

# **Modellbildung mittels eines selbstlernenden Fuzzy–Systems**

K. Küpper

Forschungsbericht Nr. 8/94

Meß-, Steuer- und Regelungstechnik

**Übersicht:** In diesem Forschungsbericht wird ein Algorithmus vorgestellt, mit dem die Parameter eines relationalen Fuzzy–Systems rekursiv so bestimmt werden können, daß es ein gefordertes Ein-/Ausgangsverhalten näherungsweise annimmt. Das zu Grunde liegende Verfahren ist die stochastische Approximation. Die Leistungsfähigkeit des resultierenden selbstlernenden Fuzzy–Modells wird durch die Identifikation eines realen Systems demonstriert.

Gerhard-Mercator-Universität - GH Duisburg

Meß-, Steuer- und Regelungstechnik

Prof. Dr.-Ing. H. Schwarz

# Inhaltsverzeichnis

<b>Tabellenverzeichnis</b>	<b>II</b>
<b>Nomenklatur</b>	<b>III</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Modellstruktur</b>	<b>3</b>
2.1 Fuzzy–Relationalmodell . . . . .	3
2.2 Dynamische Modelle . . . . .	9
<b>3 Identifikationsalgorithmus</b>	<b>12</b>
<b>4 Untersuchtes System</b>	<b>16</b>
4.1 Beschreibung . . . . .	16
4.2 Testdaten . . . . .	17
<b>5 Lösungsansätze</b>	<b>18</b>
5.1 Festlegungen . . . . .	18
5.2 Modell 1 . . . . .	18
5.3 Modellierung als $P_{T_1}$ –System . . . . .	24
5.4 Modell 2 mit Vorbelegung . . . . .	26
5.4.1 Vorbelegung durch Lernen . . . . .	26
5.4.2 Analytische Vorbelegung . . . . .	28
5.5 Modell 3: Inkrementales Modell . . . . .	34
5.6 Modell 4 . . . . .	37
5.7 Modell 5 . . . . .	38
<b>6 Zusammenfassung und Ausblick</b>	<b>41</b>
<b>7 Literatur</b>	<b>44</b>
<b>Anhang</b>	<b>48</b>
<b>A Testsignale</b>	<b>48</b>
<b>B Ergebnisse für Modell 1</b>	<b>51</b>
<b>C Ergebnisse für die Modellierung mit einem <math>P_{T_1}</math>–System</b>	<b>53</b>
<b>D Ergebnisse für Modell 2</b>	<b>55</b>
<b>E Ergebnisse für das inkrementale Modell</b>	<b>58</b>
<b>F Ergebnisse für Modell 4</b>	<b>59</b>

## Tabellenverzeichnis

5.1	Mittlerer quadratischer Fehler für verschiedene Verzögerungen $\tau_1$ und $\tau_2$ nach 100 Lernschritten . . . . .	19
5.2	Mittlerer quadratischer Fehler für verschiedene Signale, Modell 1 . . . . .	22
5.3	Mittlerer quadratischer Fehler für verschiedene Signale, $P_{T1}$ –System . . . . .	25
5.4	Mittlerer quadratischer Fehler für verschiedene Signale, Fuzzy–Modell mit $P_{T1}$ –System vorbelegt . . . . .	27
5.5	Mittlerer quadratischer Fehler für verschiedene Signale, Modell 3 . . . . .	36
5.6	Mittlerer quadratischer Fehler für verschiedene Signale, Modell 4 . . . . .	38
5.7	Mittlerer quadratischer Fehler für verschiedene Signale, Modell 5 . . . . .	40

# Nomenklatur

## Fuzzy-spezifische Größen

$b$	Zahl der Referenzfuzzy-Mengen auf $\mathbb{W}$
$c_i$	Zahl der Referenzfuzzy-Mengen auf $\mathbb{D}_i$
$d$	Zahl der Regelsätze eines Fuzzy-Systems
$\mathbb{D}_i$	Definitionsbereich des Eingangs $i$
$g_{j,l}$	Möglichkeitsgrad (Gewicht) der Regel des Regelsatzes $l$ mit der Referenzfuzzy-Menge $Y_j$ für den Ausgang
$r(p_1, p_2, \dots, p_a, j)$	Element des Tensors $\hat{\mathbf{R}}$
$\mathbf{R}$	Fuzzy-Relation eines kompositionsbasierten Fuzzy-Systems (auch <i>Relationalmatrix</i> genannt)
$\hat{\mathbf{R}}$	Tensor der Möglichkeitsgrade aller Regeln
$\hat{\mathbf{T}}$	Fuzzy-Relation der Ordnung $a$
$\mathbb{W}$	Definitionsbereich der Ausgangsgröße $y$
$x_i$	scharfe Variable des Eingangs $i$
$\hat{x}_{p_i}^i$	Element des Vektors $\hat{\mathbf{X}}^i$
$X_{p_i}^i$	$p_i$ -te Referenzfuzzy-Menge auf $\mathbb{D}_i$
$\hat{\mathbf{X}}^i$	Möglichkeitsvektor, Vektor der Möglichkeitsgrade der Eingangsfuzzy-Menge $\hat{X}^i$ in Bezug auf die Referenzfuzzy-Mengen des Eingangs $i$ .
$\hat{X}^i$	Fuzzy-Menge, die Eingang eines Fuzzy-Systems ist
$y$	scharfe Variable des Ausgangs
$y_{j, \max}$	$y$ -Koordinate des Maximalwerts einer Zugehörigkeitsfunktion aus der vertikalen Aggregation
$\hat{y}$	scharfer Ausgang des Fuzzy-Modells
$\hat{y}_j$	Element des Vektors $\hat{\mathbf{Y}}^i$
$\hat{Y}_j$	$j$ -te Referenzfuzzy-Menge auf $\mathbb{W}$
$\hat{\mathbf{Y}}^i$	Vektor der Erfülltheitsgrade für die Referenzfuzzy-Mengen des Ausgangs
$\hat{Y}$	Fuzzy-Menge, die Ausgang eines Fuzzy-Systems ist
$\hat{Y}_{j,l}$	Fuzzy-Menge, die Ergebnis der Regel des Regelsatzes $l$ mit der Referenzfuzzy-Menge $Y_j$ für den Ausgang ist
$\hat{Y}_j$	Fuzzy-Menge, die Ergebnis aller Regeln mit der Referenzfuzzy-Menge $Y_j$ für den Ausgang ist

