird im Folgenden als Einschrittpädiktor verwendet. Ähnlich wie im Abschnitt 3.1.1.1 werden die Hilfsparameter

$$\beta_{0j} = w_j \quad (4.21)$$

$$\beta_{ij} = a_i w_j, \quad i = 1, \dots, n_{\bar{v}}, \quad j = 1, \dots, m \quad (4.22)$$

und die Hilfssignale

$$x_j(k) = \Phi_j(|y(k) - \mathbf{c}_j|), \quad j = 1, \dots, m \quad (4.23)$$

zur vereinfachten Darstellung eingeführt. Mit diesen Definitionen ergibt sich Gl. (4.20) zu

$$y(k) = - \sum_{i=1}^{n_{\bar{v}}} a_i y(k-i) + \sum_{i=1}^{n_u} b_i u(k-i) - \sum_{i=0}^{n_{\bar{v}}} \sum_{j=1}^m \beta_{ij} x_j(k-i), \quad (4.24)$$

und die Einführung des Datenvektors

$$\begin{aligned} \mathbf{m}^T(k) = & [-y(k-1) \dots -y(k-n), u(k-1) \dots u(k-n_u), \\ & -x_1(k) \dots -x_1(k-n) \dots -x_m(k) \dots -x_m(k-n)] \end{aligned} \quad (4.25)$$

sowie des Parametervektors

$$\mathbf{p} = [a_1 \dots a_{n_{\bar{v}}}, b_1 \dots b_{n_u}, \beta_{01} \dots \beta_{n_{\bar{v}}1} \dots \beta_{0m} \dots \beta_{n_{\bar{v}}m}]^T \quad (4.26)$$

ermöglicht die kompakte Schreibweise von Gl. (4.20)

$$y(k) = \mathbf{m}^T(k)\mathbf{p}. \quad (4.27)$$

Das Identifikationsproblem wurde somit auf eine Standardform transformiert, für die nun bekannte Methoden, wie beispielsweise der im Abschnitt 3.1.1.1 beschriebene RLS-Algorithmus, zur Parameterschätzung benutzt werden können. Gemäß Gl. (4.26) ergeben sich die Parameter des linearen Teilsystems direkt aus dem Schätzvektor  $\hat{\mathbf{p}}$ . Somit gilt für die Parameter des Nennerpolynoms des linearen Teilsystems

$$\hat{a}_i = \hat{p}_i, \quad i = 1, \dots, n_{\bar{v}} \quad (4.28)$$

und für die Parameter des Zählerpolynoms

$$\hat{b}_i = \hat{p}_{n_{\bar{v}}+i}, \quad i = 1, \dots, n_u. \quad (4.29)$$

Zur Bestimmung der Netzwichte muss für jedes Gewicht  $w_i, i = 1, \dots, m$  das überbestimmte Gleichungssystem

$$\begin{aligned} \hat{\beta}_{0j} &= 1\hat{w}_j \\ \hat{\beta}_{1j} &= \hat{a}_1\hat{w}_j \\ \hat{\beta}_{2j} &= \hat{a}_2\hat{w}_j \\ &\vdots \\ \hat{\beta}_{n_{\bar{v}}j} &= \hat{a}_{n_{\bar{v}}}\hat{w}_j, \end{aligned} \quad (4.30)$$

gelöst werden. Eine Möglichkeit dieses Problem zu lösen, ist die lineare Regression bezüglich der Netzgewichte, für die sich die Lösung wie folgt berechnet:

$$\hat{w}_j = \frac{\hat{\beta}_{0j} + \sum_{i=1}^{n_{\bar{v}}} \hat{a}_i \hat{\beta}_{ij}}{1 + \sum_{i=1}^{n_{\bar{v}}} \hat{a}_i^2}, \quad j = 1, \dots, m. \quad (4.31)$$

Die Berechnung der Netzgewichte anhand der Gl. (4.31) beruht auf der Annahme, dass die geschätzten Parameter des linearen Teilsystems mit den tatsächlichen Parametern übereinstimmen (Gewissheitsprinzip). Die gleichzeitige Berechnung der voneinander abhängigen Parameter  $\hat{a}_i$  und  $\hat{w}_j$  ist ebenfalls möglich (Bai, 1998). Diese Berechnung erfordert jedoch in jedem Abtastschritt die Singulärwert-Zerlegung einer Matrix der Dimension  $n_{\bar{v}} \times m$  und ist deshalb für die Realisierung in Echtzeit auf Systeme niedriger Ordnung und kleine Netze beschränkt.

Bei der vorgestellten Regelungsstruktur für Wiener-Systeme wird nur geringes Wissen über die Regelstrecke für den Entwurf benötigt. Dies ist hauptsächlich die Ordnung des linearen Teilmodells, welche durch Strukturprüfverfahren (Unbehauen, 1988) im Vorhinein ermittelbar ist. Durch die Integration dieses Wissens in die Regelungsstruktur erhält man mit Netzen, die aus weniger als 30 Neuronen in der versteckten Schicht aufgebaut sind, gute Regelungsergebnisse. Dies wird in den Abschnitten 6.4 und 6.5 anhand von Simulationen gezeigt. Durch diesen kompakten Aufbau ist die einfache Implementierung eines WANNNA-Reglers in einen Mikroregler (*Microcontroller*) möglich und somit deren Umsetzung für reale Problemstellungen gegeben.

## 5 Das erweiterte *Backstepping*-Verfahren

Das im Abschnitt 2.1 beschriebene Syntheseverfahren bietet die Möglichkeit, für Systeme in der PSF-Struktur, adaptive Regelungen sehr systematisch zu entwerfen. Hierbei werden sowohl im Stellgesetz als auch im Adaptionsgesetz die nichtlinearen Funktionen des Systems benötigt. Es ist somit eine Grundvoraussetzung, dass die Nichtlinearitäten bekannt und konstant sind, und dass die unbekannten Parameter linear mit ihnen verknüpft sind. Dies ist aber bei realen Anlagen selten der Fall. Die nichtlinearen Funktionen sind hier im Allgemeinen ebenfalls unbekannt und können nur grob abgeschätzt werden. Desweiteren findet man nur in Ausnahmefällen eine lineare Parametrisierung vor.

Um das Problem der linearen Parametrisierung zu lösen, wird beispielsweise die Einführung einer Taylor-Reihe erster Ordnung zu Approximation der Nichtlinearitäten vorgeschlagen (Karsenti et al., 1996), wobei jedoch die Kenntnis über deren Verlauf wesentlich für diesen Entwurf ist. Für Regelstrecken mit begrenzten Unsicherheiten wurde ein robustes *Backstepping*-Verfahren vorgeschlagen (Freeman et al., 1996). Beide Erweiterungen können bei unbekannten und/oder zeitvarianten Nichtlinearitäten nur sehr begrenzt verwendet werden.

Im Folgenden wird eine Erweiterung des *Backstepping*-Verfahrens vorgeschlagen, mit der es möglich ist, die beschriebenen Probleme zu lösen (Knohl und Unbehauen, 2000b). Zusätzlich können Regelstrecken in der neu definierten nichtlinearen PSF-Struktur (NPSF) geregelt werden. Ausgangspunkt für diese Erweiterung ist die NPSF-Struktur, wobei aber für die nichtlinea-

ren Funktionen jeweils ein ANN angesetzt wird. Anders als bei Standard-Regelverfahren mit ANN, wird also nicht ein einziges großes ANN zum Erlernen des Gesamtverhaltens der Regelstrecke verwendet. Durch die Berücksichtigung der Regelstreckenstruktur erhält man wieder mehrere kleine Netze, welche die im Abschnitt 2.6.1 beschriebenen Vorteile bezüglich ihrer Initialisierung besitzen. Für die Synthese des Reglers wird nur die Ordnung der Regelstrecke benötigt und basierend auf dem *Backstepping*-Verfahren werden das Stellgesetz und die Lernregeln für die Gewichte der ANN ermittelt.

## 5.1 ANN *Backstepping*-Regelung

Die im Abschnitt 2.1 definierte PSF-Struktur wird im Folgenden zur nichtlinearen PSF-Struktur entsprechend

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \varphi_1(x_1, \mathbf{p}), \\
 \dot{x}_2 &= x_3 + \varphi_2(x_1, x_2, \mathbf{p}), \\
 &\vdots \\
 \dot{x}_{n-1} &= x_n + \varphi_{n-1}(x_1, \dots, x_{n-1}, \mathbf{p}), \\
 \dot{x}_n &= \beta(\mathbf{x})u + \varphi_n(\mathbf{x}, \mathbf{p})
 \end{aligned} \tag{5.1}$$

erweitert. Bei dieser Struktur sind die nichtlinearen Funktionen  $\varphi_i, i = 1, \dots, n$ , abhängig vom unbekannten Parametervektor  $\mathbf{p}$ . Im Vergleich zur PSF-Struktur, mit linearer Parametrisierung, bedeutet dies eine starke Erweiterung der Systemklasse, für die mit dem *Backstepping*-Verfahren Regelungen entworfen werden können.

Im Weiteren wird nun angenommen, dass neben dem Parametervektor  $\mathbf{p}$  ebenfalls die Funktionen  $\varphi_i$  unbekannt sind. Aus diesem Grund werden zu ihrer Approximation  $n$  ANN eingesetzt und somit ergibt sich die Zustandsraumdar-

stellung der NPSF-Struktur durch diese Erweiterung zu

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \left[ p_{0,1} + \sum_{i=1}^{m_1} p_{i,1} \Phi_{1,i}(|x_1(k) - \mathbf{c}_{i,1}|) \right] + \tilde{\varphi}_1(x_1, \mathbf{p}), \\
 \dot{x}_2 &= x_3 + \left[ p_{0,2} + \sum_{i=1}^{m_2} p_{i,2} \Phi_{2,i}(\|\mathbf{x}_2(k) - \mathbf{c}_{i,2}\|) \right] + \tilde{\varphi}_2(\mathbf{x}_2, \mathbf{p}), \\
 &\vdots \\
 \dot{x}_{n-1} &= x_n + \left[ p_{0,n-1} + \sum_{i=1}^{m_{n-1}} p_{i,n-1} \Phi_{n-1,i}(\|\mathbf{x}_{n-1}(k) - \mathbf{c}_{i,n-1}\|) \right] \\
 &\quad + \tilde{\varphi}_{n-1}(\mathbf{x}_{n-1}, \mathbf{p}), \\
 \dot{x}_n &= \beta(\mathbf{x})u + \left[ p_{0,n} + \sum_{i=1}^{m_n} p_{i,n} \Phi_{n,i}(\|\mathbf{x}_n(k) - \mathbf{c}_{i,n}\|) \right] + \tilde{\varphi}_n(\mathbf{x}_n, \mathbf{p}).
 \end{aligned} \tag{5.2}$$

Die Terme in den eckigen Klammern in den Gln. (5.2) beschreiben hierbei das Ein-Ausgangsverhalten von RBF-Netzen (siehe Abschnitt 2.6.1),

$$\tilde{\varphi}_\nu() = \varphi_\nu() - \left[ p_{0,\nu} + \sum_{i=1}^{m_\nu} p_{i,\nu} \Phi_{\nu,i}(\|\mathbf{x}_\nu(k) - \mathbf{c}_{i,\nu}\|) \right] \quad \nu = 1, \dots, n \tag{5.3}$$

sind die  $n$  Approximationsfehler und

$$\mathbf{x}_i = [x_1 \dots x_i]^\text{T}, \quad i = 2, \dots, n \tag{5.4}$$

sind die zu  $\varphi_i$  gehörigen Zustände des Systems, die bei dieser Struktur als bekannt angenommen werden. Die Zentren  $\mathbf{c}_i$  der Basisfunktionen haben die Dimension der entsprechenden Zustandsvektoren  $\mathbf{x}_i$ . Für den Spezialfall  $\beta(\mathbf{x}) = 1$  und  $\tilde{\varphi}_\nu() = 0, \nu = 1, \dots, n$  ist das Blockdiagramm dieser nichtlinearen Struktur im Bild 5.1 dargestellt. Sie besteht aus einer Integratorkette und  $n$  ANN. Die Ableitungen der Zustände  $\mathbf{x}_i$  berechnen sich aus der Addition des  $i$ -ten ANN und dem  $i+1$ -ten Zustand. Das Eingangssignal des  $i$ -ten ANN setzt sich aus den Zuständen eins bis  $i$  zusammen.

Mit den Gln. (5.2) ist ein allgemeines System in NPSF-Form beschrieben, welches sich, ohne strukturelle Änderungen, durch Parameteradaption an verschiedene Regelstrecken der Ordnung  $n$  anpassen kann. In dieser Struktur sind die folgenden Parameter zu bestimmen oder müssen angepasst werden.

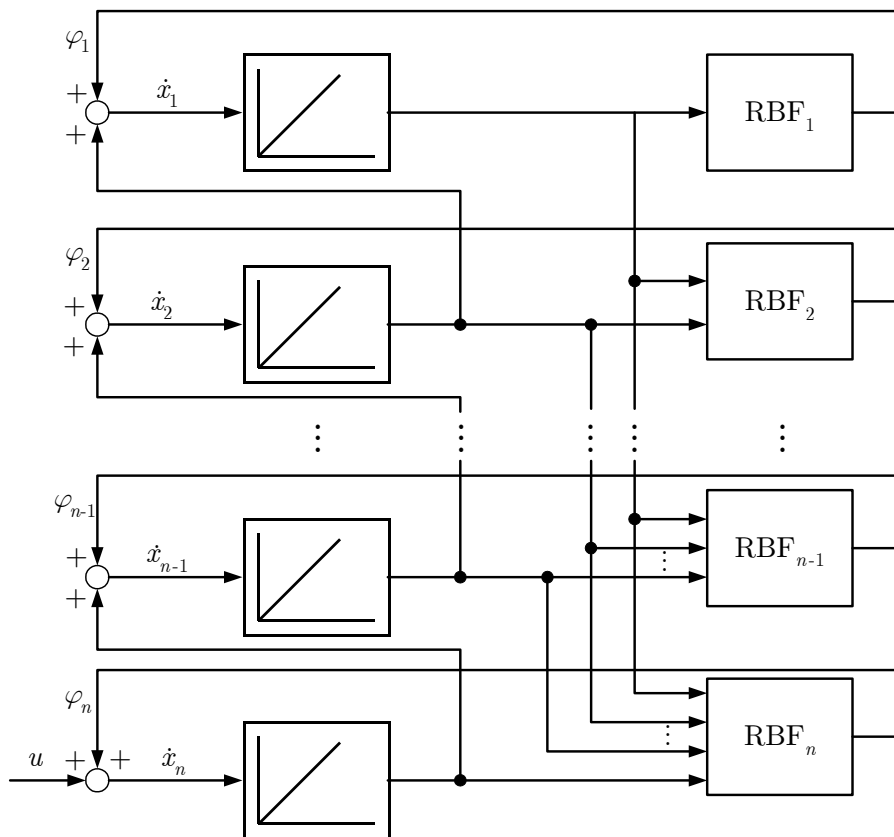


Bild 5.1: Ein allgemeines NPSF-System mit RBF-Netzen

1. Der Verlauf der Basisfunktionen  $\Phi_{\nu,i}$ ,
2. die Anzahl der Basisfunktionen für jedes ANN  $m_\nu$ ,
3. die Position der Zentren  $\mathbf{c}_{i,\nu}$ ,
4. die Formparameter der Basisfunktionen  $\mathbf{w}$  und
5. der Parametervektor  $\mathbf{p}_\nu$ .

Die Problematik der Optimierung der Initialisierungsparameter (Parameter eins bis vier) wurde schon in den Abschnitten 2.6.1 und 3.1.2 diskutiert. Bei der vorgeschlagenen Struktur ist zu beachten, dass die Wahl der Zentren der Basisfunktionen mit steigender Systemordnung immer schwieriger wird. Anders als beim HANNNA- oder WANNNA-Regler, steigt die Ordnung der Zentren mit der des Systems. Zur Bestimmung der Initialisierungsparameter wurden verschiedene Verfahren vorgeschlagen (Pottmann und Seborg, 1992; Liang und ElMaraghy, 1993). Gerade bei der adaptiven Regelung werden sie aber häufig bei ersten Simulationen optimiert und dann während des Experiments fixiert. Eine vernünftige Wahl dieser Parameter vorausgesetzt, sind nur noch die Netzgewichte, repräsentiert durch den Parametervektor  $\mathbf{p}$ , zu optimieren. Durch die Erweiterung der NPSF-Struktur mit ANN ist das resultierende System gemäß den Gln. (5.2) LIP, weshalb das adaptive *Backstepping*-Verfahren zum Reglerentwurf benutzt werden soll. Hierzu muss die gegebene Systembeschreibung nach den Gln. (5.2) umgeschrieben werden zu

$$\begin{aligned}
 \dot{x}_1 &= x_2 + \Phi_1^T \mathbf{p} + \tilde{\varphi}_1(x_1, \mathbf{p}), \\
 \dot{x}_2 &= x_3 + \Phi_2^T \mathbf{p} + \tilde{\varphi}_2(x_1, x_2, \mathbf{p}), \\
 &\vdots \\
 \dot{x}_{n-1} &= x_n + \Phi_{n-1}^T \mathbf{p} + \tilde{\varphi}_{n-1}(x_1, \dots, x_{n-1}, \mathbf{p}), \\
 \dot{x}_n &= \beta(\mathbf{x})u + \Phi_n^T \mathbf{p} + \tilde{\varphi}_n(\mathbf{x}, \mathbf{p}),
 \end{aligned} \tag{5.5}$$

mit  $\mathbf{p}$  und  $\Phi_\nu^T$  wie folgt definiert

$$\mathbf{p}^T = [\mathbf{p}_1^T \dots \mathbf{p}_\nu^T] \tag{5.6}$$



und

$$\Phi_\nu^T = \left[ \mathbf{0}_{(1 \times \sum_{j=0}^{\nu-1} m_j)}^T | 1 \Phi_{\nu,1} \dots \Phi_{\nu,m_\nu} | \mathbf{0}_{(1 \times \sum_{j=\nu+1}^n m_j)}^T \right], m_0 = 0. \quad (5.7)$$

Das mit Gl. (5.5) beschriebene NPSF-System, wobei diese Beschreibung äquivalent zu der in Gl. (5.1) ist, kann mit einem *Backstepping*-Regler, der durch das Stellgesetz aus Gl. (2.6), dem Parameteradaptionsgesetz aus Gl. (2.7) und den Definitionen gemäß den Gln. (2.8) und (2.9) beschrieben ist, stabilisiert werden, so dass die Beschränktheit aller Systemsignale und Reglerparameter garantiert ist.