$\alpha_{l,i}$	Erfülltheitsgrad der $i$ -ten Partialprämisze des $l$ -ten Regelsatzes
$\alpha_l$	Erfülltheitsgrad der Prämisze des $l$ -ten Regelsatzes
$\mu_{j,l}$	Erfülltheitsgrad der Regel des Regelsatzes $l$ mit der Referenzfuzzy-Menge $Y_j$ für den Ausgang
$\mu_{j,max}$	Maximalwert der Zugehörigkeitsfunktion aus der vertikalen Aggregation
$\mu_{\hat{T}}(p_1, p_2, \dots, p_a)$	Zugehörigkeitsfunktion der Fuzzy-Relation $\hat{T}$
$\mu_{\tilde{Y}}$	Zugehörigkeitsfunktion der Fuzzy-Menge $\tilde{Y}$

## Sonstige Formelzeichen

$a_0, a_1, a_2$	Polynomkoeffizienten
$e(k)$	Fehler zwischen Modell und System
$E[\cdot]$	Erwartungswert
$h_0$	Skalierungsfaktor
$p_1$	Speicherdruck
$p_2$	Systemdruck
$q_m$	mittlerer quadratischer Fehler
$s_i$	Steigungen
$T$	Abtastzeit
$\beta$	Skalierungsfaktor
$\gamma(k)$	Folge von Lernfaktoren (Skalar)
$\boldsymbol{\gamma}(k)$	Folge von Lernfaktoren (Matrix)
$\Theta$	Parametervektor
$\tau_i$	Zeitverzögerungen

## Indizes

$i = 1, 2, \dots, a$	Index über die Eingänge
$j = 1, 2, \dots, b$	Nummerierung der Referenzfuzzy-Mengen des Ausgangs
$l = 1, 2, \dots, d$	Index der Regelsätze
$m = 1, 2, \dots, N$	bezeichnet die einzelnen Meßdatensätze
$p_i \in \{1, 2, \dots, c_i\}$	Nummerierung der Referenzfuzzy-Mengen des Eingangs $i$

## Operatoren

---

$\otimes$	unscharfes kartesisches Produkt
$\circ$	Kompositionsoperator
$\Pi(\tilde{X}^i   X_{p_i(l)}^i)$	Möglichkeitsgrad der Fuzzy-Menge $\tilde{X}^i(x_i)$ in bezug auf die Referenfuzzy-Menge $X_{p_i(l)}^i(x_i)$
$\vee$	Disjunktion zweier Größen
$\bigvee_{i=1}^a$	Disjunktion über mehrere Größen mit dem Index $i$
$\wedge$	Konjunktion zweier Größen
$\bigwedge_{i=1}^a$	Konjunktion über mehrere Größen mit dem Index $i$

# 1 Einleitung

In der Regelungstechnik, aber auch in allen anderen technischen Bereichen, wird die Simulation technischer Systeme regelmäßig und mit großem Nutzen eingesetzt. Dabei werden meist lineare oder nichtlineare Differentialgleichungen verwendet, die vorher durch eine Modellbildung des Systems gefunden wurden. Dieses Vorgehen setzt jedoch relativ umfangreiche A-priori-Informationen über die zu modellierenden Anlagen voraus. Insbesondere müssen geeignete Linearisierungen für Arbeitspunkte gefunden werden oder aber zumindest die Struktur und Art der Nichtlinearitäten (z. B. statisch, dynamisch, Begrenzung, bilineares System) bekannt sein. In der Praxis werden häufig nur lineare Modelle eingesetzt, erst in jüngster Zeit gibt es auch Ansätze zur Nutzung nichtlinearer Systemmodelle (Schwarz 1991, Reuter 1993a, Yin 1994, Jelali 1994). Ein weiteres Kennzeichen vieler Simulationsmodelle ist deren hohe Komplexität.

Einen neuen Ansatz zur Modellierung technischer Systeme stellt Tong (1978, 1980) mit der Fuzzy-Modellbildung vor. Dabei wird der Zusammenhang zwischen Ein- und Ausgangsgrößen mit Hilfe der Fuzzy-Logik beschrieben. Das Verfahren wurde inzwischen stark weiterentwickelt (Pedrycz 1983, 1984a, 1984b, 1985, 1991, van der Rhee und van der Nauta Lemke 1987, 1990, Sugeno u. a. 1986, 1988, 1991, 1993, Takagi und Sugeno 1983, 1985, Xu 1989, Xu und Lu 1987), wobei zwischen Fuzzy-Relationalmodellen und Fuzzy-Funktionalmodellen unterschieden wird. Es wurden bereits gute Ergebnisse erzielt.

Der besondere Vorteil von Fuzzy-Modellen liegt darin, daß sie einen beliebigen, also auch einen nichtlinearen Zusammenhang zwischen Ein- und Ausgängen beschreiben können: Fuzzy-Systeme sind universelle Approximatoren (Wang 1992, 1994, Kosko 1992). Aus diesem Grund muß bei einer Fuzzy-Modellbildung nur relativ wenig Vorwissen über das zu beschreibende System vorliegen. Außerdem kann der Aufbau eines Fuzzy-Modells unter Umständen einfacher als der eines konventionellen Systems sein, da die Abbildung nichtlinearer Effekte keine besonderen Strukturen erfordert. Insbesondere der letzte Punkt macht Fuzzy-Modelle dann attraktiv, wenn komplexe Systeme durch relativ einfache Modelle beschrieben werden sollen. Es sind in Zukunft Echtzeitsimulationen mit Fuzzy-Modellen denkbar, deren Einsatzgebiet von modellbasierten Reglern bis zu Hardware-in-the-loop-Anwendungen reicht. Für diese Anwendungen muß jedoch eine hohe Modellgüte erreicht werden. Insbesondere ist es notwendig, eine Prädiktion über mehrere Abtastschritte oder eine rekursive Auswertung der Modelle zu verwirklichen. In den meisten bisher veröffentlichten Arbeiten finden sich lediglich Einschrittprädiktionen.