## 5.2 Stabilität der erweiterten *Backstepping*-Regelung

Der Beweis der Stabilität der erweiterten *Backstepping*-Regelung für NPSF-Systeme basiert auf dem Vorgehen bei PSF-Systemen (Krstić et al., 1995). Gemäß Gl. (2.8) kann die Ableitung der Fehlervariablen  $z_i, i = 1, \dots, n$  wie folgt geschrieben werden

$$\begin{aligned} \dot{z}_i &= \dot{x}_i - \dot{\alpha}_{i-1} \\ &= x_{i+1} + \Phi_i^T \mathbf{p} + \tilde{\varphi}_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} \dot{x}_j - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial \hat{p}_j} \dot{\hat{p}}_j \\ &= \alpha_i + z_{i+1} + \Phi_i^T \mathbf{p} + \tilde{\varphi}_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} (x_{j+1} + \Phi_j^T \mathbf{p}) \\ &\quad - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial \hat{p}_j} \Gamma(\Phi_j - \sum_{k=1}^{j-1} \frac{\partial \alpha_{j-1}}{\partial x_k} \Phi_k) z_j \\ &= \alpha_i + z_{i+1} + \tilde{\varphi}_i + \left( \Phi_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} \Phi_j \right)^T \mathbf{p} \\ &\quad - \sum_{j=1}^{i-1} \left[ \frac{\partial \alpha_{i-1}}{\partial x_j} x_{j+1} + \frac{\partial \alpha_{i-1}}{\partial \hat{p}_j} \Gamma(\Phi_j - \sum_{k=1}^{j-1} \frac{\partial \alpha_{j-1}}{\partial x_k} \Phi_k) z_j \right]. \end{aligned} \quad (5.8)$$

Definiert man nun

$$x_{n+1} = \beta(\mathbf{x})u, \quad (5.9)$$

$$\boldsymbol{\omega}_1 = \boldsymbol{\Phi}_1 \quad (5.10)$$

und

$$\boldsymbol{\omega}_i = \boldsymbol{\Phi}_i - \sum_{j=1}^{i-1} \frac{\partial \alpha_{i-1}}{\partial x_j} \boldsymbol{\Phi}_j, \quad i = 2, \dots, n, \quad (5.11)$$

ergibt sich durch Einsetzen der Stabilisierungsfunktion  $\alpha_i$  gemäß Gl. (2.9) das Fehlersystem (Gln. (5.8)) zu

$$\dot{z}_i = -c_i z_i - z_{i-1} + z_{i+1} + \boldsymbol{\omega}_i^T (\mathbf{p} - \hat{\mathbf{p}}_i) + \tilde{\varphi}_i \quad (5.12)$$

oder in Matrixschreibweise zu

$$\begin{aligned} \dot{\mathbf{z}} = & \begin{bmatrix} -c_1 & 1 & 0 & \dots & 0 \\ -1 & -c_2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & -c_{n-1} & 1 \\ 0 & \dots & 0 & -1 & -c_n \end{bmatrix} \mathbf{z} \\ & + \begin{bmatrix} \boldsymbol{\omega}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{\omega}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{\omega}_n \end{bmatrix}^T \begin{bmatrix} \mathbf{p} - \hat{\mathbf{p}}_1 \\ \vdots \\ \mathbf{p} - \hat{\mathbf{p}}_n \end{bmatrix} + \begin{bmatrix} \tilde{\varphi}_1 \\ \vdots \\ \tilde{\varphi}_n \end{bmatrix}, \end{aligned} \quad (5.13)$$

mit

$$\begin{bmatrix} \dot{\hat{\mathbf{p}}}_1 \\ \vdots \\ \dot{\hat{\mathbf{p}}}_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Gamma} & 0 & \dots & 0 \\ 0 & \boldsymbol{\Gamma} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{\Gamma} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{\omega}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \boldsymbol{\omega}_n \end{bmatrix} \mathbf{z}. \quad (5.14)$$

Zum Beweis der Stabilität des Fehlersystems (Gl. (5.13)) und der Adaptionssystems (Gl. (5.14)) wird die Ljapunow-Funktion

$$V = \frac{1}{2} \sum_{i=1}^n \left( z_i^2 + (\mathbf{p} - \hat{\mathbf{p}}_i)^T \boldsymbol{\Gamma}^{-1} (\mathbf{p} - \hat{\mathbf{p}}_i) \right) \quad (5.15)$$

angesetzt. Die Ableitung von Gl. (5.15) entlang der Lösungen der Gln. (5.13) und (5.14) führt zu

$$\begin{aligned} \dot{V} &= \sum_{i=1}^n z_i \dot{z}_i - \dot{\hat{\mathbf{p}}}_i^T \boldsymbol{\Gamma}^{-1} (\mathbf{p} - \hat{\mathbf{p}}_i) \\ &= \sum_{i=1}^n -c_i z_i^2 - z_i z_{i-1} + z_i z_{i+1} + \omega_i^T (\mathbf{p} - \hat{\mathbf{p}}_i) z_i + \tilde{\varphi}_i z_i - \omega_i^T (\mathbf{p} - \hat{\mathbf{p}}_i) z_i \\ &= \sum_{i=1}^n -c_i z_i^2 + \tilde{\varphi}_i z_i \\ &= \sum_{i=1}^n -\left( \sqrt{c_i} z_i - \frac{1}{2\sqrt{c_i}} \tilde{\varphi}_i \right)^2 + \frac{1}{4c_i} \tilde{\varphi}_i^2 \\ &\leq \sum_{i=1}^n -\left( \sqrt{c_i} z_i - \frac{1}{2\sqrt{c_i}} \tilde{\varphi}_i \right)^2 + \frac{1}{4c_i} \tilde{\varphi}_{\max,i}^2, \end{aligned} \quad (5.16)$$

mit  $\tilde{\varphi}_{\max,i}$  als maximalen Approximationsfehler  $\max(|\tilde{\varphi}_i|)$ , der mit Hilfe der Netzkonfiguration beliebig reduziert werden kann. Aus den Beziehungen nach Gl. (5.16) ist die einfache Stabilität nach Ljapunow ersichtlich. Für jedes  $c_i > 0$  sind alle Zustände des Fehlersystems  $\mathbf{z}$  und alle Reglerparameter  $\hat{\mathbf{p}}_i, i = 1, \dots, n$  beschränkt, da  $\dot{V} < 0$  gilt, wenn

$$\sum_{i=1}^n |z_i| > \sum_{i=1}^n \frac{\tilde{\varphi}_{\max,i}}{c_i}. \quad (5.17)$$

Aus  $z_1 = x_1$  folgt die Beschränktheit von  $x_1$  und die Beschränktheit von  $x_i, i = 2, \dots, n$  folgt aus der Beschränktheit von  $\alpha_i$  und  $x_i = z_i + \alpha_{i-1}$ . Für den Fall  $\beta(\mathbf{x}) \neq 0, \forall \mathbf{x} \in \mathbb{R}^n$  ist das Stellsignal  $u$  ebenfalls beschränkt. Die Grenzen der Signale und Parameter können beliebig klein gemacht werden, da sowohl die Reglerparameter  $c_i, i = 1, \dots, n$  als auch die Approximationsfehler  $\tilde{\varphi}_i, i = 1, \dots, n$  freie Entwurfsparameter sind.

## 5.3 Simulationsbeispiel

Die Vorteile des um ANN erweiterten Reglers gegenüber des einfachen *Backstepping*-Reglers werden anhand eines Beispiels demonstriert. Als Regelstrecke soll das nichtlineare System

$$\dot{x} = p \sin(x) + u \quad (5.18)$$

dienen. Der konstante Parameter  $p$  ist unbekannt und als Ziel der Regelung soll die Regelstrecke um den Ursprung stabilisiert werden. Dieses Beispielsystem ist aus (Krstić et al., 1995) entnommen. Um zu untersuchen, wie die beiden Regelverfahren auf Veränderungen des dynamischen Verhaltens der Regelstrecke reagieren, wird nach einer Simulationszeit von 20 s auf eine Regelstrecke mit der Darstellung

$$\dot{x} = p(\sin(x) + 2x^2 + 2) + u \quad (5.19)$$

umgeschaltet. Da das System (Gl. (5.18)) schon in PSF-Form ist, kann der Reglerentwurf mit dem *Backstepping*-Verfahren direkt begonnen werden, indem für den ersten Schritt die Ljapunow-Funktion

$$V = \frac{1}{2}x^2 \quad (5.20)$$

angesetzt wird. Die Ableitung von Gl. (5.20) nach der Zeit ergibt

$$\dot{V} = x\dot{x} = x(p \sin(x) + u) \quad (5.21)$$

und mit

$$u = -c_1 x - p \sin(x), \quad (5.22)$$

folgt

$$\dot{V} = -c_1 x^2. \quad (5.23)$$

Gl. (5.23) ist für  $c_1 > 0$  negativ definit und somit ist das geschlossene System asymptotisch stabil. Da der Parameter  $p$  unbekannt ist, muss im zweiten

Schritt die Ljapunow-Funktion aus Gl. (5.20) erweitert werden

$$V = \frac{1}{2}x^2 + \frac{1}{2\gamma}\tilde{p}^2. \quad (5.24)$$

Der Reglerparameter  $\gamma > 0$  bestimmt die Adaptionsgeschwindigkeit und  $\tilde{p}$  ist die Differenz zwischen dem realen Parameter  $p$  und dem Reglerparameter  $\hat{p}$

$$\tilde{p} = p - \hat{p}. \quad (5.25)$$

Setzt man  $\hat{p}$  und das Stellgesetz aus Gl. (5.22) ein, berechnet sich die Ableitung von Gl. (5.24) zu

$$\begin{aligned} \dot{V} &= x\dot{x} + \frac{1}{\gamma}\tilde{p}\dot{\tilde{p}} \\ &= x(-c_1x + \tilde{p}\sin(x)) + \frac{1}{\gamma}\tilde{p}\dot{\tilde{p}} \\ &= -c_1x^2 + \tilde{p}(x\sin(x) + \frac{1}{\gamma}\dot{\tilde{p}}). \end{aligned} \quad (5.26)$$

Mit der Annahme, dass der unbekannte Parameter konstant ist oder sich nur langsam ändert, ist die Approximation

$$\dot{\tilde{p}} = -\dot{\hat{p}} \quad (5.27)$$

gerechtfertigt. Das Einsetzen von Gl. (5.27) in Gl. (5.26) führt zu

$$\dot{V} = -c_1x^2 + \tilde{p}(x\sin(x) - \frac{1}{\gamma}\dot{\hat{p}}), \quad (5.28)$$

und mit

$$\dot{\hat{p}} = \gamma x \sin(x) \quad (5.29)$$

ist die Ableitung der Ljapunow-Funktion negativ semidefinit und somit ist das geschlossene System stabil.

Zusammengefasst erhält man für die Regelstrecke beschrieben durch Gl. (5.18) mit Hilfe des *Backstepping*-Verfahrens das Stellgesetz

$$u = -c_1x - \hat{p}\sin(x) \quad (5.30)$$

und das Adaptionsgesetz

$$\dot{\hat{p}} = \gamma x \sin(x). \quad (5.31)$$

Da die gegebene Regelstrecke die Ordnung eins hat, wird für die RBF NPSF-Struktur der Ansatz

$$\dot{x} = u + \Phi^T(|x - \mathbf{c}_i|)\mathbf{p} \quad (5.32)$$

gemacht. Unabhängig von der genauen Struktur der Regelstrecke ist dieser Ansatz für alle NPSF-Systeme erster Ordnung gültig. Hier ist auch der wesentliche Vorteil der beschriebenen Erweiterung zu sehen. Mit dieser Regelstreckenbeschreibung wird nun der zuvor beschriebene *Backstepping*-Entwurf durchgeführt. Zunächst wird mit der Annahme, dass die optimalen Gewichte des ANN bekannt sind, die Ljapunow-Funktion aus Gl. (5.20) angesetzt und die Ableitung zu

$$\dot{V} = x\dot{x} = x(\Phi^T(|x - \mathbf{c}_i|)\mathbf{p} + u) \quad (5.33)$$

bestimmt. Mit dem Stellgesetz

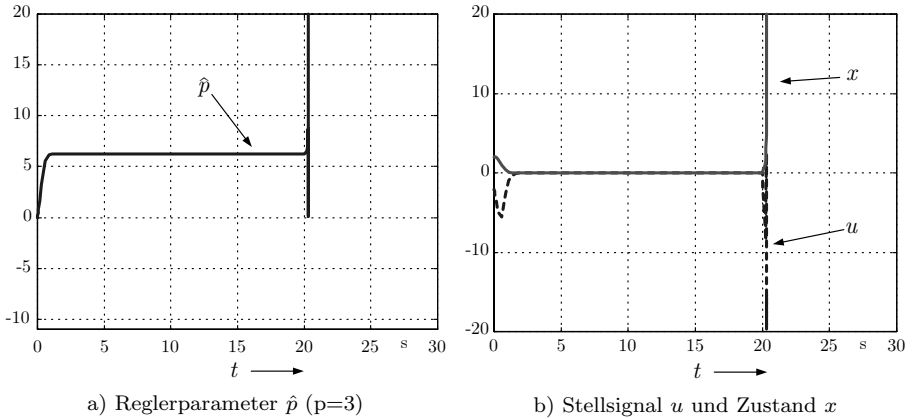
$$u = -c_1 x - \Phi^T(|x - \mathbf{c}_i|)\mathbf{p} \quad (5.34)$$

ist sie negativ definit. Wie zuvor besteht der zweite Schritt in der Einführung eines Parameters  $\hat{\mathbf{p}}$  und der damit verbundenen erweiterten Ljapunow-Funktion (G. (5.24)). Nach kurzer Rechnung ergibt sich das Adaptionsgesetz für die Netzgewichte zu

$$\dot{\hat{\mathbf{p}}} = \gamma \Phi(|x - \mathbf{c}_i|)x. \quad (5.35)$$

Sowohl das Stell- und Adaptionsgesetz des Standard-*Backstepping*-Reglers (Gln. (5.30) und (5.31)) als auch des erweiterten Reglers (Gln. (5.34) und (5.35)) hätten mit Hilfe der Entwurfsgleichungen (Gln. (2.6) bis (2.9)) berechnet werden können. Zur Verdeutlichung der Entwurfsmethode sind die Berechnungen für dieses Beispiel schrittweise aufgeführt worden.

Die Simulationsergebnisse der Regelung mit dem *Backstepping*-Verfahren sind im Bild 5.2 dargestellt. Hierbei wurde für die Reglerkonstante  $c_1 = 1$



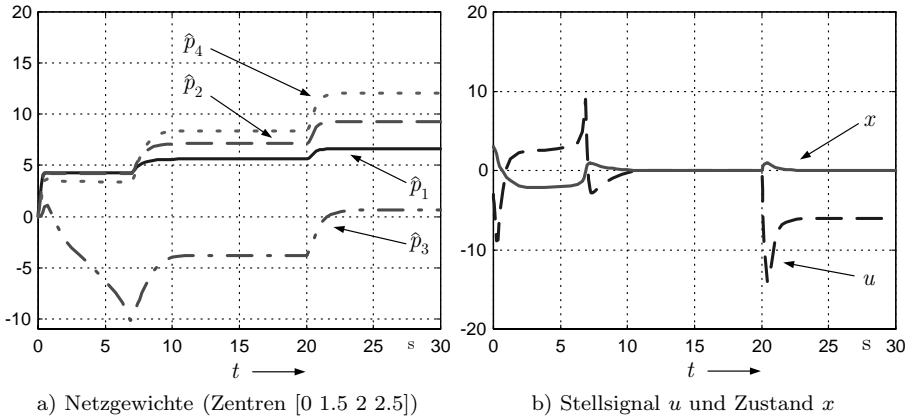
**Bild 5.2:** Regelung mit dem *Backstepping*-Verfahren

und für die Adaptionsgeschwindigkeit  $\gamma = 6$  gewählt. Im rechten Teilbild ist die Stellgröße  $u$  und die Zustandsgröße  $x$  über der Zeit aufgetragen. Es ist zu beobachten, dass die Regelung für das nominelle System schnell ist. Schon nach ca. zwei Sekunden ist das System, ausgehend von  $x(0) = 2$ , im Ursprung und die Parameteradaption (linkes Teilbild) abgeschlossen. Anders ist das zum Zeitpunkt  $t = 20$  s, wo auf die Regelstrecke gemäß Gl. (5.19) umgeschaltet wird. Sowohl die Zustandsgröße als auch der Parameter vergrößern sich und das System wird instabil.

Im Bild 5.3 sind die entsprechenden Signalverläufe der Regelung mit dem um ein RBF-Netz erweiterten *Backstepping*-Verfahren aufgezeichnet. Es wurden hierbei insgesamt vier RBF-Neuronen mit der Gauß-Funktion

$$\Phi(r) = \frac{1}{\sqrt{2\pi}\mathfrak{w}} \exp\left(-\frac{r^2}{2\mathfrak{w}^2}\right) \quad (5.36)$$

als Aktivierungsfunktion gewählt, deren Zentren bei 0, 1.5, 2, 2.5 lagen. Die weiteren Reglerparameter wurden aus der ersten Simulation übernommen. Im rechten Teilbild ist zu erkennen, dass diese Regelung eine wesentlich längere Adaptionsphase hat. Erst nach ca. zehn Sekunden ist das System im Ursprung stabilisiert. Dieses Verhalten ist aber zu erwarten, da die Regelung mit ANN



**Bild 5.3:** Regelung mit dem erweiterten *Backstepping*-Verfahren

zwar allgemeiner, aber für eine spezielle Regelstrecke langsamer ist. Interessant ist aber hier das Verhalten zum Umschaltunkt. Anders als in der ersten Simulation kann der Regler das System trotz dieser großen Veränderung stabilisieren. Im linken Teilbild sind die Verläufe der Netzparameter dargestellt.

## 5.4 Zusammenfassung und Bewertung

Im vorliegenden Kapitel wurde die Erweiterung des *Backstepping*-Verfahrens beschrieben. Mit dieser Erweiterung kann das Entwurfsverfahren auf Regelstrecken in NPSF-Struktur angewendet werden. Ein Stabilitätsbeweis wurde vorgenommen und an einem einfachen Beispiel wurde der Entwurf des *Backstepping*-Verfahrens und dessen Erweiterung mit ANN gezeigt. Bei den Simulationen zeigt sich, dass das einfache *Backstepping*-Verfahren für den Fall, dass die Nichtlinearität bekannt ist, wesentlich schneller reagiert, als das erweiterte Verfahren. Sobald sich aber das dynamische Verhalten des Systems ändert oder das Modell zu ungenau ist, zeigt sich die große Gefahr dieses Entwurfs. Das Gesamtsystem kann instabil werden. Anders ist dies beim erweiterten Verfahren. Durch die Verwendung von ANN ist es im Prinzip nicht mehr notwendig, die



nichtlinearen Funktionen des Systems genau zu kennen. Es ist immer möglich, mit einer ausreichend großen Anzahl von RB-Funktionen, ein stabiles Verhalten des Gesamtsystems zu erzwingen. Probleme, die beim Entwurf mit dem erweiterten Verfahren auftreten, sind zum einen die Bestimmung der Basenanzahl und zum anderen die Festlegung der Zentren. Ist hier einmal eine Auswahl getroffen, ist die Regelung wesentlich flexibler als die Regelung mit dem einfachen Entwurfsverfahren. Ein weiterer Vorteil ist der Reglerentwurf selbst. Bei dem erweiterten Entwurfsverfahren kann mit den Parametern

- RB-Funktion
- Basenanzahl
- Zentren
- Startparameter

ein Stellgesetz berechnet werden, ohne spezielle Informationen über die Nichtlinearitäten zu benötigen, wie es beim einfachen Verfahren der Fall ist. Abschließend sei noch bemerkt, dass die praktische Realisierung beider Verfahren kritisch ist. Dies betrifft beispielsweise den enormen Rechenaufwand und die große Anzahl von freien Reglerparametern, für die keine Einstellregeln existieren. Des Weiteren wird beim Stabilitätsbeweis unbeschränkte Stellenergie vorausgesetzt, was bei realen Regelstrecken nie der Fall ist.

# 6 Erprobung der Reglerstrukturen

In den Kapiteln 3 und 4 wurden neue, nichtlineare adaptive Regler für Hammerstein- und Wiener-Systeme vorgestellt. Die Leistungsfähigkeit der Verfahren zur Regelung eines Hammerstein-Systems werden nachfolgend an den Simulationsmodellen eines hydraulischen Systems und eines flexiblen Manipulators sowie an einer Laboranlage bestehend aus einem hydraulischen Vorschubantrieb gezeigt.

Die Eigenschaften der WANNNA-Regelung werden durch die Behandlung von Simulationsbeispielen mit Regelstrecken, die typische Ausgangsnichtlinearitäten enthalten sowie anhand des Modells eines Neutralisationsreaktors untersucht. Das erstellte Modell des Neutralisationsreaktors basiert auf Messungen an einer Laboranlage.

Die Verfahren wurden auf einem Standard-PC implementiert. Die hierzu verwendeten Programmpakete *Matlab/Simulink* der Firma *The MathWorks, Inc.* ermöglichten eine benutzerfreundliche Umsetzung. Diese Programmpakete realisieren, basierend auf der Matrizenrechnung, eine wissenschaftlich-technisch orientierte Entwicklungsumgebung. Die Echtzeitanbindung wurde mit der *InteractiveRealtime*-Toolbox (Vetch Consulting, 1997) durchgeführt. Mit dieser Erweiterung ist die Anbindung von *Simulink*-Strukturen an eine technische Anlage, z. B. eine Laboranlage, ohne wesentlichen Aufwand möglich. Der Regelstreckenblock wird durch einen speziellen Messkartenblock ersetzt, in dem das Ausgeben und Einlesen von Anlagensignalen zu vorgegebenen Abtastzeitpunkten realisiert ist. Es ist zu beachten, dass das Programm unter dem Betriebssystem *Windows NT* läuft und somit das Echtzeitverhalten nicht

gesichert ist. Bei Belastung des Simulationsrechners durch weitere Prozesse sind Zeitüberläufe möglich, die jedoch dem Benutzer gemeldet werden. Die *InteractiveRealtime*-Toolbox ist deshalb nur für die Entwicklung in einer Testumgebung geeignet. Sie bietet aber durch die direkte Anbindung von *Matlab/Simulink*-Strukturen an eine Laboranlage eine zeitsparende und preiswerte Alternative zu professionellen Echtzeitsystemen.

## 6.1 Positionsregelung eines hydraulischen Vorschubzylinders

Die zu regelnde Anlage ist im Bild 6.1 schematisch dargestellt. Es handelt sich um ein hydraulisches System, das aus einem Proportionalventil (1), einem Vorschubzylinder (2) und einer variablen Lastkraft besteht. Diese Kraft ist aus einem Masse-Feder-Dämpfer-System zusammengesetzt, bei der die Masse (3), die Dämpfung (4) und die Vorspannung der Rückstellfeder (5) variiert werden können. Durch die flexible Konfiguration können verschiedene, für die Praxis relevante, Situationen untersucht werden. Beispielsweise die Positionierung bei variabler Lastmasse oder bei Störung durch verschiedene Dämpfungen und Federkräfte. Als Proportionalventil wurde bewußt eine einfache Version gewählt, die sich im wesentlichen durch eine ausgeprägte Totzone beschreiben läßt. Die Stellgröße  $v$  ist der Steuerstrom  $I$ , mit dem das Ventil angesteuert wird. Die durch den Steuerstrom erzeugte Kraft wirkt auf den Steuerschieber im Ventil und sorgt somit für ein Druckgefälle im Differenzialzylinder. In Abhängigkeit von der Druckdifferenz variiert die Position des Schlittens und der Lastmasse. Die Position der Lastmasse ist die Regelgröße  $y$ . Stell- und Regelgröße sind für die Verarbeitung im Rechner auf den Bereich  $\pm 10$  V normiert. Ein vollständiges Modell dieser Regelstrecke ist stark nichtlinear (Merrett, 1967) und deshalb nicht für einen Reglerentwurf geeignet. Durch Linearisierung (Knohl und Unbehauen, 2000a) erhält man schließlich ein handhabbares Modell mit der Übertragungsfunktion

$$\frac{Y(s)}{V(s)} = \frac{b_0}{s(a_2 s^2 + a_1 s + a_0)}, \quad (6.1)$$

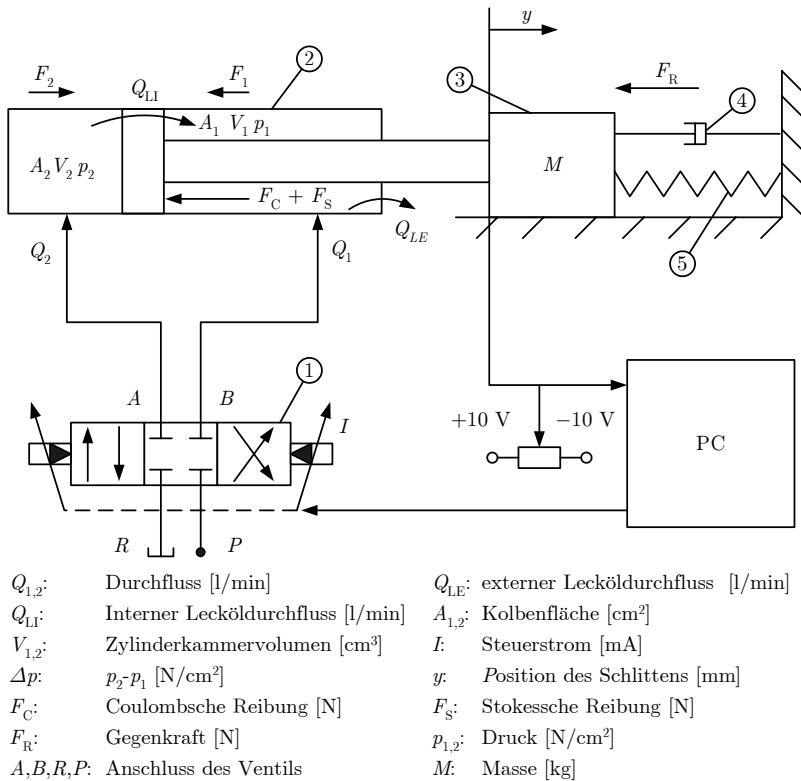
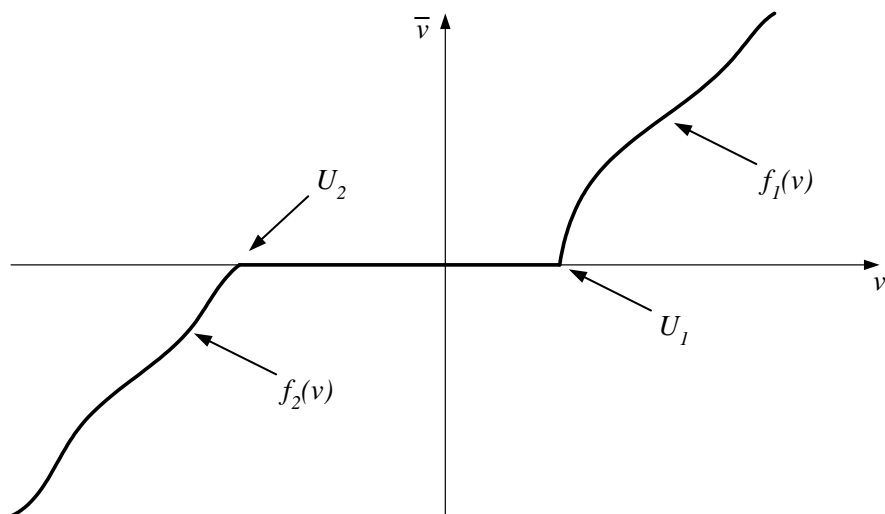


Bild 6.1: Versuchsanlage: Hydraulischer Vorschubantrieb

wobei das  $PT_2$ -Verhalten die Dynamik des hydraulischen Kreises beschreibt und der Vorschubzylinder als Integrator betrachtet wird. Nach Diskretisierung mit einem Halteglied nullter Ordnung und Parameteridentifikation an der Anlage (Abtastzeit  $T=40$  ms), wurde die diskrete Übertragungsfunktion

$$\frac{Y(z^{-1})}{V(z^{-1})} = \frac{0.1449z^{-1} + 0.0435z^{-2} - 0.0001z^{-3}}{1 - 0.8898z^{-1} - 0.1472z^{-2} + 0.0414z^{-3}} \quad (6.2)$$

für die Simulationen benutzt. Zur Berücksichtigung der Nichtlinearität des Proportionalventils wird diesem System eine Totzone vorgeschaltet. Anders als bei den Standardansätzen wird jedoch für die reale Anlage angenommen, dass die Totzone im Allgemeinen unsymmetrisch und in den Teilstücken nichtlinear ist (Bild 6.2).



**Bild 6.2:** Nichtlineare und unsymmetrische Totzone

Mathematisch ist sie beschrieben durch:

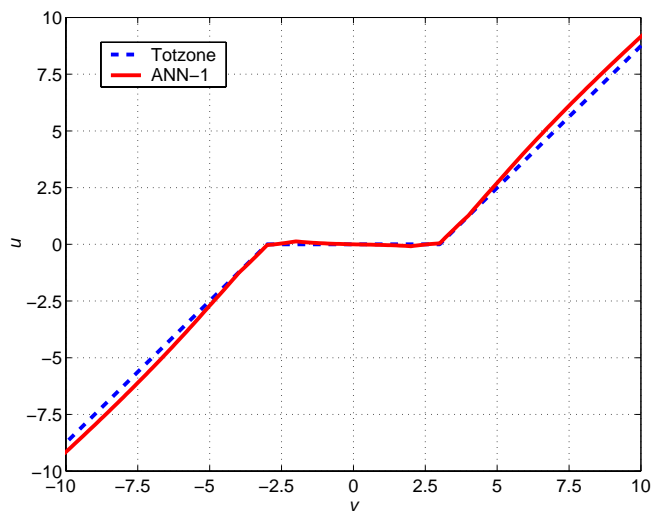
$$\bar{v} = \begin{cases} f_1(v) & v > U_1 > 0 \\ 0 & U_1 \geq v \geq U_2 \\ f_2(v) & 0 > U_2 > v. \end{cases} \quad (6.3)$$

Zusammengefaßt läßt sich die Anlage vereinfacht als Serienschaltung einer Totzone, eines Integrators und eines PT<sub>2</sub>-Systems beschreiben, was ein Hammerstein-System darstellt. Durch Messungen an der Laboranlage wurden die Grenzen der Totzone ungefähr zu  $\pm 3$  V bestimmt. Der HANNNA-Regler und das Modell der Anlage wurden in *Matlab/Simulink* implementiert. Die Versuche wurden auf einem Pentium-II/200 MHz Rechner mit einer Abtastzeit von 40 ms durchgeführt.

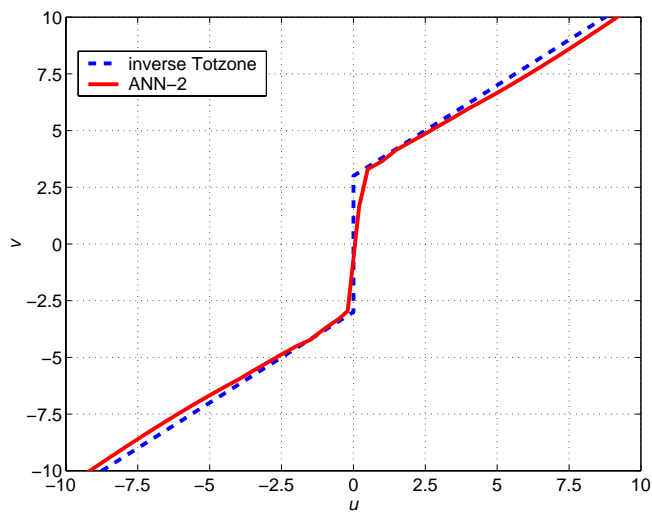
Die Parameter der Netze wurden — wie im Abschnitt 3.1.2 beschrieben — während der ersten Simulationsversuche festgelegt. Gutes Regelverhalten ergab sich bei der Verwendung von elf Neuronen für ANN-1 und vierzehn Neuronen für ANN-2 in der versteckten Schicht. Als Führungssignal diente ein pulsartiges Zufallssignal mit einer Pulsbreite von 5 s. Die Parameter der Netze sind in Tabelle 6.1 aufgeführt und die Startwerte der Gewichte wurden zu null gesetzt. Im Bild 6.3 sind die statischen Kennlinien des trainierten Netzes ANN-1 und einer Totzone und im Bild 6.4 die entsprechenden Verläufe der Inversen dargestellt. Beide Netze haben die Kennlinienverläufe nahezu exakt gelernt. Bild 6.5 zeigt den Soll- und Istwertverlauf des simulierten Systems. Nach einer kurzen Adaptionsphase von ca. 20 s folgt das System dem Sollwert rasch und mit nur geringen Überschwingungen. Um die Flexibilität des HANNNA-Reglers zu zeigen, wurde die Totzone gegen die Funktion  $f(v) = 0.03v^3$  ausgetauscht.