In diesem Bericht wird ein neuer Algorithmus zur Fuzzy-Modellbildung vorgestellt, der mit Hilfe eines stochastischen Gradientenverfahrens die Parameter des Relationalmodells optimiert. Die Leistungsfähigkeit des Verfahrens demonstriert die Anwendung auf ein reales hydraulisches System, wobei insbesondere Wert auf die rekursive Auswertung des resultierenden Fuzzy-Modells gelegt wird.

Der nächste Abschnitt erläutert die Struktur des verwendeten Fuzzy-Relationalmodells und dessen Auswertung. Außerdem werden einige spezielle Eigenschaften des verwendeten

Modells näher betrachtet. Daran schließt sich in Abschnitt 3 die Erläuterung des Identifikationsverfahrens an, das auf der sogenannten stochastischen Approximation beruht. Das untersuchte hydraulische System wird, ebenso wie die verwendeten Trainings- und Testsignale in Abschnitt 4 beschrieben. Die Ergebnisse der Fuzzy–Modellbildung sind Inhalt von Abschnitt 5. Sechs verschiedene Lösungsansätze und die dazugehörigen Ergebnisse werden ausführlich vorgestellt und diskutiert. Eine Zusammenfassung und ein Ausblick schließen den Bericht.

## 2 Modellstruktur

### 2.1 Fuzzy–Relationalmodell

Die in diesem Bericht betrachteten Modelle sind relationale Fuzzy–Systeme, wie sie von Pedrycz (1984a) eingeführt wurden. Im Gegensatz zu funktionalen Fuzzy–Systemen besitzen die verwendeten linguistischen Regeln sowohl im Bedingungsteil als auch im Schlußfolgerungsteil linguistische Aussagen. Bei ausschließlicher Betrachtung von Systemen mit mehreren Eingängen und nur einem Ausgang (MISO–Systeme), was ohne Beschränkung der Allgemeinheit möglich ist, nimmt eine einzelne Regel dann die Form

$$\begin{aligned} \text{WENN } & (\text{Eingang 1 ist klein}) \text{ und } (\text{Eingang 2 ist mittel}) \text{ und } \dots \\ & \text{DANN } (\text{Ausgang ist groß}) \mid g \end{aligned} \quad (2.1)$$

an. Dabei bezeichnet  $g > 0$  den Möglichkeitsgrad der Regel und kann als Gewichtung interpretiert werden.

Eine präzisere Definition als die gerade genannte, rein linguistische Beschreibung erlaubt die folgende Form:

Ein Fuzzy–System habe  $a$  Fuzzy–Mengen  $\tilde{X}^i(x_i)$ ,  $i = 1, 2, \dots, a$  als Eingänge. Die Grundbereiche der Eingänge seien  $\mathbb{D}_i$  mit

$$\tilde{X}^i(x_i) : \mathbb{D}_i \rightarrow [0; 1] . \quad (2.2)$$

Außerdem seien auf jedem Grundbereich  $c_i$  sogenannte Referenzfuzzy–Mengen  $X_{p_i}^i$  definiert mit  $i = 1, 2, \dots, a$  und  $p_i = 1, 2, \dots, c_i$ . Der einzige Ausgang  $\tilde{Y}$  des Fuzzy–Systems habe den Grundbereich  $\mathbb{W}$ , auf dem  $b$  Referenzfuzzy–Mengen  $Y_j$  mit  $j = 1, 2, \dots, b$  definiert sind. Dann besitzt das Fuzzy–System

$$d = \prod_{i=1}^a c_i \quad (2.3)$$

Regelsätze der Form

$$\begin{aligned} \text{WENN } & (\tilde{X}^1 \text{ ist } X_{p_1(l)}^1) \text{ und } (\tilde{X}^2 \text{ ist } X_{p_2(l)}^2) \text{ und } \dots \text{ und } (\tilde{X}^a \text{ ist } X_{p_a(l)}^a) \\ & \text{DANN } (\tilde{Y}_{jl} \text{ ist } Y_j) \mid g_{jl} \quad j = 1, 2, \dots, b, \quad l = 1, 2, \dots, d . \end{aligned} \quad (2.4)$$

Dabei besteht jeder Regelsatz aus  $b$  einzelnen Regeln, die den gleichen Voraussetzungsteil haben, sich aber in der Referenzfuzzy–Menge  $Y_j$  für die Folgerung und im Möglichkeitsgrad (oder Gewicht)  $g_{jl}$  unterscheiden. Einige Überlegungen erfordern es, die Gewichte  $g_{jl}$  in einem Tensor anzugeben, der „Gewichtstensor“ genannt werden soll. In der Literatur (Pedrycz 1984a, 1984b, Xu und Lu 1989, Postlethwaite 1991) ist es jedoch üblich, ihn als Relationalmatrix zu bezeichnen:

$$\begin{aligned} \hat{\mathbf{R}} = & \{r(p_1, p_2, \dots, p_a, j) \mid r(p_1, p_2, \dots, p_a, j) = g_{jl(p_1, p_2, \dots, p_a)}\} , \\ & p_i = 1, 2, \dots, c_i, \quad j = 1, 2, \dots, b \end{aligned} \quad (2.5)$$

Diese Bezeichnung wird deshalb auch in diesem Bericht verwendet und nur an den entsprechenden Stellen auf eventuell notwendige Unterscheidungen hingewiesen.

Es folgt eine Erklärung der Auswertung des Fuzzy–Systems: Nach der Fuzzifizierung der scharfen Eingangsgrößen werden bei der hier verwendeten regelbasierten Inferenz zunächst die Partialprämissen ausgewertet. Dabei wird die von Pedrycz (1989) definierte *Möglichkeit* des Eingangs  $\tilde{X}^i(x_i)$  in bezug auf die Referenzmenge  $X_{p_i(l)}^i(x_i)$

$$\Pi(\tilde{X}^i | X_{p_i(l)}^i) = \sup_{x \in \mathbb{D}_i} [\min(\tilde{X}^i(x_i), X_{p_i(l)}^i(x_i))] \quad (2.6)$$

verwendet. Der Operator darf nicht mit dem Produkt (vgl. Gl. 2.3) verwechselt werden. Statt des Minimums kann auch jede andere t–Norm verwendet werden. Das Ergebnis sind die Erfülltheitsgrade der Partialprämissen, also scharfe Werte, die hier mit

$$\alpha_{li} = \Pi(\tilde{X}^i | X_{p_i(l)}^i) \quad (2.7)$$

bezeichnet werden sollen. Die Zusammenfassung der  $\alpha_{li}$  zum Erfülltheitsgrad  $\alpha_l$  der Prämisse des Regelsatzes  $l$  erfolgt durch die Konjunktion, die den Fuzzy–UND–Operator in Gl. (2.4) modelliert:

$$\alpha_l = \bigwedge_{i=1}^a \alpha_{li} \quad . \quad (2.8)$$

Nach der Prämissenauswertung wird der Erfülltheitsgrad einer Regel dadurch bestimmt, daß der Erfülltheitsgrad der Partialprämisse wiederum konjunktiv mit dem entsprechenden Regelgewicht verknüpft wird:

$$\mu_{jl} = \alpha_l \wedge g_{jl} \quad . \quad (2.9)$$

Der unscharfe Ausgang einer Regel (2.4) ergibt sich dann abschließend aus der wiederum konjunktiven Verknüpfung des Erfülltheitsgrades  $\mu_{jl}$  der Regel und der Referenzmenge für den Ausgang:

$$\tilde{Y}_{jl}(y) = \mu_{jl} \wedge Y_j(y) \quad . \quad (2.10)$$

Die Bestimmung des Ausgangs einer Fuzzy–Regel nach der Prämissenauswertung wird *Aktivierung* genannt (Bertram u.a. 1994). Die  $d \times b$  Regeln eines Fuzzy–Systems sind im allgemeinen mit einem linguistischen „oder“ verknüpft. Nach Lee (1990) gibt es hierfür mehrere fuzzy–logische Realisierungsmöglichkeiten. In diesem Bericht wird der Fuzzy–ODER–Operator, also eine disjunktive Verknüpfung, gewählt. Dieser *Aggregation* genannte Vorgang kann noch in eine vertikale Aggregation für jede Referenzfuzzy–Menge des Ausgangs

$$\tilde{Y}_j = \bigvee_{l=1}^d \tilde{Y}_{jl} \quad (2.11)$$

und eine horizontale Aggregation über den Grundbereich des Ausgangs

$$\tilde{Y} = \bigvee_{j=1}^b \tilde{Y}_j \quad (2.12)$$

unterteilt werden.

Die bisher dargestellte und in den nächsten Abschnitten ausschließlich verwendete regelbasierte Inferenz unterscheidet sich von der sogenannten kompositionsbasierten Inferenz (Bertram 1991, Driankow u.a. 1993). Letztere verwendet bei diskreten Grundbereichen für die Ein- und Ausgänge eine sogenannte Relationalmatrix  $\mathbf{R}$  (zu unterscheiden von dem oben erwähnten Gewichtstensor  $\hat{\mathbf{R}}$ ), in der alle Informationen der Regelbasis, also über Fuzzy-Mengen für Ein- und Ausgänge (analog zu den Referenzfuzzy-Mengen), deren Verknüpfungen sowie verwendete Fuzzy-Operatoren, enthalten sind (Zadeh 1973, Lee 1990, Bertram 1991, Driankov u.a. 1993). Die Auswertung erfolgt über den Kompositionsoperator  $\circ$ :

$$\tilde{Y} = \tilde{X}^1 \circ \tilde{X}^2 \circ \dots \circ \tilde{X}^a \circ \mathbf{R} \quad . \quad (2.13)$$

Jedoch kann man die regelbasierte Inferenz unter bestimmten Voraussetzungen auf eine zu Gl. (2.13) analoge Schreibweise bringen. Dazu werden zunächst die Möglichkeiten der Eingänge  $\tilde{X}^i$  in Bezug auf die Referenzfuzzy-Mengen (Gl. 2.6) in Vektoren angeordnet:

$$\tilde{X}^i = [\Pi(\tilde{X}^i | X_1^i) \dots \Pi(\tilde{X}^i | X_{c_i}^i)]^T = [\tilde{x}_1^i, \tilde{x}_2^i, \dots, \tilde{x}_{c_i}^i]^T \quad (2.14)$$

Jeder *Möglichkeitsvektor* enthält also in umsortierter Form die Erfülltheitsgrade der Partialprämissen  $\alpha_{li}$  (Gl. 2.7). Er kann als Fuzzy-Menge auf dem Grundbereich  $\{1, 2, 3, \dots, c_i\}$  interpretiert werden. Damit ist auch das kartesische Produkt der Fuzzy-Logik für die Möglichkeitsvektoren definiert:

$$\begin{aligned} \mathbb{X}^a &= \tilde{X}^1 \otimes \tilde{X}^2 \otimes \dots \otimes \tilde{X}^a \\ \text{mit } \mu_{\mathbb{X}^a}(p_1, p_2, \dots, p_a) &= \mu_{\tilde{X}^1}(p_1) \wedge \mu_{\tilde{X}^2}(p_2) \wedge \dots \wedge \mu_{\tilde{X}^a}(p_a) = \\ &= \tilde{x}_{p_1}^1 \wedge \tilde{x}_{p_2}^2 \wedge \dots \wedge \tilde{x}_{p_a}^a \quad . \end{aligned} \quad (2.15)$$

Anschaulich bedeutet dies die Bildung aller möglichen Verknüpfungen zwischen den Partialprämissen. Das Ergebnis ist also ein Tensor  $\mathbb{X}^a$  der Ordnung  $a$ , in dem sich die Erfülltheitsgrade der Prämissen  $\alpha_l$  (vgl. Gl. 2.8) wiederfinden.