**Tabelle 6.1:** Netzparameter des HANNNA-Reglers

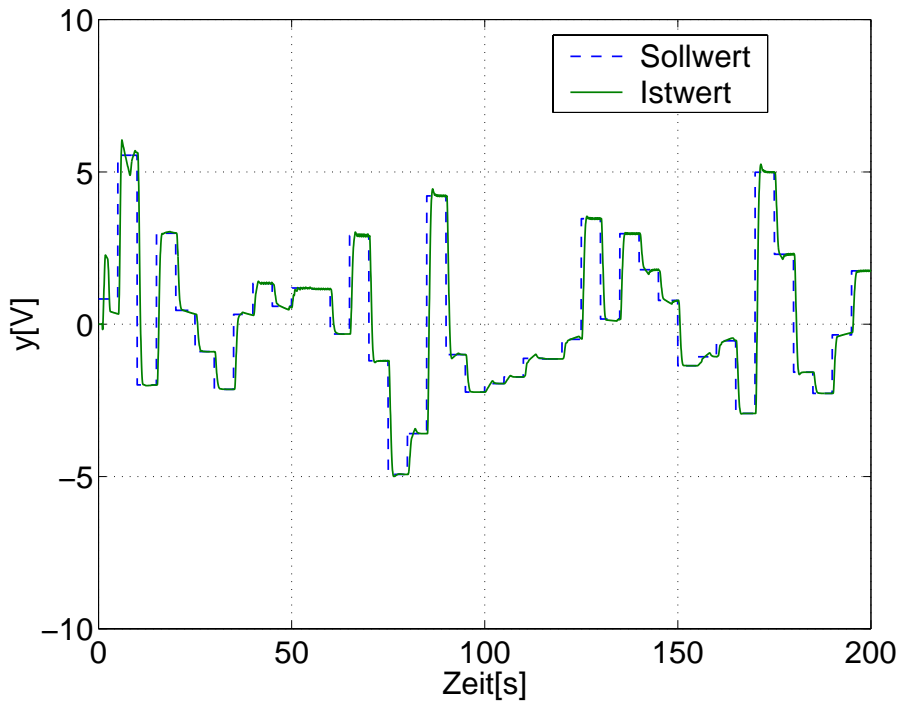
Netz	$\gamma$	Zentren der Basisfunktionen
ANN-1	5	-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5
ANN-2	6	-4, -3, -2, -1.5, -1, -0.5, -0.2, 0.2, 0.5, 1, 1.5, 2, 3, 4



**Bild 6.3:** Statische Kennlinien des Netzes ANN-1 und der Totzone



**Bild 6.4:** Statische Kennlinien des Netzes ANN-2 und der inversen Totzone

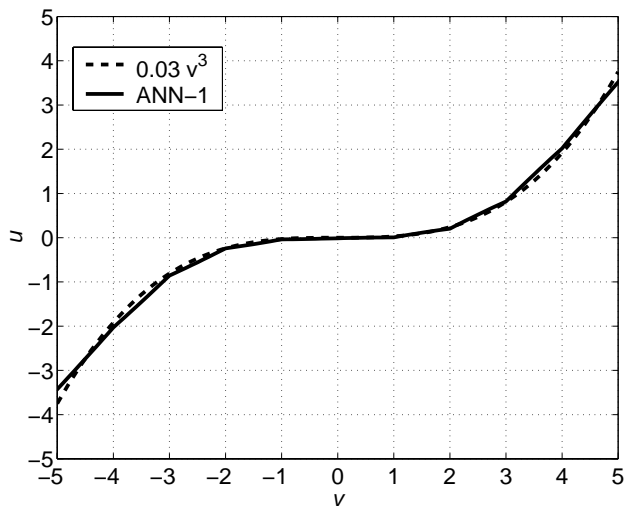


**Bild 6.5:** Soll- und Istwertverlauf bei HANNNA-Regelung mit der Eingangs-nichtlinearität Totzone

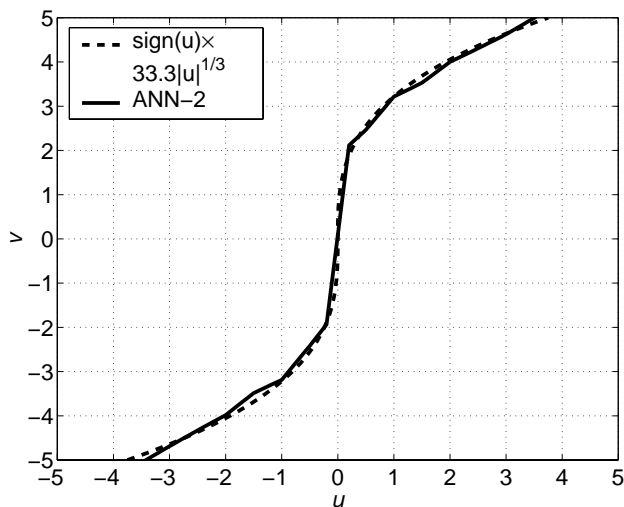


Alle anderen Einstellungen wurden hierbei nicht verändert. Die entsprechenden Simulationsergebnisse sind in den Bildern 6.6, 6.7 und 6.8 dargestellt. Auch bei dieser Konfiguration lernen die Netze sowohl den Verlauf der Eingangsnichtlinearität als auch deren Inverse sehr genau. Am Signalverlauf ist zu sehen, dass nach ca. 80 s der Adaptionsvorgang abgeschlossen ist und dass das Systemausgangssignal dem Sollverlauf folgt. Das stark schwingende Verhalten des Ausgangssignals während der Adaptionsphase deutet auf ein noch unvollständiges Training der Netze ANN-1 und ANN-2 hin. Trotz unveränderter Reglerkonfiguration erzielt man mit dem HANNNA-Regler für beide Systeme nach entsprechender Adaptionszeit ein sehr gutes Verhalten des geschlossenen Regelkreises. Ausgehend von den in der Simulation ermittelten Reglerparametern wurden die Versuche an der Laboranlage „Hydraulischer Vorschubantrieb“ durchgeführt. Da die Rechengeschwindigkeit des verwendeten Rechners zu gering ist, wurde für das Training des zweiten Netzes (ANN-2) anstatt des LSE-Verfahrens das RLS-Verfahren zur Schätzung der Gewichte eingesetzt. Alle weiteren Einstellungen wurden von der Simulation übernommen. Die Ergebnisse der Versuche sind in den Bildern 6.9, 6.10 und 6.11 dokumentiert. Da der genaue Verlauf der statischen Eingangsnichtlinearität unbekannt ist, ist keine Aussage über den Lernerfolg möglich. Es wird aber erwartungsgemäß der angenäherte Verlauf einer Totzone gelernt. Der Unterschied zur idealen Totzone kann verschiedene Gründe haben. Zum einen ist es möglich, dass das ANN-1 nicht richtig konfiguriert war. Wahrscheinlicher ist aber, dass sich die Eingangsnichtlinearität aus verschiedenen Effekten zusammensetzt und tatsächlich den erlernten Verlauf aufweist. Das Ausgangssignal folgt nach ca. 15 s dem Sollwertverlauf wiederum rasch, ohne Überschwingerungen und auch über einen längeren Zeitraum mit konstanter Genauigkeit.

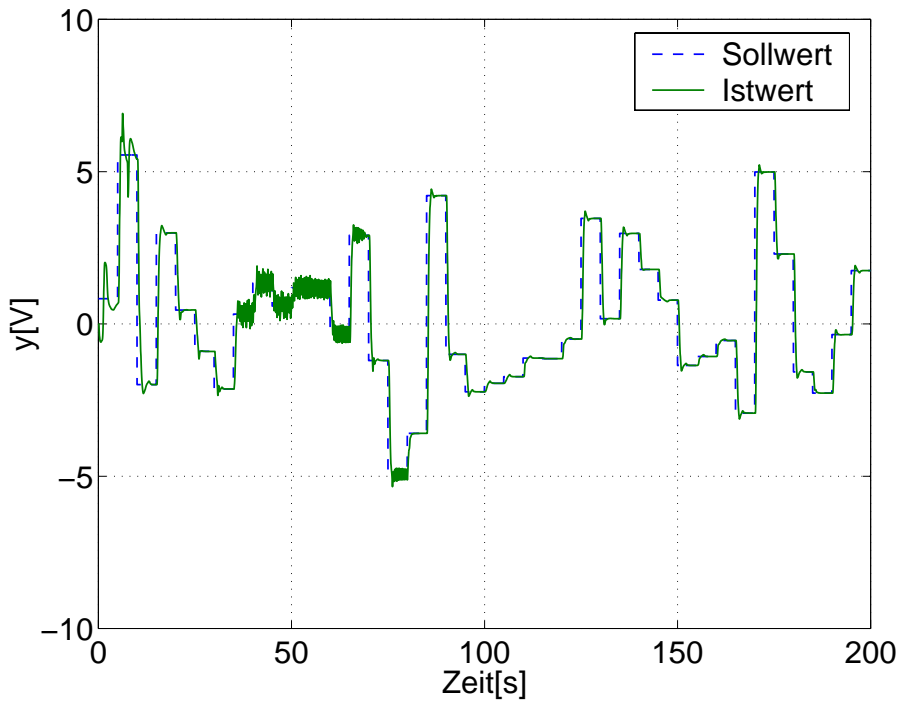
Zum Vergleich wurde ein Versuch an der Anlage durchgeführt, bei dem beide Netze aus dem Regler entfernt wurden, so dass nur noch die Identifikation über das lineare Modell aktiv ist und keine Kompensation der Eingangsnichtlinearität erfolgt. Es ergibt sich ein adaptiver LQ-Regler, bei dem der Gewichtungsfaktor  $\rho$  wie zuvor beim HANNNA-Regler auf 80 eingestellt wurde. Der Verlauf des Ausgangssignals ist im Bild 6.12 dargestellt. Ohne Kompensati-



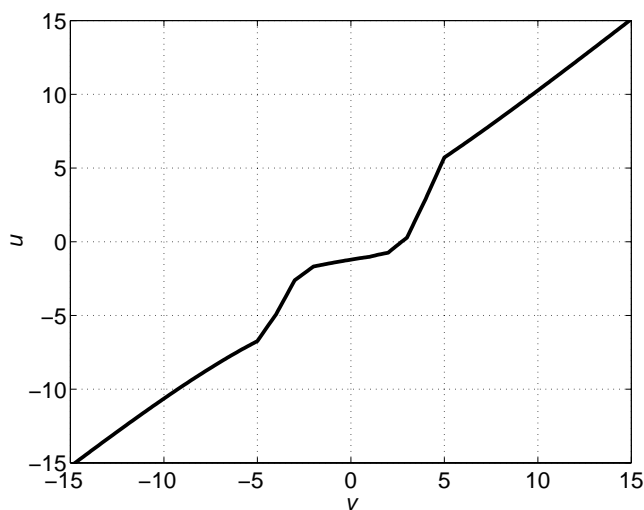
**Bild 6.6:** Statische Kennlinien des Netzes ANN-1 und der Funktion  $f(v) = 0.03v^3$



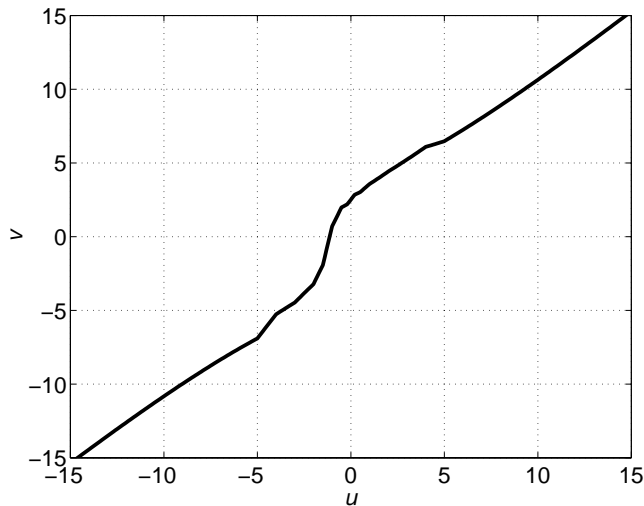
**Bild 6.7:** Statische Kennlinien des Netzes ANN-2 und der Funktion  $f(u) = 33.3 \operatorname{sgn}(u) \sqrt[3]{|u|}$



**Bild 6.8:** Soll- und Istwertverlauf bei HANNNA-Regelung mit der Eingangs-nichtlinearität  $f(v) = 0.03v^3$

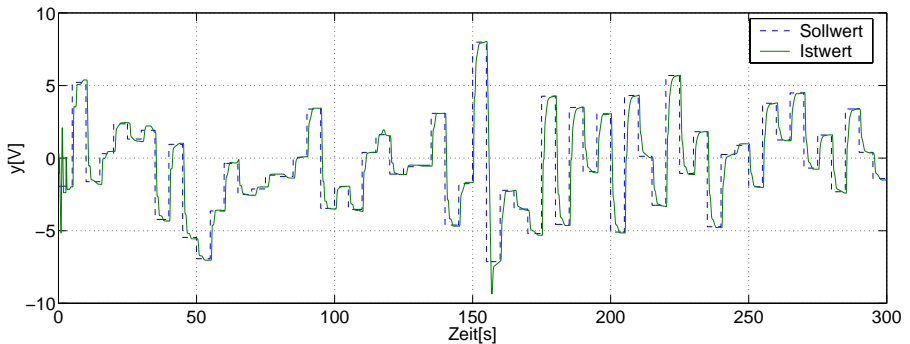


**Bild 6.9:** Statische Kennlinien des Netzes ANN-1 bei Versuchen am hydraulischen Vorschubantrieb

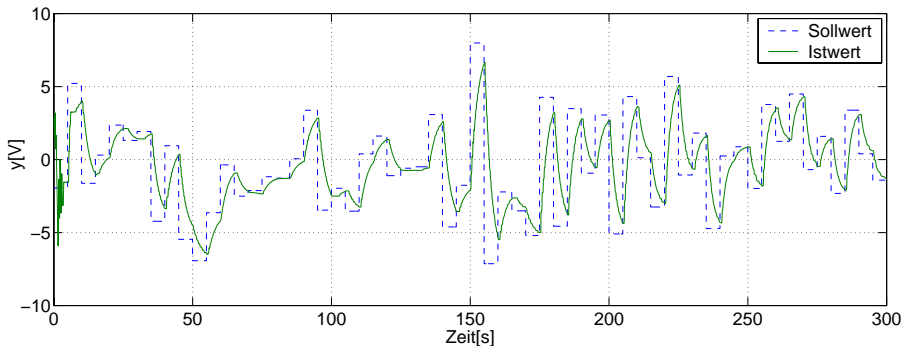


**Bild 6.10:** Statische Kennlinien des Netzes ANN-2 bei Versuchen am hydraulischen Vorschubantrieb

on der Eingangsnichtlinearität erhält man inakzeptables Regelverhalten. Mit diesem Regler gelingt es nicht, das Ausgangssignal des Systems dem Führungssignal innerhalb von 5 s nachzuführen.



**Bild 6.11:** Soll- und Istwertverlauf bei HANNNA-Regelung am hydraulischen Vorschubantrieb

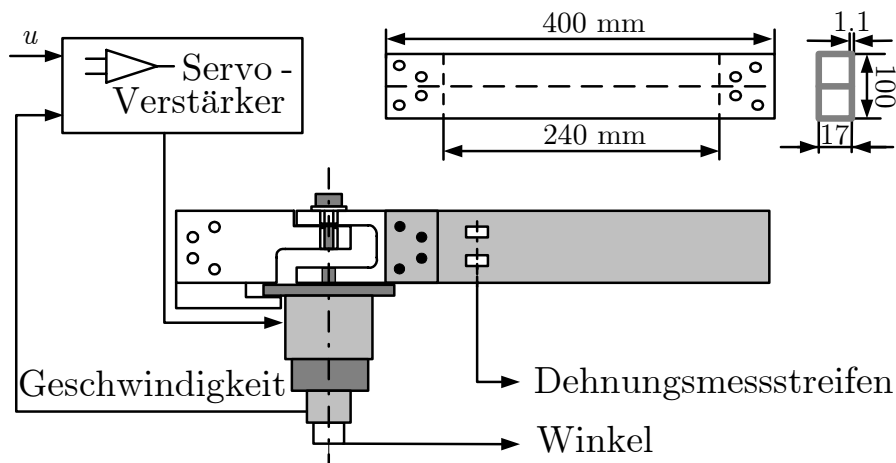


**Bild 6.12:** Soll- und Istwertverlauf bei LQ-Regelung am hydraulischen Vorschubantrieb

Zusammengefasst erhält man mit dem HANNNA-Regler ein deutlich besseres Verhalten des geschlossenen Regelsystems als bei der Verwendung eines Standardreglers. Ohne die Netzstruktur ändern zu müssen, konnten verschiedene Eingangsnichtlinearitäten erlernt und erfolgreich kompensiert werden.

## 6.2 Regelung eines flexiblen Manipulators

Als zweites Beispiel für die HANNNA-Regelung dient ein eingliedriger, planarer Manipulatorarm, wie er im Bild 6.13 dargestellt ist. Der Arm bewegt



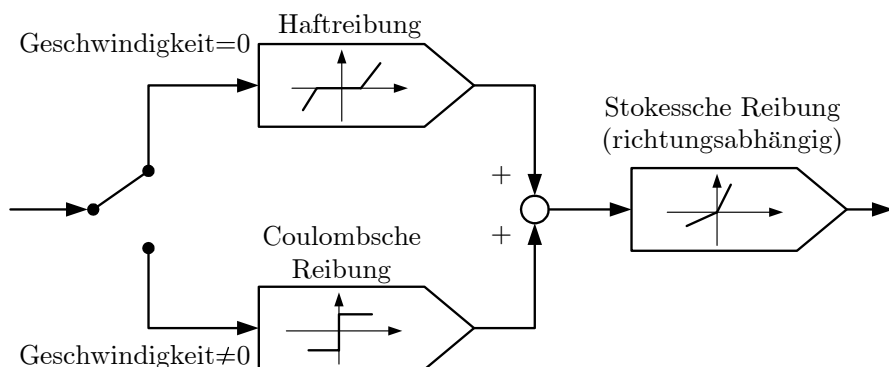
**Bild 6.13:** Flexibler planarer Manipulator

sich rotatorisch in der horizontalen Ebene und ist auf Grund seiner Bauweise elastisch. Die bei der Bewegung entstehenden Schwingungen können über die am Arm angebrachten Dehnungsmessstreifen gemessen werden. Dieser Arm ist ein Teil eines viergliedrigen Manipulators, der am Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität Bochum zur Untersuchung moderner Regelungskonzepte aufgebaut ist (Fabritz, 1997). Die Regelgröße bei dieser Regelstrecke ist der Drehwinkel des Arms in Grad. Als Stellglied wird ein Gleichstrommotor mit *Harmonic Drive* Getriebe eingesetzt. Diese Antriebe werden in Industrierobotern häufig verwendet. Ein Hauptmerkmal dieser Stelleinheit ist das geringe Getriebeispiel, welches aber mit einer verhältnismäßig großen Reibung erkauft wird. Zur Modellierung eines solchen Manipulators existieren zahlreiche Ansätze (Zeng, 1999). Eines dieser Modelle für die Berücksichtigung der Reibungseffekte des *Harmonic Drive* Getriebes ist

ein Hammerstein-Modell, bei dem die Dynamik des Manipulators mit einem linearen zeitdiskreten Modell zweiter Ordnung

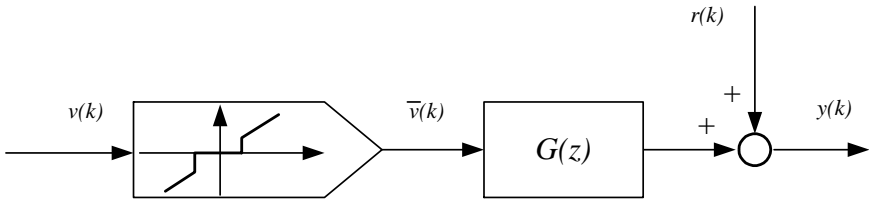
$$G(z^{-1}) = \frac{0.1098z^{-1} + 0.0831z^{-2}}{1 - 1.8334z^{-1} + 0.8334z^{-2}}, \quad (6.4)$$

und die Reibung mit einer statischen Ausgangskennlinie modelliert werden. Die Abtastzeit wurde zu  $T = 0.04$  s gewählt. Die Parameter des linearen Teilmodells wurden anhand einer Identifikation an der Laboranlage bestimmt. Eine schematische Darstellung der nichtlinearen Kennlinie ist im Bild 6.14 gezeigt.



**Bild 6.14:** Modell der Reibung des Manipulators

Sie besteht aus der Kombination der Haftreibung, der Coulombschen und einer richtungsabhängigen Stokesschen Reibung. Fasst man die drei Reibungseffekte zu *einer* statischen Nichtlinearität zusammen, ist mit dem linearen Teilmodell gemäß Gl. (6.4) somit das Hammerstein-Modell des eingliedrigen, flexiblen Manipulators entsprechend Bild 6.15 gegeben. Ist die durch das Eingangssignal erzeugte Kraft kleiner als die Haftreibung, bewegt sich der Arm nicht, was einer Totzone entspricht. Übertrifft die Antriebskraft schließlich die Haftreibung, ändert sich das Eingangssignal  $\bar{v}(k)$  des linearen Teilmodells, durch die Coulombsche Reibung bedingt, sprungartig und steigt dann bei wachsender Antriebskraft entsprechend der Stokesschen Reibung an. Für die Simulation wurden die Parameter der Eingangsnichtlinearität wie folgt eingestellt:

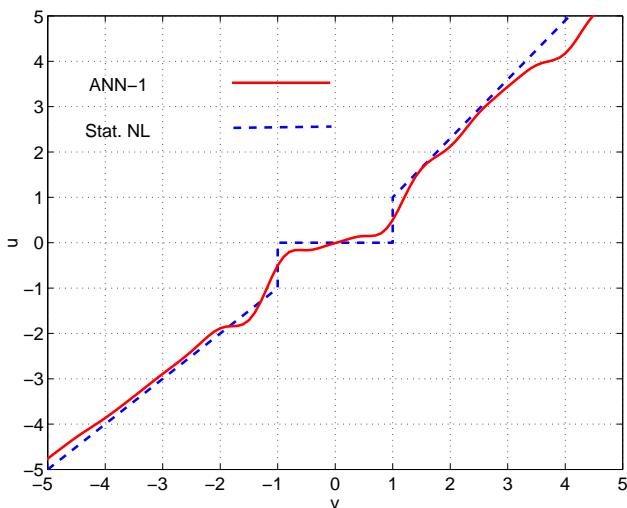


**Bild 6.15:** Hammerstein-Modell des eingliedrigen, flexiblen Manipulators

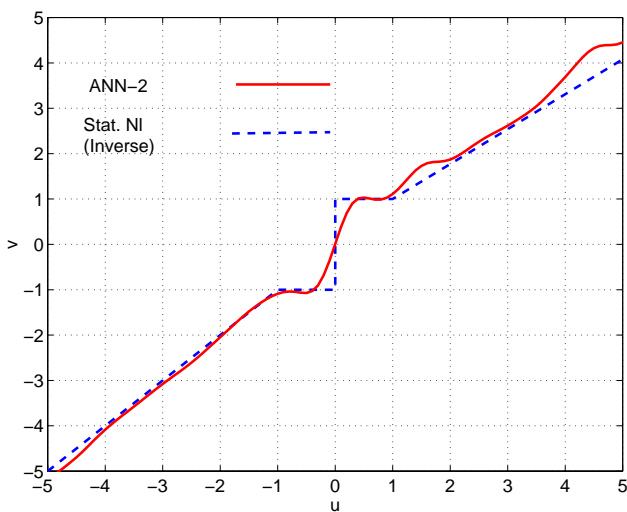
- linke/rechte Grenze der Totzone:  $-1/1$
- Coulombscher Reibungskoeffizient: 1
- Stokesscher Reibungskoeffizient (richtungsabhängig):  $1/1.3$

Aufgrund des anspruchsvollen Verlaufs der Nichtlinearität wurden 28 Neuronen für das ANN-1 und 24 für das ANN-2 zur Regelung verwendet. Das Referenzsignal war wieder ein pulsformiges Zufallssignal mit einer Pulsbreite von 5 s. Das Ein-Ausgangsverhalten der trainierten Netze ANN-1 und ANN-2 ist mit den zugehörigen Verläufen der optimalen Nichtlinearitäten in den Bildern 6.16 und 6.17 dargestellt. Es ist zu sehen, dass beide Netze die Ziel-funktion nach einer Trainingszeit von 200 s gut gelernt haben. Dies gilt für die Totzone, die Sprünge bei  $\pm 1$  und die unterschiedlichen Steigungen in gleichem Maße. Im Bild 6.18 sind der Sollwertverlauf und die Regelgröße mit HANNNA-Regler und mit LQ-Regler und im Bild 6.19 die zugehörigen Stellgrößen dargestellt. In den ersten 30 Sekunden erzielten beide Regler ungefähr das gleiche Regelverhalten. Nach dieser Adaptionsphase ist das Regelverhalten des HANNNA-Reglers deutlich besser als das des LQ-Reglers. Die Regelstrecke folgt dem Sollwert rasch und ohne Überspringen. An den im Bild 6.18 mit einem Pfeil markierten Stellen beträgt die maximale Überspringweite bei LQ-Regelung ca. 20% des Sollwertes. Mit der HANNNA-Regelung kann dieser Wert auf unter 6% gesenkt werden. Nach der Adaptionsphase bleibt die Qualität der Regelung über den gesamten Simulationszeitraum konstant gut. Durch die Kompensation der Eingangsnichtlinearität ist das Verhalten des geschlossenen Systems somit wesentlich verbessert. Am Verlauf der Stellgröße ist

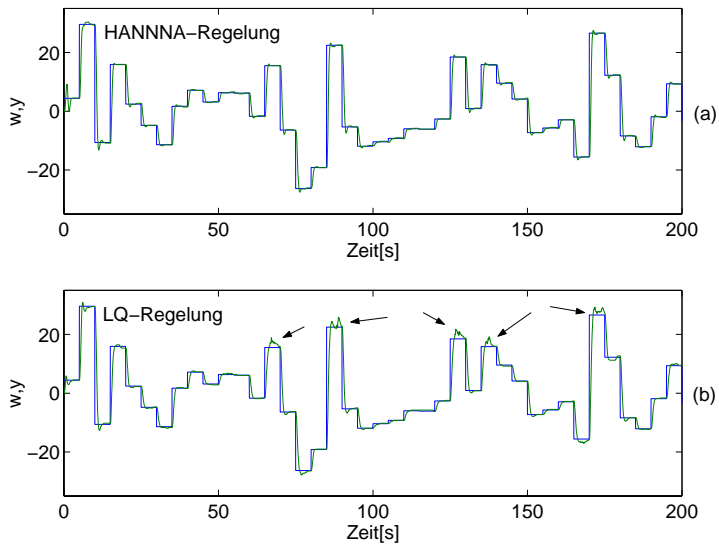




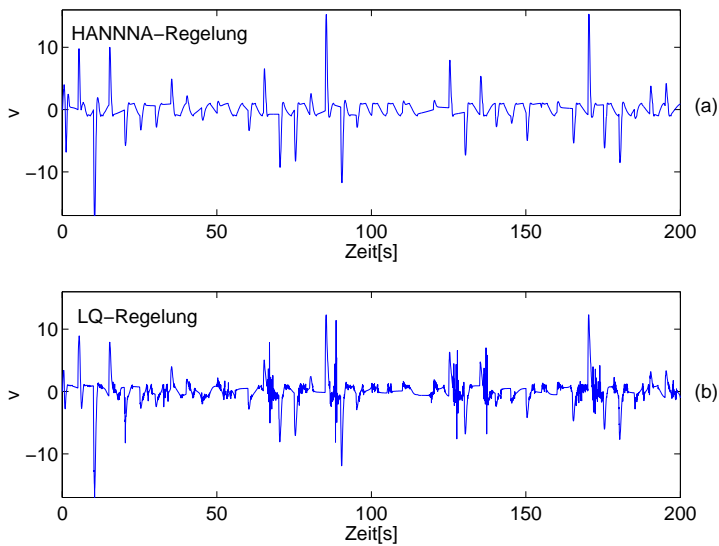
**Bild 6.16:** Kennlinien des trainierten Netzes ANN-1 und der statischen Nichtlinearität (Trainingszeit : 200 s)



**Bild 6.17:** Kennlinien des trainierten Netzes ANN-2 und der inversen statischen Nichtlinearität (Trainingszeit : 200 s)



**Bild 6.18:** Sollwertverlauf und Regelgröße mit (a) HANNNA- und (b) LQ-Regler

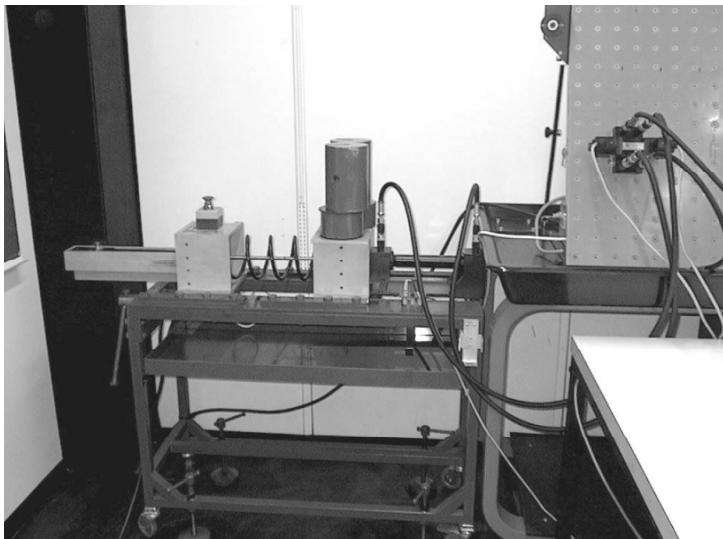


**Bild 6.19:** Stellgrößenverlauf bei (a) HANNNA- und (b) LQ-Regelung

ebenfalls ein deutlich ruhigeres Stellverhalten des HANNNA-Reglers zu erkennen. Der Bereich in dem das Stellsignal innerhalb des Bereichs, der Totzone entspricht, liegt, wird beim HANNNA-Regler, wie erwartet, übersprungen. Bei der LQ-Regelung treten hier starke Schwingungen auf.

## 6.3 Duale Regelung eines hydraulischen Vorschubzylinders

Bild 6.20 zeigt den Versuchsstand „Hydraulischer Vorschubantrieb“ wie er am Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität Bochum aufgebaut ist. Die einzelnen Komponenten und das mathematische Modell



**Bild 6.20:** Fotografie der Laboranlage „Hydraulischer Vorschubantrieb“

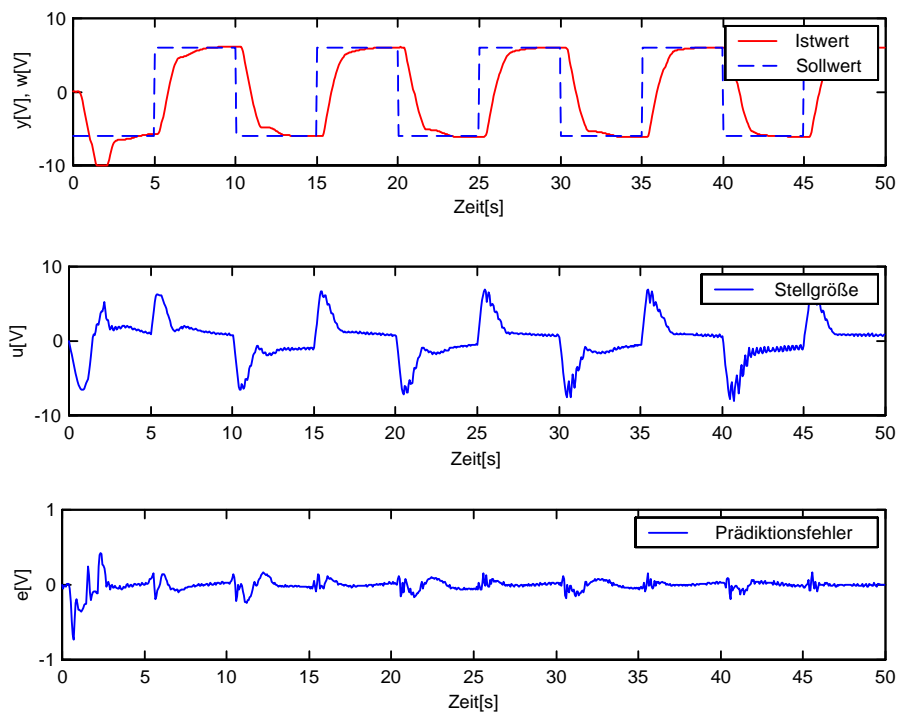
dieser Regelstrecke wurden bereits im Abschnitt 6.1 beschrieben. Der in *Matlab/Simulink* programmierte D-HANNNA-Regler wurde über die *InteractiveRealtime*-Toolbox mit der Regelstrecke verbunden. Bei der Realisierung dieses

Reglers wurde das im Abschnitt 3.2.1.2 vorgestellte Tiefpassfilter mit dem Filterkoeffizienten  $T_f = 0.93$  zur Verbesserung des Stellverhaltens berücksichtigt und die Abtastzeit zu  $T = 60$  ms gewählt. Die Position der Lastmasse (Regelgröße) im Bereich von 0 mm bis 175 mm und die Eingangsspannung des Magnetventils im Bereich von -24 V bis 24 V (Stellgröße) wurden auf den Bereich  $\pm 10$  V normiert. Als Referenzregler diente der HANNNA-Regler und bei beiden Reglern betrug der Gewichtungsfaktor des LQ-Reglers 120. Während der ersten Versuche an der Anlage wurden verschiedene Netzkonfigurationen für ANN-1 und ANN-2 getestet. Die für die durchgeführte Untersuchung eingestellten Netzparameter sind in der Tabelle 6.2 zusammengefasst.