Die Komposition ist wie folgt definiert (Erweiterung von Böhme 1990):

### Definition 2.1

Auf den Grundbereichen  $\mathbb{D}_1, \mathbb{D}_2, \dots, \mathbb{D}_n$  mit den Elementen  $x_1, x_2, \dots, x_i$  seien die Fuzzy-Relationen

$$\begin{aligned} R_1 &= R_1(\mathbb{D}_1, \mathbb{D}_2, \dots, \mathbb{D}_l) \subset \mathbb{D}_1 \times \mathbb{D}_2 \times \dots \times \mathbb{D}_l \\ R_2 &= R_2(\mathbb{D}_k, \dots, \mathbb{D}_n) \subset \mathbb{D}_k \times \dots \times \mathbb{D}_n \quad , 1 \leq k \leq l \leq n \end{aligned} \quad (2.16)$$

erklärt. Dann ist die Komposition von  $R_1$  und  $R_2$  die Relation

$$R_2 \circ R_1 = R_2 \circ R_1(\mathbb{D}_1, \dots, \mathbb{D}_{k-1}, \mathbb{D}_{l+1}, \dots, \mathbb{D}_n) \subset \mathbb{D}_1 \times \dots \times \mathbb{D}_{k-1} \times \mathbb{D}_{l+1} \times \dots \times \mathbb{D}_n, \quad (2.17)$$

deren Komponenten durch die Zugehörigkeitsfunktion

$$\mu_{R_2 \circ R_1} = \bigvee_{x_k, \dots, x_l} (\mu_{R_1}(x_1, x_2, \dots, x_l) \wedge \mu_{R_2}(x_k, \dots, x_l)) \quad (2.18)$$

bestimmt sind. Der Disjunktionsoperator ist dabei kommutativ und assoziativ.  $\square$

Die regelbasierte Komposition kann nun als

$$\hat{Y} = (\hat{X}^1 \otimes \hat{X}^2 \otimes \dots \otimes \hat{X}^a) \circ \hat{\mathbf{R}} \quad (2.19)$$

geschrieben werden, wobei  $\hat{\mathbf{R}}$  der Gewichtstensor aus Gl. (2.5) mit der Ordnung  $a + 1$  ist. Das Ergebnis ist ein Vektor  $\hat{Y}$  mit  $b$  Elementen, der die Erfülltheitsgrade für die Referenzfuzzy-Menge des Ausgangs enthält. Die Ausgangsfuzzy-Menge ergibt sich demnach zu

$$\hat{Y} = \bigvee_{j=1}^b (\hat{y}_j \wedge Y_j(y)) \quad . \quad (2.20)$$

Im Vergleich zu Gln. (2.10), (2.11) und (2.12) fällt auf, daß die vertikale Aggregation durch die Disjunktion bereits im Rahmen der Komposition abgearbeitet wird. Die Gleichheit der Ergebnisse aus Gl. (2.12),

$$\hat{Y} = \bigvee_{j=1}^b \left( \bigvee_{l=1}^d (\mu_{jl} \wedge Y_j(y)) \right) \quad , \quad (2.21)$$

und Gl. (2.20) ist gewährleistet, wenn die verwendete Konjunktion distributiv über der Disjunktion ist, d. h. wenn

$$(A \wedge B) \vee (A \wedge C) = A \wedge (B \vee C) \quad (2.22)$$

gilt. Das ist für viele der verwendeten Disjunktionen und Konjunktionen der Fall, z. B. max – min, max – prod, sum – prod. Selbstverständlich kann auf die Distributivität verzichtet werden, wenn die eingangs beschriebene Abarbeitung der Regeln entsprechend geändert würde.

Statt der Gl. (2.19) findet man in der Literatur häufig eine zu Gl. (2.13) analoge Form:

$$\hat{Y} = \hat{X}^1 \circ \hat{X}^2 \circ \dots \circ \hat{X}^a \circ \hat{\mathbf{R}} \quad . \quad (2.23)$$

Für eine einfache Betrachtung werden zunächst beide Gleichungen in Form ihrer Zugehörigkeitsgrade geschrieben. Aus Gl. (2.19) ergibt sich

$$\mu_{\hat{Y}} = \bigvee_{p_1, p_2, \dots, p_a} \left( x_{p_1}^1 \wedge x_{p_2}^2 \wedge \dots \wedge x_{p_a}^a \wedge \mu_{\hat{\mathbf{R}}}(p_1, p_2, \dots, p_a) \right) \quad , \quad (2.24)$$

und aus Gl. (2.23)

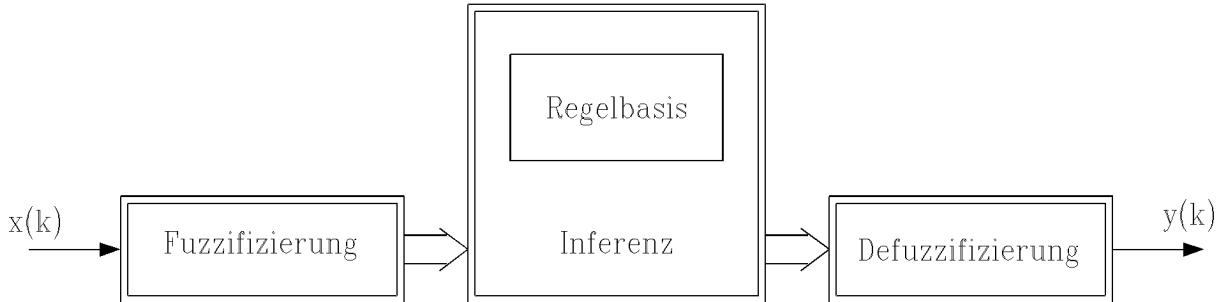
$$\mu_{\hat{Y}} = \bigvee_{x_1} \left( x_{p_1}^1 \wedge \bigvee_{x_2} \left( x_{p_2}^2 \wedge \dots \wedge \bigvee_{x_a} \left( x_{p_a}^a \wedge \mu_R(p_1, p_2, \dots, p_a) \right) \dots \right) \right). \quad (2.25)$$

Die beiden Ausdrücke sind wiederum identisch, wenn die verwendete Konjunktion distributiv über der Disjunktion ist (Gl. 2.22). Also lässt sich unter dieser schwachen Bedingung die regelbasierte Inferenz als

$$\hat{Y} = X^1 \circ \hat{X}^2 \circ X^3 \circ \dots \circ \hat{X}^a \circ \hat{R} \quad (2.26)$$

darstellen. Dabei ist, wenn zusätzlich die Kommutativität des Disjunktionsoperators vorausgesetzt wird, die Reihenfolge der Vektoren  $\hat{X}^i$  beliebig.