**Tabelle 6.2:** Netzparameter des HANNNA- und des D-HANNNA-Reglers

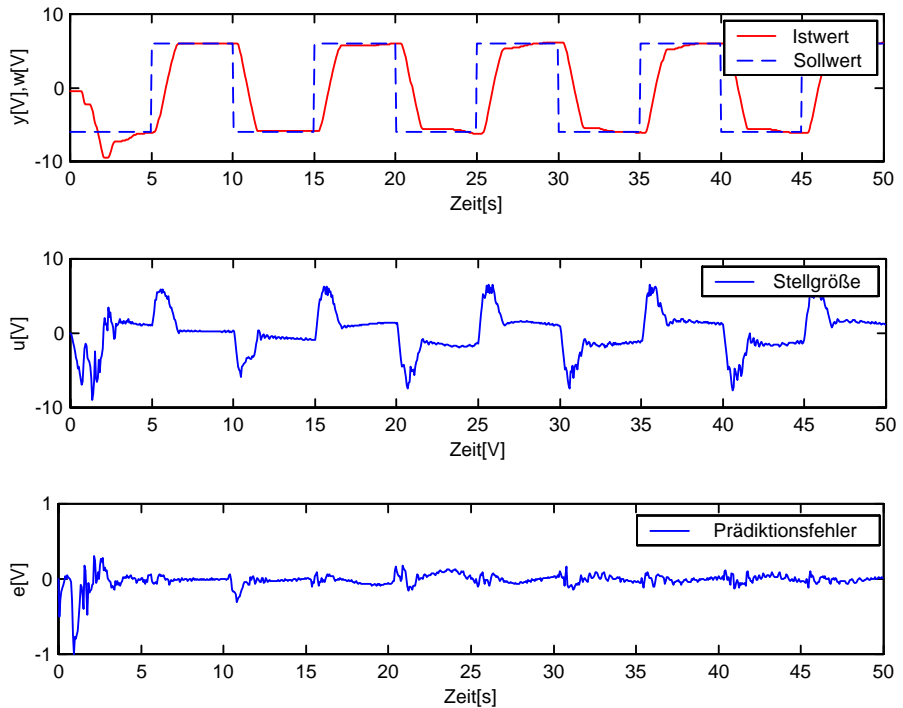
Netz	$\gamma$	Zentren der Basisfunktionen
ANN-1	2.5	-10 -8.5 -7.5 -6 -5 -4 -3 -2 2 3 4 5 6 7.5 8.5 10
ANN-2	1.6	-5 -2.5 -1.5 -1 -0.5 -0.2 0.2 0.5 1 1.5 2.5 5

Die Stärke des dualen Reglers zeigt sich im Fall eines großen Parameterschätzfehlers, der hauptsächlich während der Anfangsadaption und bei schnellen Regelstreckenänderungen auftritt. Bei kleinen Schätzfehlern geht das Stellverhalten des dualen Reglers in das eines Reglers, der auf dem Gewissheitsprinzip basiert, über. Um die Eigenschaften zu untersuchen, wurde deshalb die Versuchszeit auf 50 s beschränkt und als Sollwert ein rechteckförmiges Signal mit einer Amplitude von 7 V und einer Frequenz von 0.1 Hz gewählt. Die Verläufe des Sollwertes  $w$ , der Istwertes  $y$ , des Stellsignals  $u$  und des Prädiktionsfehlers  $e$  sind im Bild 6.21 für den nicht dualen HANNNA-Regler und im Bild 6.22 für den D-HANNNA-Regler dargestellt. Wie schon im Abschnitt 6.1 beschrieben, zeigt der HANNNA-Regler gutes Regelverhalten an der Laboranlage. Der Sollwert bleibt innerhalb der Begrenzung und der Prädiktionsfehler sinkt nach ca. 45 s auf einen Wert unter 0.17 V. Nur während der ersten 5 s fährt der Zylinder für ca. 1 s in die Begrenzung. Vergleicht man dieses Ergebnis mit der D-HANNNA-Regelung, lassen sich trotzdem die Eigenschaften des



**Bild 6.21:** HANNNA-Regelung der Laboranlage „Hydraulischer Vorschubantrieb“

dualen Reglers sehr schön erkennen. Für die gesamte Versuchszeit erhält man ebenfalls gutes Regelverhalten. Im Detail ist zu erkennen, dass, bedingt durch das vorsichtige Stellen bei großen Parameterfehlern, die Regelgröße nur auf einen Wert von -9.4 V absinkt, und damit nicht die Begrenzung erreicht. Die Überschwingweite ist somit wenigstens 10% des Sollwertes kleiner als beim HANNNA-Regler. Das Stellsignal ist in den ersten 5 s unruhiger. Dies wird durch das zusätzliche Erregungssignal zur Verbesserung der Parameteridentifikation verursacht. Nach 10 s Versuchszeit ist das Verhalten der Regelstrecke bei beiden Reglern nahezu gleich, wobei mit dem D-HANNNA-Regler ein Soll-



**Bild 6.22:** Duale HANNNA-Regelung der Laboranlage „Hydraulischer Vorschubantrieb“

wertsprung etwas schneller ausgeregelt wird. Im Bereich von 45 s bis 50 s ist der Prädiktionsfehler unter 0.12 V abgesunken und somit 25% geringer als beim HANNNA-Regler.

Zusammengefasst ist deutlich, dass die Leistungsfähigkeit des HANNNA-Reglers durch die Erweiterung zu einem dualen Regler wesentlich gesteigert werden kann. Dies betrifft hauptsächlich das Regelverhalten bei großen Parameterschätzfehlern, aber auch die Qualität der Modellidentifikation bezüglich des Prädiktionsfehlers. Die ausgeprägte Totzone des Stellventils der untersuchten Laboranlage stellt ein in der Praxis häufig auftretendes Problem dar. Durch

die Verwendung des D-HANNNA-Reglers ist es möglich, diese Eingangsnichtlinearität zu kompensieren.

## 6.4 Simulationsbeispiele mit der RPE-WANNNA-Regelung

Für die Untersuchung des WANNNA-Reglers mit dem RPE-Algorithmus zur Parameteridentifikation wird das lineare Teilsystem mit der diskreten Übertragungsfunktion

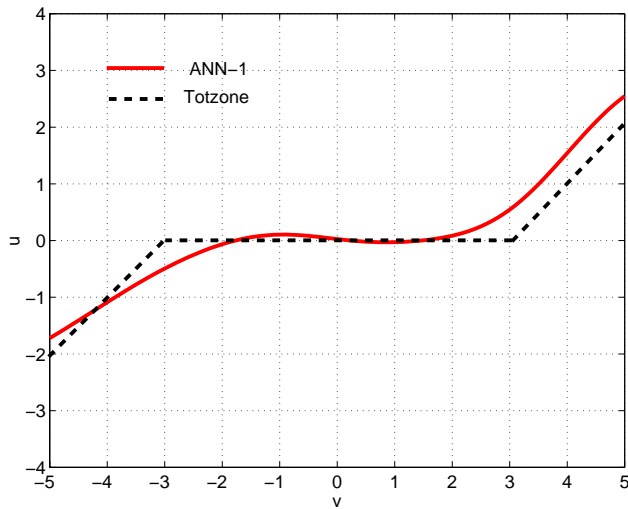
$$G(z^{-1}) = \frac{0.04308z^{-1} + 0.03885z^{-2}}{1 - 0.8z^{-1} + 0.15z^{-2}} \quad (6.5)$$

und eine Totzone mit den Grenzen  $\pm 3$  und den Geradensteigungen eins gewählt. Die Abtastzeit beträgt  $T = 40$  ms und der Gewichtungsfaktor  $\rho$  für den LQ-Regler ist gleich 60. Für das Netz ANN-1 wurden 12 und für ANN-2 14 Neuronen in der versteckten Schicht eingesetzt. Die Parameter der Netze sind in Tabelle 6.3 zusammengefasst. Als Führungsgröße wurde ein Sinussignal mit

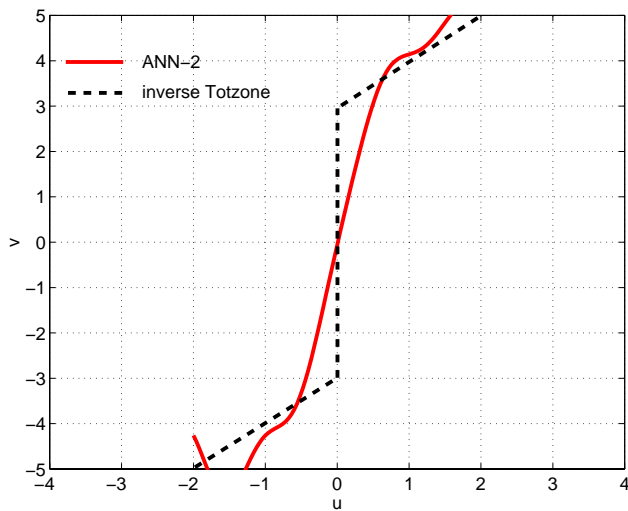
**Tabelle 6.3:** Netzparameter des RPE-WANNNA-Reglers

Netz	$\gamma$	Zentren der Basisfunktionen
ANN-1	2	-5.5 -2.5 -2 -1.5 -1 -0.5 0.5 1 1.5 2 2.5 5.5
ANN-2	1	-5 -4 -3 -2 -1 -0.7 -0.1 0.1 0.7 1 2 3 4 5

der Amplitude eins und einer Frequenz von 0,05 Hz aufgeschaltet. Die Dauer der Simulation und damit die Trainingszeit wurde auf 200 s beschränkt. Im Bild 6.23 sind die Kennlinien von ANN-1 und einer Totzone und im Bild 6.24 ist der Verlauf von ANN-2 sowie eine inverse Totzone dargestellt. Beide Netze haben die jeweilige Zielfunktion am Ende der Trainingszeit gelernt. Es ist jedoch zu erkennen, dass der Trainingserfolg schlechter als bei der HANNNA-Regelung ist. Dies liegt hauptsächlich an der nichtlinearen Parametrisierung und den damit verbundenen Schwierigkeiten bei der Parameteradaptation. Da



**Bild 6.23:** Statische Kennlinien des trainierten Netzes ANN-1 und der Totzone (Trainingszeit : 200 s)



**Bild 6.24:** Statische Kennlinien des trainierten Netzes ANN-2 und der inversen Totzone (Trainingszeit : 200 s)



besonders in der Anfangsphase der Simulation der Schätzfehler groß ist, wurde das Training von ANN-2 erst nach einer Simulationszeit von 50 s aktiviert. Davor wurden die Parameter von ANN-2 so eingestellt, dass es eine Gerade mit der Steigung eins beschreibt. Im Bild 6.25 ist das Führungssignal und das Ausgangssignal mit LQ- und WANNNA-Regelung dargestellt. Deutlich ist die Verbesserung der Regelung nach Einschalten der Kompensation nach 50 s Simulationszeit zu erkennen.

Die Totzone kann aber während der gesamten Simulation nicht vollständig kompensiert werden. Dies liegt an dem schon im Abschnitt 2.5 eingeführten Problem, dass keine eindeutige rechtsseitige Inverse der Totzone existiert. Dieser Sachverhalt ist im Bild 6.26 dargestellt. Für den Wertebereich außerhalb der Totzone kann die Inverse definiert werden. Innerhalb der Totzone werden alle Werte in den Ursprung abgebildet. Bei der Definition der Inversen muss also für einen Eingangswert von null ( $y = 0$ ) der Ausgangswert  $y'$  festgelegt werden. Dieser kann beispielsweise null sein. In diesem Fall führt das zur Beschreibung einer möglichen inversen Totzone mit

$$y'(t) = \begin{cases} \frac{1}{m_r}(y(t) + m_r b_r) & \text{für } y(t) > 0 \\ 0 & \text{für } y(t) = 0 \\ \frac{1}{m_l}(y(t) + m_l b_l) & \text{für } 0 > y(t), \end{cases} \quad (6.6)$$

wobei  $m_l$  die Steigung der linken Geraden und  $b_l$  der zugehörige Achsenabschnitt ist sowie  $m_r$  und  $b_r$  die entsprechenden Parameter für die rechte Seite sind. Dieses Ergebnis entspricht interessanterweise dem Ein-Ausgangsverhalten, welches das Netz ANN-2 gelernt hat. Die Verbesserung des Regelungsverhaltens ist also mit der Kompensation der Nichtlinearität für Ausgangswerte  $y$  außerhalb der Totzone zu erklären.

Fazit dieses Simulationsbeispiels ist, dass ANN zur Kompensation von statischen Ein- und Ausgangskennlinien auch bei nicht invertierbaren Kennlinien verwendet werden können. Durch das Einstellen der Netzgewichte mittels LSE-Schätzung erhält man ein Kompensationsnetz, mit dem die Summe der Fehlerquadrate minimal ist (siehe Abschnitt 3.1.1.3). Diese Eigenschaft ist besonders wichtig bei der *online*-Parameteradaption, wo sich im Allgemeinen

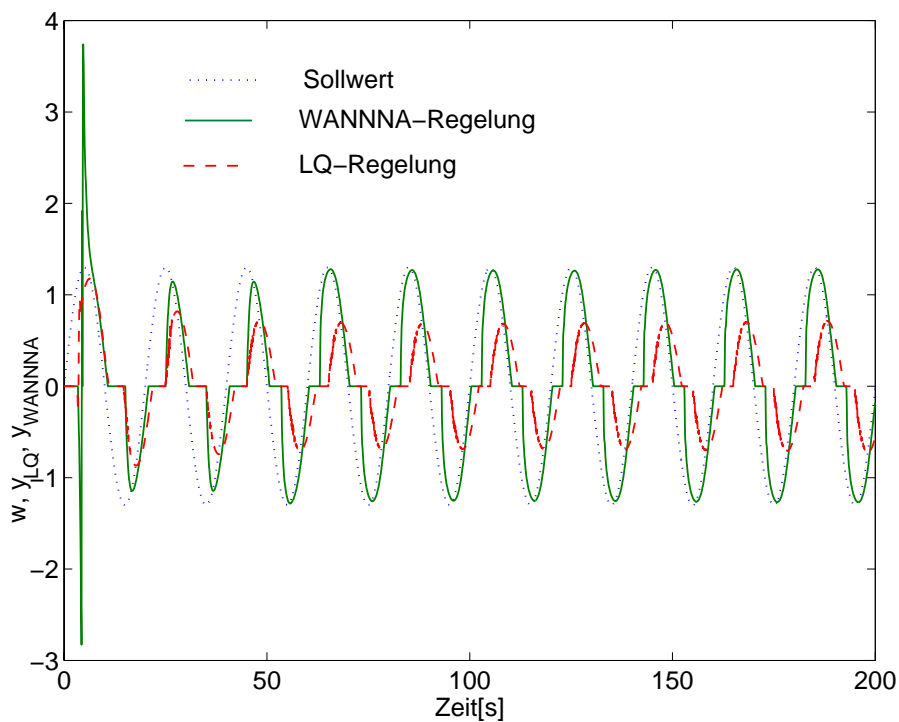
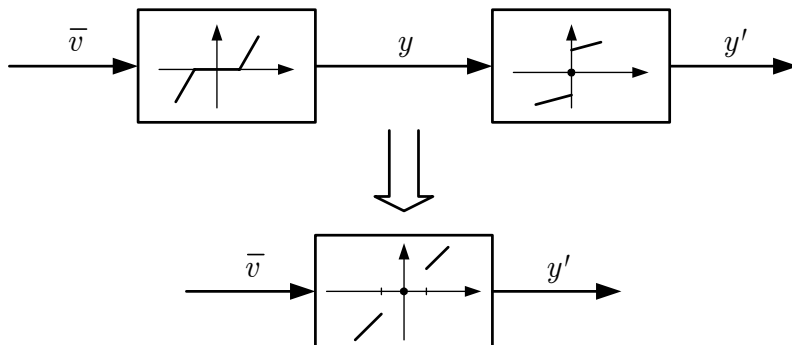


Bild 6.25: Soll- und Istwertverlauf bei LQ- und WANNNA-Regelung



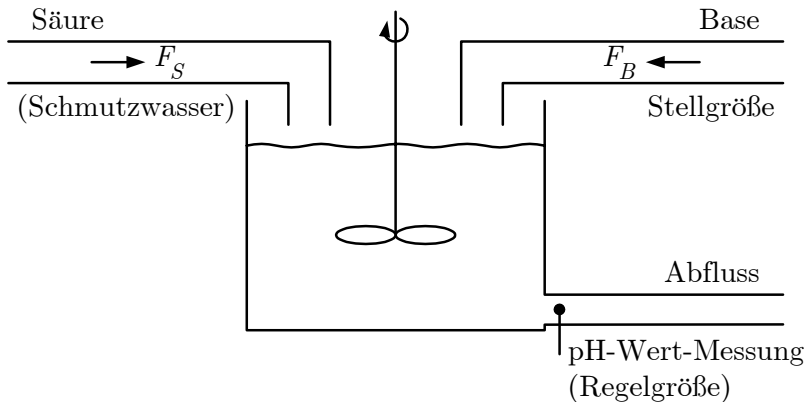
**Bild 6.26:** Darstellung der Totzone und der rechtsseitigen Inversen

mehrdeutige Netze ANN-1 ergeben. Verfahren zur numerischen Invertierung müssen hier scheitern.

## 6.5 pH-Wert-Regelung eines Neutralisationsreaktors

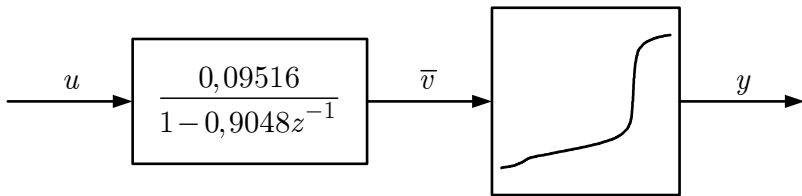
Anhand der pH-Wertregelung eines Neutralisationsreaktors wird die RLS-WANNNA-Regelung untersucht. Die Neutralisation von Schmutzwasser ist in der Verfahrenstechnik ein häufig auftretendes Problem, welches nur schwierig mit Standardreglern zu beherrschen ist. Sie wird in einem Reaktor durchgeführt, wie er im Bild 6.27 dargestellt ist. Dem Reaktor fließt Schmutzwasser, hier als konstanter Säurestrom angenommen, zu, und zur Neutralisation dient als Stellgröße ein in der Durchflussmenge variabler Basestrom. Innerhalb des Reaktors werden diese Flüssigkeiten kontinuierlich vermischt, und der pH-Wert im Abfluss, also die eigentliche Regelgröße, wird mittels eines Sensors gemessen. Der pH-Wert hängt logarithmisch von der Base- und Säurekonzentration ab, so dass die statische Verstärkung dieser Regelstrecke stark vom jeweiligen pH-Wert abhängig ist.