Bisher wurde lediglich die allgemeine Modellstruktur bezüglich der Regelbasis, also des „Fuzzy–Teils“ eines Fuzzy–Systems beschrieben. Im Bereich der Fuzzy–Control wird jedoch normalerweise mit scharfen (= nicht–fuzzy) Ein- und Ausgangsgrößen gerechnet. Die sich daraus ergebende Struktur für das Gesamtsystem ist in Bild 2.1 dargestellt. Die Fuzzifizierung wandelt eine scharfe Eingangsgröße in eine Fuzzy–Menge. Üblich sind



**Bild 2.1:** Relationaler Fuzzy–Regler

zwei Arten der Fuzzifizierung (Lee 1990). Bei der Fuzzifizierung mit Fuzzy–Einermengen (Singletons) wird aus dem scharfen Eingangswert  $x_0$  eine Fuzzy–Einermenge (Singleton) mit dem Träger  $x_0$ . Bei der zweiten Methode wird dem scharfen Eingangswert eine andere, etwa dreieckförmige, Fuzzy–Menge zugeordnet. Dies erscheint insbesondere bei verrauschten Meßwerten sinnvoll.

In vielen anwendungsbezogenen Veröffentlichungen wird die Fuzzifizierung weiter gefaßt. Dabei geht man von über dem Definitionsbereich des Eingangs definierten Referenzfuzzy–Mengen aus und faßt die Schritte „Fuzzifizierung mit Fuzzy–Einermengen (Singletons)“ und „Bestimmung des Möglichkeitsvektors“ (vgl. Gl. 2.14) zusammen. Dieses Vorgehen soll *erweiterte Fuzzifizierung* genannt werden.

Um das Ergebnis der Inferenz, die Fuzzy–Menge  $\hat{Y}$ , in einen scharfen Wert  $\hat{y}$  zurückzuwandeln, erfolgt die sogenannte Defuzzifizierung. Hier gibt es eine Vielzahl von Methoden, von denen nur zwei kurz dargestellt werden. Bei der erweiterten Schwerpunktmetode wird

der Schwerpunkt der Fläche unter der Zugehörigkeitsfunktion  $\mu_{\tilde{Y}}(y)$  von  $\tilde{Y}$  als Ausgangswert bestimmt:

$$\hat{y} = \frac{\int_{\mathbb{W}} \mu_{\tilde{Y}}(y) y \, dy}{\int_{\mathbb{W}} \mu_{\tilde{Y}}(y) \, dy} . \quad (2.27)$$

Um auch die Grenzen des Wertebereichs  $\mathbb{W}$  von  $y$  erreichen zu können, werden die in den Bereichsgrenzen liegenden Zugehörigkeitsfunktionen der Referenzfuzzy-Mengen  $Y_k$  symmetrisch erweitert (sog. erweiterte Schwerpunktmetode).

Die Höhenmethode (Driankov u. a. 1993) verzichtet auf die horizontale Aggregation (Gl. 2.12). Stattdessen werden die  $y$ -Koordinaten  $y_{j_{max}}$  der Maximalwerte der aus der vertikalen Aggregation stammenden Fuzzy-Mengen  $\tilde{Y}_j$  (vgl. Gl. 2.11) mit ihren Maximalwerten  $\mu_{j_{max}}$  gewichtet gemittelt:

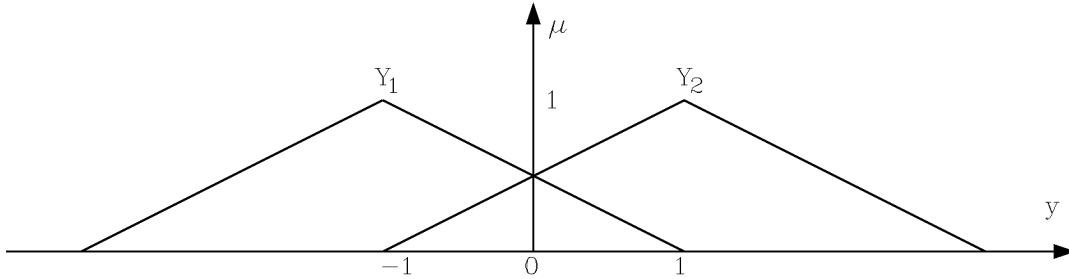
$$\hat{y} = \frac{\sum_{j=1}^b y_{j_{max}} \mu_{j_{max}}}{\sum_{j=1}^b \mu_{j_{max}}} . \quad (2.28)$$

Bisher wurden alle Gleichungen für allgemeine Operatoren der Fuzzy-Logik (Bertram 1991, Lee 1990) beschrieben. Für die in diesem Bericht betrachteten Fuzzy-Systeme finden jedoch im weiteren ausschließlich die folgenden Operatoren Verwendung:

- Fuzzifizierung über Fuzzy-Einermengen (Singletons).
- Konjunktion bei der Prämisseauswertung (Gl. 2.8): Produkt.
- Konjunktion bei der Verknüpfung von Prämisse und Regelgewicht (Gl. 2.9): Produkt.
- Konjunktion bei der Verknüpfung von Erfülltheitsgrad und Referenzfuzzy-Menge (Gl. 2.10): Produkt.
- Disjunktion bei der vertikalen Aggregation: Summe.
- Disjunktion bei der horizontalen Aggregation: Summe.
- Defuzzifizierung: Erweiterte Schwerpunktmetode.

Es ist üblich, alle oben angeführten Konjunktionen bzw. Disjunktionen gleich zu wählen und zum Kompositionsooperator zusammenzufassen. In diesem Bericht wird also die „sum-prod-Komposition“ (Mizumoto 1987, Larsen 1980) verwendet. Für diese Komposition (und gleichgroße Flächen unter den Referenzfuzzy-Mengen des Ausgangs) entsprechen sich außerdem die beiden oben angeführten Defuzzifizierungsmethoden.

Alle Ein- und Ausgangsgrößen sind auf das Intervall  $[-1; 1]$  normiert. Als Referenzfuzzy-Mengen werden Mengen mit gleichverteilten, gleichschenkligen Dreiecken als Zugehörigkeitsfunktionen verwendet, die ein sogenanntes *Fuzzy-Informationssystem* (Meyer-Gramann und Jüngst 1993) bilden. Ihre Anzahl pro Eingang ist variabel. Für den Ausgang werden nur zwei Mengen verwendet (Bild 2.2). Bertram, Küpper und Schwarz (1993) zei-



**Bild 2.2:** Referenzfuzzy-Mengen für den Ausgang

gen, daß mit diesen Mengen bei der sum–prod–Komposition beliebige andere Mengen ohne Einschränkung ersetzt werden können. Die gewählten Operatoren und Fuzzy–Mengen führen zu einem geschlossenen Ausdruck für den Ausgang  $\hat{y}$  des Fuzzy–Modells:

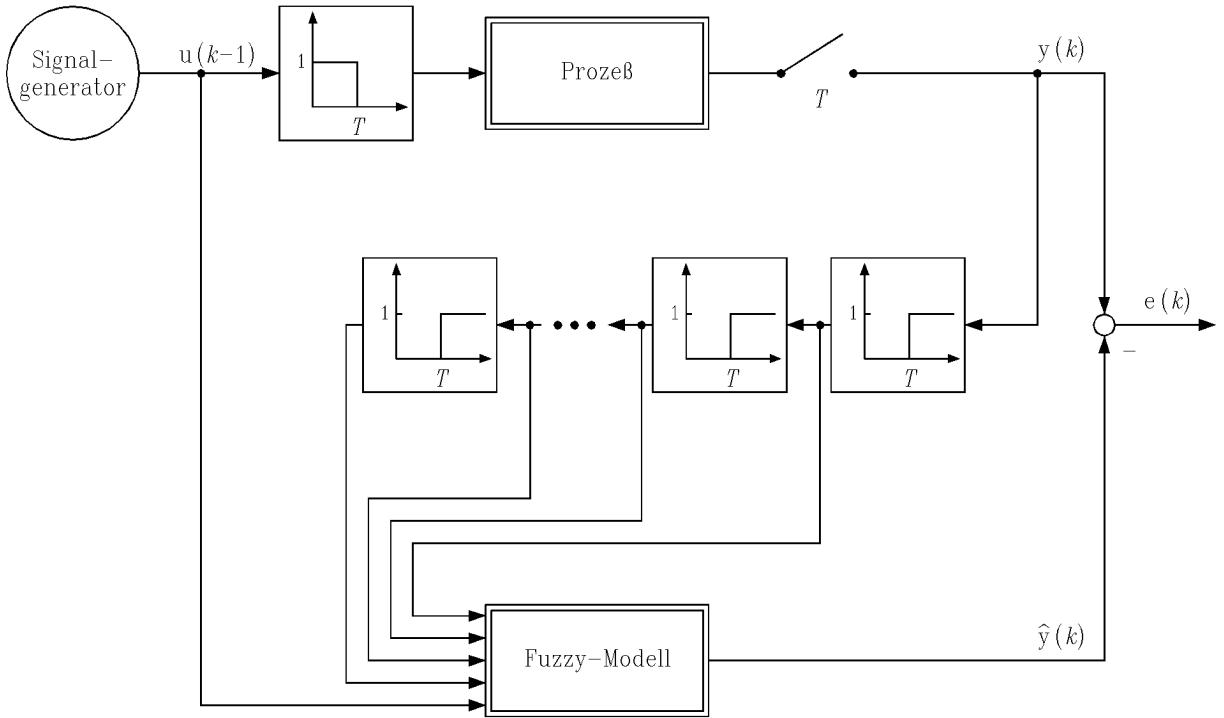
$$\hat{y} = \frac{\sum_{l=1}^d \alpha_l (-g_{1l} + g_{2l})}{\sum_{l=1}^d \alpha_l (g_{1l} + g_{2l})} \quad (2.29)$$

Die Herleitung findet sich in (Küpper 1993b).

In diesem Bericht findet, wie auch in den meisten anwendungsbezogenen Veröffentlichungen, ausschließlich die regelbasierte Inferenz Anwendung. Mit diesen Einschränkungen kann deshalb die Notation im folgenden etwas vereinfacht werden: Der Tensor der Möglichkeitsgrade der Regeln wird nur als  $\mathbf{R}$  bezeichnet (statt  $\mathbf{\hat{R}}$ ), die Vektoren der Möglichkeitsgrade als  $X^i$  bzw.  $Y$  (statt  $\mathbf{\hat{X}}^i$  und  $\mathbf{\hat{Y}}$ ). So wird die Lesbarkeit der Formeln deutlich verbessert, ohne daß auf Exaktheit in der Darstellung verzichtet werden muß.