Das physikalische Modell dieser Anlage basiert auf Massenbilanzen der Ba-



**Bild 6.27:** Schematische Darstellung eines Neutralisationsreaktors

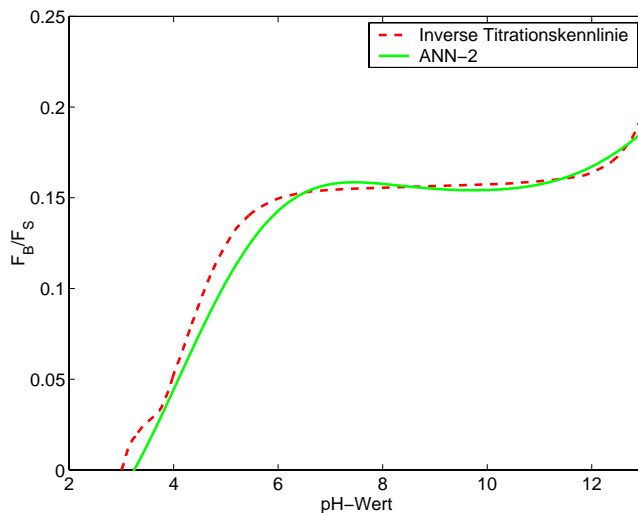
se und Säure, und bei Berücksichtigung der Elektroneutralisationsbedingungen kann eine Beziehung zwischen Basezufluss und pH-Wert angegeben werden (Klatt, 1995). Als Ergebnis der Modellierung erhält man eine nichtlineare Zustandsraumbeschreibung, die sich nicht direkt für einen Reglerentwurf eignet. Deshalb wird für den Entwurf des WANNNA-Reglers der Neutralisationsreaktor mit einem Wiener-Modell beschrieben. Das Zeitverhalten der Anlage, bedingt durch die chemische Reaktion, die Mischung der Flüssigkeiten und die pH-Wertmessung, wird mit einem linearen Verzögerungsglied 1. Ordnung approximiert, dessen Zeitkonstante vom Aufbau des Reaktors abhängig ist. Diese wurde für die Simulation zu  $T = 50$  s gewählt. Die Ausgangsnichtlinearität ergibt sich aus der Titrationskennlinie. Diese Kennlinie zeigt den sich im Gleichgewicht einstellenden pH-Wert als Funktion des Verhältnisses von Basen- ( $F_B$ ) und Säurezufluss ( $F_S$ ). Anders als bei den zuvor betrachteten statischen Kennlinien, ist diese nicht ursprungssymmetrisch. Nach Diskretisierung des linearen Teilmodells mit einem Halteglied nullter Ordnung und einer Abtastzeit von 5 s ist das Simulationsmodell entsprechend der Darstellung im Bild 6.28 gegeben. Die dargestellte Titrationskennlinie wurde mittels Messungen an einer Laboranlage des Lehrstuhls für Elektrische Steuerung und



**Bild 6.28:** Simulationsmodell eines Neutralisationsreaktors

Regelung, Ruhr-Universität Bochum bestimmt.

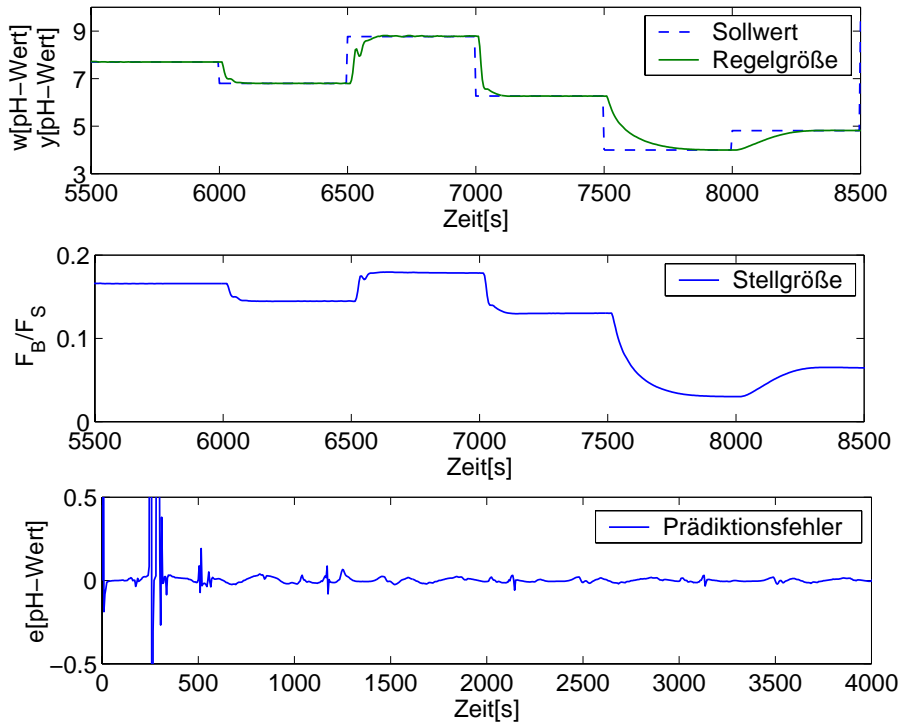
Zur Regelung dieser Regelstrecke wurde der WANNNA-Regler mit der im Abschnitt 4.4 beschriebenen Parameteradaption implementiert. Durch die direkte Bestimmung der Gewichte des Netzes ANN-2 wird bei dieser Realisierung das Netz ANN-1 nicht benötigt. Das Netz ANN-2 bestand bei dieser Simulation aus elf Neuronen in der versteckten Schicht, deren Zentren  $\mathbf{c}$  bei  $[1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11]$  lagen. Der Formparameter  $\mathbf{w}$  wurde für alle Neuronen einheitlich auf 2.3 und der Gewichtungsfaktor  $\rho$  des LQ-Reglers auf 80 gesetzt. Als Sollwert wurde in den ersten 4000 s der Simulation die Überlagerung eines Sinussignals mit der Frequenz 1 mHz und einer Amplitude von 5 mit einem konstanten Wert von 7 verwendet. Durch die Anregung der Regelstrecke im gesamten Arbeitsbereich vom pH-Wert 2 bis zum pH-Wert 12 konnte die inverse Titrationskurve, wie im Bild 6.29 dargestellt, sehr gut gelernt werden. Nach 4000 s wechselte der Sollwert von dem Sinussignal auf ein pulsformiges Zufallssignal mit einer Pulsbreite von 500 s. Im Bild 6.30 sind die Verläufe des Sollwertes, der Regelgröße und der Stellgröße im Bereich 5500 s–8500 s und der Prädiktionsfehler im Bereich 0 s–4000 s dargestellt. Die Regelgröße folgt dem Sollwert im gesamten Arbeitsbereich mit geringer Zeitverzögerung und ohne Überschwüngen. Dies ist ein interessantes Ergebnis, da sich die statische Verstärkung der Regelstrecke, bedingt durch den Verlauf der Titrationskurve, von 0.19 bei einem pH-Wert von 4 bis 13.3 bei einem pH-Wert von 7 ändert. Diese starke Änderung führt bei Verwendung eines PI-Reglers zu schwingendem Verhalten des geschlossenen Regelkreises (Klatt, 1995). Die Stellgröße verläuft ruhig und



**Bild 6.29:** Statische Kennlinien des Netzes ANN-2 und der inversen Titrationskennlinie

kann mit dem Gewichtungsfaktor  $\rho$  des LQ-Reglers an eventuelle Beschränkungen einer Anlage angepasst werden. Wie zu erwarten, ist der Prädiktionsfehler zu Beginn der Simulation groß. Nach ca. 1500 s ist der Fehler des pH-Wertes aber unter eine Schranke von 0.05 gesunken.

Die Ergebnisse zeigen, dass der WANNNA-Regler zur Regelung eines Neutralisationsreaktors sehr gut geeignet ist. Die Inverse der Titrationskurve wurde mit kleiner Abweichung gelernt und durch die Kompensation wurde gutes Regelverhalten über den gesamten Arbeitsbereich erzielt.



**Bild 6.30:** Simulation der pH-Wertregelung eines Neutralisationsreaktors mit dem RLS-WANNNA-Regler

## 7 Zusammenfassung

Die adaptive Regelung nichtlinearer dynamischer Regelstrecken ist für den Regelungstechniker ein ausgesprochen anspruchsvolles Aufgabengebiet. Das bei der Reglersynthese überwiegend gewählte Vorgehen ist die Linearisierung der Regelstrecke in einem bestimmten Arbeitspunkt oder Arbeitsbereich. Diese Vereinfachung führt bei Arbeitspunktänderungen zu unerwünschtem Verhalten des Regelkreises. Ein nicht optimaler Verlauf der Regelgröße oder der Stellgröße bedeutet immer einen höheren Energieverbrauch und damit erhöhte Betriebskosten, die bei technischen Anlagen in einem ausgewogenen Verhältnis zum erzeugten Produkt stehen müssen. Erst in jüngster Zeit wurden adaptive Regelverfahren beschrieben, die auf nichtlinearen Modellen aufbauen. Diese Weiterentwicklung führt im Allgemeinen zu aufwendigeren Regelungsstrukturen, man erzielt aber ein deutlich besseres Regelverhalten des geschlossenen Regelkreises. Eine kompakte Darstellung der aktuellen Ansätze zur nichtlinearen adaptiven Regelung wurde im Kapitel 2 gegeben.

Für Regelstrecken die mit einem Hammerstein-Modell, einem Wiener-Modell oder in der NPSF-Zustandsraumdarstellung beschreibbar sind, wurden in dieser Arbeit fünf neue adaptive Regler vorgestellt. Der Grundgedanke bei diesen Reglerstrukturen war existierendes Wissen über die Regelstrecke, wie beispielsweise die Modellstruktur, zu berücksichtigen und unbekannte Teile mit künstlichen neuronalen Netzen zu approximieren. Das Ergebnis dieser Vorgehensweise ist die ANNNA-Regelung.

Die ersten beiden Reglerstrukturen, beschrieben im Kapitel 3, ermöglichen die Regelung von Hammerstein-Systemen. Hierbei wurde beim HANNNA-



Regler die unbekannte statische Eingangsnichtlinearität mit einem ANN approximiert. Ein zweites Netz, welches den inversen Verlauf lernt, wurde dann zur Kompensation vor den Eingang der Regelstrecke geschaltet. Für das jetzt lineare System, bestehend aus Kompensationsnetz und Regelstrecke, wurde ein LQ-Regler implementiert. Zur Parameter- und Gewichtsadaption diente das RLS-Verfahren, welches die rasche Konvergenz der Schätzung sichert. Als Erweiterung des HANNNA-Reglers wurde dann der D-HANNNA-Regler vorgeschlagen. Zusätzlich zu den Komponenten des HANNNA-Reglers sind bei diesem Regler eine Bewertung des Parameterschätzfehlers und ein Erregungssignal zur Verringerung des Prädiktionsfehlers realisiert. Diese Erweiterung erwies sich speziell bei der Verwendung von ANN als hilfreich, da während der Anfangsadaption das Vorwissen über die zu approximierende Nichtlinearität sehr gering ist. Durch den ganz speziellen Einsatz der ANN erhält man schon mit einer kleinen Anzahl von Neuronen sehr geringe Schätzfehler für beide Strukturen.

Für die Regelung von Wiener-Systemen wurden im 4ten Kapitel dieser Arbeit ebenfalls zwei Regler vorgeschlagen. Bei diesem Regelstreckentyp befindet sich die statische Nichtlinearität im Ausgang. Um sie zu kompensieren, wurden zwei Kompensationsnetze verwendet. Eines ist der Regelstrecke nachgeschaltet und erzeugt somit das Ausgangssignal des linearen Teilsystems der Regelstrecke. Mit dem zweiten Netz wird der Sollwert transformiert, so dass der transformierte Sollwert dem des linearen Teilsystems entspricht. Mit diesen beiden Netzen wurde also erreicht, dass die Ausgangsnichtlinearität keinen Einfluss auf den geschlossenen Regelkreis mehr hat. Die beiden transformierten Signale wurden wiederum einem LQ-Regler zugeführt. Die eigentliche Schwierigkeit bei diesem Regler ist die Parameteridentifikation, da das Wiener-System nichtlinear in den unbekannten Parametern ist. Bei dem ersten Ansatz (RPE-WANNNA-Regelung) wurde deshalb das RPE-Verfahren zur Identifikation der Parameter des linearen Teilsystems und der Gewichte des ANN verwendet. Basierend auf dem ersten ANN wurde dann das Kompensationsnetz trainiert. Da für die Regelung nur das Kompensationsnetz benötigt wird, wurde dieses bei dem zweiten Ansatz (RLS-WANNNA-Regelung) direkt in die Modellbeschei-

bung eingesetzt. Man erhält durch diesen Schritt eine Darstellung, die linear in den unbekannten Parametern ist, wodurch wieder Standardschätzverfahren, wie beispielsweise das RLS-Verfahren, zur Parameterschätzung eingesetzt werden können.

Im Kapitel 5 wurde ein erweiterter *Adaptive Backstepping*-Regler beschrieben. Zunächst wurde hierfür die NPSF-Zustandsraumbeschreibung als Verallgemeinerung der PSF-Zustandsraumbeschreibung definiert. Anders als bei der PSF-Form ist die NPSF-Form nichtlinear in den unbekannten Regelstreckenparametern. Durch die Einführung von jeweils einem ANN zur Approximation der nichtlinearen Funktionen in den einzelnen Zuständen hat diese Regelungsstruktur wieder den Vorteil, dass die Netzstrukturen klein gehalten werden können und das Regelsystem flexibel auf Strukturänderungen reagieren kann. Für den geschlossenen Regelkreis ergeben sich somit, der Ordnung der Regelstecke entsprechend,  $n$  ANN, deren Gewichte mit Hilfe des *Backstepping*-Algorithmus berechnet wurden. Zusätzlich wurde gezeigt, dass trotz dieser Erweiterung alle Systemsignale begrenzt bleiben.

Die Erprobung der neuen Regler für Hammerstein- und Wiener-Systeme wurde im Kapitel 6 beschrieben. Hierfür wurden Simulationen mit typischen nichtlinearen Kennlinien, wie beispielsweise einer Totzone oder einer durch Reibungseffekte bedingten Kennlinie, durchgeführt. Desweiteren wurden die Reglerstrukturen an einem Modell eines Neutralisationsreaktors und an der Laboranlage „Hydraulischer Vorschubantrieb“ untersucht. In diesen Studien zeigte sich, dass sich durch die Einführung von ANN zur Kompensation der unbekannten Nichtlinearitäten das Regelverhalten im Vergleich mit einem Regler ohne Kompensation deutlich verbesserte. Die jeweiligen Kennlinien wurden nahezu exakt gelernt und es zeigte sich, dass das Stellsignal ruhiger verlief und dass sich durch das Einführen des dualen Reglers zusätzlich das Überschwingen der Regelgröße während der Adaptionsphase und der Prädiktionsfehler verringerten. Die Umsetzung der Regler für Echtzeitanwendungen erwies sich als unproblematisch. Simulationsergebnisse des erweiterten *Backstepping*-Reglers zeigten ebenfalls seine deutliche Überlegenheit gegenüber dem Standardregler. Nach einer sprunghaftigen Änderung der Regelstrecke wurde das geschlossene

Regelsystem mit einem Standardregler instabil. Der erweiterte Regler hingegen stabilisierte die veränderte Regelstrecke nach einer kurzen Adaptionsphase.

Zur Synthese der vorgestellten Reglerstrukturen wird hauptsächlich die Ordnung der Regelstrecke benötigt. Bei allen Ansätzen wurden gute Regelergebnisse erzielt mit Netzen, die aus weniger als 30 Neuronen bestehen. Der durch die Integration der ANN zusätzlich entstehende Rechenaufwand ist somit gering und bei Betrachtung der erzielbaren Verbesserungen bezüglich der Reglerleistung in jedem Fall gerechtfertigt. Ein weiterer interessanter Aspekt dieser Reglerstrukturen ist die durch den kompakten Aufbau bedingte mögliche Implementierung auf einem Mikroregler, womit die Realisierung für praktische Problemstellungen gegeben ist.

Durch die Spezialisierung auf die besprochenen Regelstreckenstrukturen ergaben sich leistungsfähige adaptive Regler, welche kompakt sind und sich mit der heutzutage zur Verfügung stehenden Rechenleistung einfach implementieren lassen. Bei der Erprobung wurde hierbei deutlich, dass diese Spezialisierung für den praktischen Einsatz nicht wesentlich ist. Viele Regelstrecken aus unterschiedlichen Bereichen können mit Hilfe der untersuchten Strukturen beschrieben werden.

# Literaturverzeichnis

- AGARWAL, M. A systematic classification of neural-network-based control. *IEEE Control System Magazin*, 17(2):75–93, 1997.
- AL-DUWAISH, H., M. N. KARIM und V. CHANDRASEKAR. A new method for the identification and control of Hammerstein model. In *Proc. of the 29th Annual Conference an Information Sciences and Systems*, Seiten 668–673. 1995.
- AL-DUWAISH, H., M. N. KARIM und V. CHANDRASEKAR. Use of multilayer feedforward neural networks in identification and control of Wiener model. *IEE Proc. Control Theory and Applications*, 143(3):255–258, 1996.
- AL-DUWAISH, H. und M. N. KARIM. A new method for the identification of Hammerstein model. *IEE Proc. Control Theory and Applications*, 33(10):1871–1875, 1997.
- ÅSTRÖM, K. A. und B. WITTENMARK. Problems of identification and control. *Journal of Mathematical Analysis & Applications*, 34(1):90–113, 1971.
- ÅSTRÖM, K. A. und B. WITTENMARK. *Computer-Controlled Systems: Theory and Design*. London: Prentice-Hall, 3 Auflage, 1996.
- BAI, E. W. An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems. *Automatica*, 34(3):333–338, 1998.

- BERSTECHE, R. *Entwurf eines adaptiven Fuzzy sliding-mode-Reglers*. Dissertation, Ruhr-Universität Bochum, Fortschr.-Ber. VDI Reihe 8 Nr. 684, Düsseldorf: VDI-Verlag, 1998.
- BRONSTEIN, I. N., K. A. SEMENDJAJEW, G. MUSIOL und H. MÜLIG. *Taschenbuch der Mathematik*. Frankfurt a. M.: Verlag Harri Deutsch, 4 Auflage, 1999.
- BROOMHEAD, D. S. und D. LOWE. Multivariable functional interpolation and adaptive networks. *Journal of Complex Systems*, 2:321–355, 1988.
- CHEN, S. und S. A. BILLINGS. Neural networks for nonlinear dynamic system modelling and identification. *Int. Journal of Control*, 56(2):319–346, 1992.
- EMELJANOV, S. V. *Automatische Regelsysteme mit veränderlicher Struktur*. München: R. Oldenbourg Verlag, 1969.
- ERKEN, G. Erstellung einer *Matlab/Simulink*-Bibliothek zur Simulation von Stellgeräten unter Berücksichtigung nichtlinearer Eigenschaften. Dipl.-Arb. ESR-9804, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1999.
- FABRITZ, N. *Ein offenes Automatisierungssystem für einen mehrgliedrigen elastischen Manipulator*. Dissertation, Ruhr-Universität Bochum, Düsseldorf: VDI-Verlag, 1997.
- FILATOV, N. und H. UNBEHAUEN. Adaptive dual control for continuous-time systems: A simple example. In *Proc. of IEE International Conference on Control*, Band 1, Seiten 39–43. Swansea, UK, 1999.
- FILATOV, N. und H. UNBEHAUEN. Survey of adaptive dual control methods. *IEE Proc. Control Theory and Applications*, 147(1):118–128, 2000.
- FRADKOV, A. L., I. V. MIROSHNIK und V. O. NIKIFOROV. *Nonlinear and Adaptive Control of Complex Systems*. Kluwer Academic Publishers, 1999.

- FRADKOV, A. L. und A. A. STOTSKY. Speed gradient adaptive control algorithms for mechanical systems. *Int. Journal of Adaptive Control and Signal Processing*, 6(3):211–220, 1992.
- FRADKOV, A. L. Speed-gradient scheme and its application in adaptive control problems. *Automation and Remote Control*, 40(9):1333–1342, 1979.
- FREEMAN, R., M. KRSTIĆ und P. V. KOKOTOVIĆ. Robustness of adaptive nonlinear control to bounded uncertainties. In *Proc. of IFAC World Congress*, Band F, Seiten 329–334. San Francisco, USA, 1996.
- GIROSI, F. und T. POGGIO. Networks and the best approximation property. *Biological Cybernetics*, 63(3):169–176, 1990.
- GOODWIN, G. C. und K. S. SIN. *Adaptive Filtering Prediction and Control*. London: Prentice-Hall, 1984.
- HAGAN, M. T. und H. B. DEMUTH. Neural networks for control. In *Proc. of American Control Conference*, Seiten 1642–1656. 1999.
- HUNT, K. J., D. SBARBARO, R. ZBIKOWSKI und P. J. GAWTHROP. Neural networks for control systems — a survey. *Automatica*, 28(6):1083–1112, 1992.
- ISIDORI, A. *Nonlinear Control Systems*. Berlin: Springer-Verlag, 1989.
- JUNGE, T. F. „On-Line“-Identifikation und lernende Regelung nichtlinearer Regelstrecken mittels neuronaler Netze. Dissertation, Ruhr-Universität Bochum, Fortschr.-Ber. VDI Reihe 8 Nr. 807, Düsseldorf: VDI-Verlag, 1999.
- KANELLAKOPOULOS, I., P. V. KOKOTOVIĆ, R. MARINO und P. TOMEI. Adaptive control of nonlinear systems with partial state feedback. In *Proc. of European Control Conference*, Seiten 1322–1327. Grenoble, France, 1991a.
- KANELLAKOPOULOS, I., P. V. KOKOTOVIĆ und R. H. MIDDLETON. Indirect adaptive output-feedback control of a class of nonlinear systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 2714–2719. Honolulu, Hawaii, 1990.

- KANELLAKOPOULOS, I., P. V. KOKOTOVIĆ und A. S. MORSE. Adaptive output-feedback control of a class of nonlinear systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 1082–1087. Brighton, England, 1991b.
- KANELLAKOPOULOS, I., P. V. KOKOTOVIĆ und A. S. MORSE. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Trans. AC*, 36(11):1241–1253, 1991c.
- KANELLAKOPOULOS, I., M. KRSTIĆ und P. V. KOKOTOVIĆ.  $\kappa$ -Adaptive control of output-feedback nonlinear systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 1061–1066. San Antonio, Texas, 1993.
- KARSENTI, L., F. LAMNABHI-LAGARRIGUE und G. BASTIN. Backstepping technique extended to nonlinear parameterization. In *Proc. of IFAC World Congress*, Seiten 179–184. San Francisco, USA, 1996.
- KLATT, K. U. *Nichtlineare Regelung chemischer Reaktoren mittels exakter Linearisierung und Gain-Scheduling*. Dissertation, Universität Dortmund, Schriftenreihe des Lehrstuhls für Anlagensteuerung, Aachen: Verlag Shaker, 1995.
- KNOHL, T. und H. UNBEHAUEN. KNN-NAC – Ein Verfahren zur Synthese adaptiver Regler für stark nichtlineare Regelstrecken mit schnell veränderlicher Dynamik durch Einsatz von KNN. Interner Bericht ESR 9603, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1996.
- KNOHL, T. und H. UNBEHAUEN. Adaptive control of linear systems with unknown memoryless input nonlinearities using ANN. In *Proc. of IFAC Workshop on Adaptive Systems in Control and Signal Processing*, Seiten 446–471. Glasgow, UK, 1998.
- KNOHL, T. und H. UNBEHAUEN. Adaptive position control of electrohydraulic servo systems using ANN. *Mechatronics*, 10(1-2):127–143, 2000a.

- KNOHL, T. und H. UNBEHAUEN. ANNNAC – extension of adaptive backstepping algorithm with artificial neural networks. *IEEE Proc. Control Theory and Applications*, 147(2):177–183, 2000b.
- KNOHL, T. und H. UNBEHAUEN. Nichtlineare adaptive Regelung von Hammerstein-Systemen mittels RBF-Netze. *Automatisierungstechnik*, 48(11):547–555, 2000c.
- KNOHL, T., W. XU und H. UNBEHAUEN. Dual adaptive neural control and its application to Hammerstein systems. In *International Conference on Control, Automation, Robotics and Vision*, Seite FP6.4. Singapore, 2000a.
- KNOHL, T., H. ZENG und H. UNBEHAUEN. Adaptive position control of a flexible manipulator using ANNNAC. In *Proc. of IFAC Symposium on System Identification*, Seiten ThMD1–2. Santa Barbara, California, 2000b.
- KNOHL, T. Erstellung einer *Matlab*-Toolbox zur Systemidentifikation mittels RBF-Netze. Stud.-Arb. ESR-9413, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1994a.
- KNOHL, T. Künstliche neuronale Netze mit dynamischen Neuronen zur Identifikation nichtlinearer Systeme. Dipl.-Arb. ESR-9422, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1994b.
- KNOHL, T. Adaptive Regelung nichtlinearer Systeme : Zusammenfassung. Interner Bericht ESR 9602, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1996.
- KOKOTOVIĆ, P. V., M. KRSTIĆ und I. KANELAKOPOULOS. Backstepping to passivity: Recursive design of adaptive systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 3276–3280. Tucson, Arizona, 1992.
- KOKOTOVIĆ, P. V. *Lecture Notes in Control and Information Sciences: Foundations in Adaptive Control*. Berlin: Springer Verlag, 1991.



- KORTMANN, M. und H. UNBEHAUEN. Application of a recursive prediction error method to the identification of nonlinear systems using Wiener model. In *Proc. of IMACS Symp. on Modelling and Simulation for Control of Lumped and Distributed Parameter Systems*, Seiten 281–285. 1986.
- KORTMANN, M. *Die Identifikation nichtlinearer Ein- und Mehrgrößensysteme auf der Basis nichtlinearer Modellansätze*. Dissertation, Ruhr-Universität Bochum, Fortschr.-Ber. VDI Reihe 8 Nr. 177, Düsseldorf: VDI-Verlag, 1988.
- KRASOVSKII, A. A. Optimal algorithms in problems of identification with an adaptive model. *Avtom. Telemekh.*, Seiten 75–82, 1976.
- KRSTIĆ, M., I. KANELAKOPOULOS und P. V. KOKOTOVIĆ. *Nonlinear and Adaptive Control Design*. New York: John Wiley & Sons, 1995.
- KRSTIĆ, M. und P. V. KOKOTOVIĆ. Adaptive nonlinear control with nonlinear swapping. In *Proc. of IEEE Conference on Decision and Control*, Seiten 1073–1080. San Antonio, Texas, 1993.
- KRSTIĆ, M. und P. V. KOKOTOVIĆ. Adaptive nonlinear design with controller-identifier separation and swapping. *IEEE Trans. AC*, 40(3):426–440, 1995.
- KUNG, M. C. und B. F. WOMACK. Discrete time adaptive control of linear dynamic systems with a two-segment piecewise-linear asymmetric nonlinearity. *IEEE Trans. AC*, 29(2):170–172, 1984.
- LEWIS, F. L., J. CAMPOS und R. SELMIC. On adaptive critic architectures in feedback control. In *Proc. of IEEE Conference on Decision and Control*, Seiten 1677–1684. Phoenix, Arizona, 1999.
- LIANG, F. und H. A. ELMARAGHY. Multistep localized adaptive learning RBF networks for nonlinear system identification. In *Proc. of European Control Conference*, Seiten 111–116. Groningen, Netherland, 1993.
- LI, Y. und E. B. LEE. Model reference sliding control of nonlinear systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 289–290. Tucson, Arizona, 1992.

- LJUNG, L. und T. SÖDERSTRÖM. *Theory and Practice of Recursive Identification*. Cambridge: MIT Press, 1986.
- LLANES-SANTIAGO, O., M. T. ESANDI und H. SIRA-RAMÍREZ. Adaptive chattering-free sliding mode control of nonlinear uncertain systems. In *Proc. of European Control Conference*, Seiten 829–834. Groningen, Netherland, 1993.
- MAO, C. Nichtlineare adaptive Lageregelung eines Flugkörpers mit zwei Freiheitsgraden. Stud.-Arb. ESR-9713, Lehrstuhl für Elektrische Steuerung und Regelung, Ruhr-Universität-Bochum, 1997.
- MARINO, R., P. TOMEI, I. KANELAKOPOULOS und P. V. KOKOTOVIĆ. Adaptive tracking for a class of feedback linearizable systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 1081–1086. San Antonio, Texas, 1993.
- MERRIT, H. E. *Hydraulic Control System*. New York: John Wiley & Sons, 1967.
- MOSCA, E. *Optimal, Predictive and Adaptive Control*. Englewood Cliffs, New Jersey: Prentice Hall, 1995.
- NARENDRA, K. S. und K. PARTHASARATHY. Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks*, 1(1):4–27, 1990.
- NG, G. W. *Application of Neural Networks to Adaptive Control of Nonlinear Systems*. New York: John Wiley & Sons, 1997.
- NIJMEIJER, H. und G. CAMPION. Feedback linearization of parametrized nonlinear time-varying system. In *Proc. of European Control Conference*, Seiten 31–36. Groningen, Netherland, 1993.
- NIJMEIJER, H. und A. J. VON DER SCHAFT. *Nonlinear Dynamical Control Systems*. Berlin: Springer-Verlag, 1990.

- ORLOV, Y. V. Sliding mode - model reference adaptive control of distributed parameter systems. In *Proc. of IEEE Conference on Decision and Control*, Seiten 2438–2445. San Antonio, Texas, 1993.
- PAJUNEN, G. Adaptive control of Wiener type nonlinear systems. *Automatica*, 28(4):781–785, 1992.
- POTTMANN, M. und D. E. SEBORG. Identification of nonlinear processes using reciprocal multiquadratic functions. *Journal of Process Control*, 2:189–203, 1992.
- SCHWARZ, H. *Nichtlineare Regelungssysteme*. München: Oldenbourg, 1991.
- SERON, M. M., D. J. HILL und A. L. FRADKOV. Nonlinear adaptive control of feedback passive systems. *Automatica*, 31(7):1053–1060, 1995.
- SETO, D., A. M. ANNASWAMY und J. BAILLIEUL. Adaptive control of a class of nonlinear systems with a triangular structure. In *Proc. of IEEE Conference on Decision and Control*, Seiten 278–283. Tucson, Arizona, 1992.
- SETO, D., A. M. ANNASWAMY und J. BAILLIEUL. Adaptive control of nonlinear systems with triangular structure. *IEEE Trans. AC*, 39(7):1411–1428, 1994.
- ŠLEMIĆ, R. R. und F. L. LEWIS. Dead-zone compensation in motion control systems using neural networks. *IEEE Trans. AC*, 45(4):602–613, 2000.
- SLOTINE, J. J. E. und W. LI. *Applied Nonlinear Control*. London: Prentice Hall, 1991.
- TAO, G. und P. V. KOKOTOVIĆ. Adaptive control of plants with unknown dead-zones. In *Proc. of American Control Conference*, Seiten 2710–2714. Chicago, Illinois, 1992.
- TAO, G. und P. V. KOKOTOVIĆ. Adaptive control of systems with unknown output backlash. In *Proc. of IEEE Conference on Decision and Control*, Seiten 2635–2640. San Antonio, Texas, 1993.

- TAO, G. und P. V. KOKOTOVIĆ. Adaptive control of plants with unknown dead-zones. *IEEE Trans. AC*, 39(1):50–68, 1994.
- TAO, G. und M. TIAN. Design of adaptive dead-zone inverse for non-minimum phase plants. In *Proc. of American Control Conference*, Seiten 2059–2063. Seattle, Washington, 1995.
- TAO, G. Adaptive control of partially known systems. In *Proc. of American Control Conference*, Seiten 3007–3011. Seattle, Washington, 1995.
- UNBEHAUEN, H. *Regelungstechnik III*. Braunschweig: Vieweg, 3 Auflage, 1988.
- UTKIN, V. I. *Sliding Modes in Control Optimazation*. Berlin: Springer-Verlag, 1992.
- UTKIN, V., J. GULDNER und J. SHI. *Sliding Mode Control in Electromechanical Systems*. London: Taylor & Francics, 1999.
- VETCH CONSULTING. *InteractiveRealtime User's Manual*. Bern, Schweiz-land, 1997.
- WON, M. und J. K. HEDRICK. Discrete time adaptive sliding control of nonlinear systems without matching condition. In *Proc. of American Control Conference*, Seiten 1309–1313. Seattle, Washington, 1995.
- XIONG, Y. F. und H. UNBEHAUEN. Adaptive control of systems with a class of direction-dependent nonlinearities. In *Proc. of IMACS Symposium on Identification and System Parameter Estimation*, Seiten 1008–1012. Budapest, Hungary, 1991a.
- XIONG, Y. F. und H. UNBEHAUEN. Adaptive control of systems with unknown dead-zone nonlinearities. In *Proc. of IMACS Symposium MCTS*, Seiten 27–33. Lille, France, 1991b.
- ZENG, H. *Modeling, Identification and Control of a flexible Multi-link Leight-weight Manipulator*. Dissertation, Ruhr-Universität Bochum, Göttingen: Verlag Cuvillier, 1999.

- ZHU, Y. und T. J. CONTROL. Parametric Wiener model identification for control. In *Proc. of IFAC World Congress*, Seiten 37–42. Beijing, China, 1999.
- ZUFIRIA, P. J., J. FRAILE-ARDANUY, R. RIAZA und J. I. ALONSO. Neural adaptive control of non-linear plants via a multiple inverse model approach. *Int. Journal of Adaptive Control and Signal Processing*, 13(4):219–239, 1999.

# Lebenslauf

Name : Torsten Knohl

Geburtsdatum/Ort : 04.01.1968 in Marl

Staatsangehörigkeit : deutsch

Familienstand : verheiratet

Anschrift : Schillerstraße 56, 45768 Marl

E-mail : Torsten.Knohl@epost.de

## Werdegang

1974 – 1978	Goethe-Schule Marl (Grundschule)
1978 – 1981	Ernst-Immel-Schule Marl (Realschule)
1981 – 1984	Erich-Klausener-Schule Dorsten (Realschule)
1984 – 1987	Albert-Schweitzer-Gymnasium Marl
1987 – 1989	Zivildienst
1989 – 1995	Ruhr-Universität Bochum Studium der Elektrotechnik Abschluss: Diplomingenieur
1995 - 31.12.00	Wissenschaftlicher Mitarbeiter am Lehrstuhl für Elektrische Steuerung und Regelung Ruhr-Universität Bochum