## 2.2 Dynamische Modelle

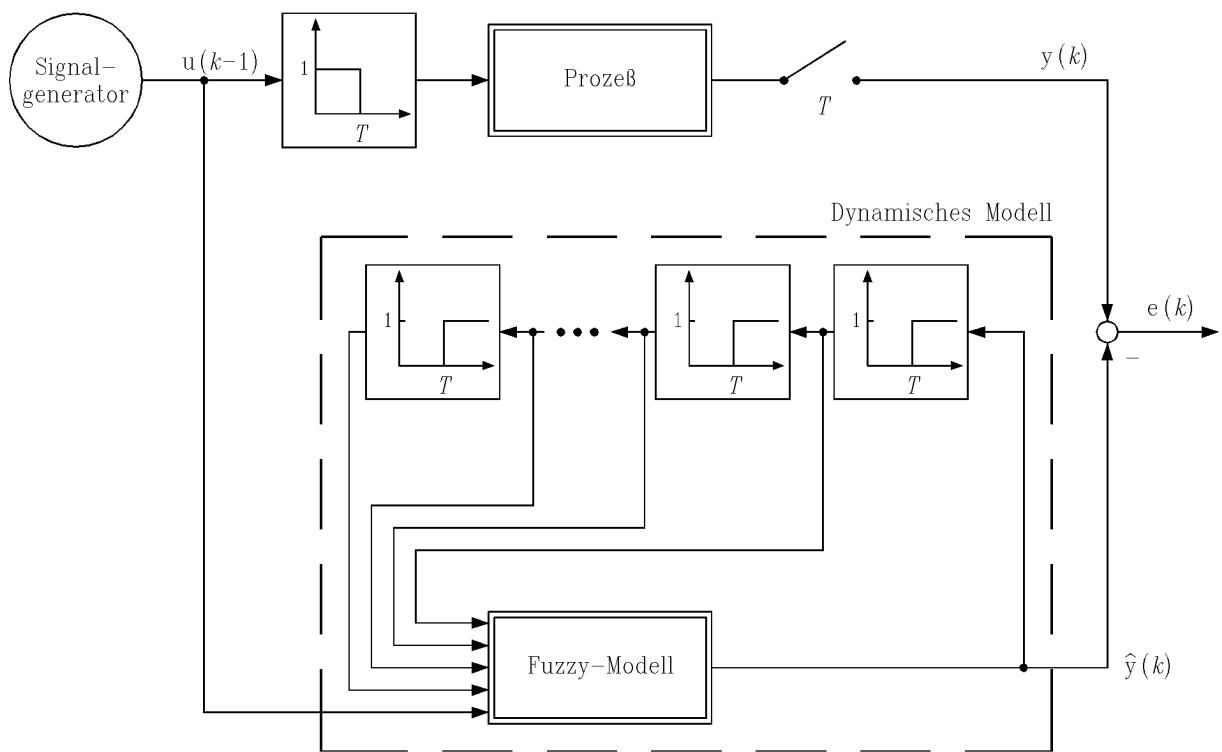
Fuzzy–Systeme können als statische, nichtlineare Kennfelder interpretiert werden. Sie stellen eine Abbildung der Eingangsgrößen auf die Ausgangsgrößen dar. Erst durch die Rückführung des Ausgangs entsteht ein dynamisches System. Da Fuzzy–Systeme in den allermeisten Fällen auf Digitalrechnern abgearbeitet werden, beschränkt sich dieser Bericht auf zeitdiskrete Betrachtungen von Fuzzy–Modellen. In diesem Fall sind die Eingänge des Fuzzy–Modells zum einen die Eingangsgrößen des zu modellierenden Prozesses und zum anderen dessen abgetastete und zeitverzögerte Ausgangsgrößen (Bild 2.3). Damit erhält das Fuzzy–System Informationen über die Dynamik des Prozesses und kann diese abbilden. Es ist selbstverständlich auch möglich, zeitliche Ableitungen als Modelleingänge



**Bild 2.3:** Fuzzy-Modell, Einschrittprädiktion

zu verwenden. Davon soll hier allerdings kein Gebrauch gemacht werden. Der Fehler  $e(k)$  zwischen dem abgetasteten Prozeßausgang  $y(k)$  und dem des Fuzzy-Modells  $\hat{y}(k)$  wird, wie im nächsten Abschnitt beschrieben, als Lernsignal für das Fuzzy-Modell benutzt.

Werden die Totzeitglieder wie in Bild 2.3 aus dem Prozeßausgang gespeist, soll von einer *Einschrittprädiktion* gesprochen werden. Es ist in diesem Fall zur Modellauswertung immer nötig, auf den realen Prozeß — wenn auch um einen Abtastschritt zeitverzögert — zugreifen zu können. Erst durch den Ersatz des Prozeßausgangs durch den Modelausgang als Eingang für die Totzeitglieder ist ein „Stand-alone-Betrieb“ des Fuzzy-Modells möglich (Bild 2.4). Es liegt dann ein Fuzzy-System vor, das durch eine externe Rückführung dynamisiert ist. Eine solche Auswertung des Fuzzy-Modells soll *rekursiv* genannt werden. Sie erlaubt erst die Verwendung des Modells für Systemsimulationen, den Einsatz als Mehrschrittprädiktor und Simulationen mit Hardware-in-the-loop, räumlich und zeitlich unabhängig vom modellierten Prozeß.



**Bild 2.4:** Fuzzy-Modell, rekursive Auswertung

### 3 Identifikationsalgorithmus

Sowohl die klassische wie auch die Fuzzy-Modellbildung besteht aus drei grundsätzlichen Schritten (Unbehauen 1988, Sugeno und Yasukawa 1993): Zunächst muß die Modellstruktur gefunden und festgelegt werden. Darauf folgt die Bestimmung der Modellparameter, die sogenannte Parameteridentifikation. Die Verifikation des gefundenen Modells mit Testdaten, die sich von den zur Parameteridentifikation verwendeten Trainings- oder Identifikationsdaten unterscheiden sollten, schließt die Modellbildung ab.

Ein Punkt bei der Strukturidentifikation für Fuzzy-Modelle besteht in der Auswahl geeigneter Operatoren. Dafür gibt es noch keine anerkannten oder allgemein gültigen Regeln. Jedoch haben sich die in Abschnitt 2 gewählten Operatoren sowohl bezüglich ihrer Implementierbarkeit als auch bezüglich ihrer Leistungsfähigkeit in früheren Versuchen bewährt (Bertram u.a. 1993, Bertram und Schwarz 1993, Küpper 1993a,b). Deshalb wurde auf notwendigerweise heuristische Versuche zu ihrer optimalen Bestimmung verzichtet. Ein weiterer wichtiger Aspekt bei der Strukturidentifikation liegt in der Wahl der Modellein- und -ausgänge sowie der Wahl der Referenzfuzzy-Mengen. Eine hervorragende Übersicht zu diesem Thema geben Sugeno und Yasukawa (1993). Bei dem in diesem Bericht untersuchten Modell ergeben sich Ein- und Ausgänge sehr schnell. Die wenigen notwendigen Versuche sind im Abschnitt 4 beschrieben. Als Zugehörigkeitsfunktionen der Referenzfuzzy-Mengen wurden gleichverteilte, gleichschenklige Dreiecke mit einem Überschneidungsgrad von 0,5 gewählt.

Bevor die Parameteridentifikation diskutiert werden kann, müssen selbstverständlich die zu identifizierenden Parameter ausgewählt werden. Bei den relationalen Fuzzy-Systemen bieten sich die Elemente der Relationalmatrix an. Dieser Ansatz wird hier verfolgt (Küpper 1993a). Eine Identifikation der Schwerpunkte der Ausgangsfuzzy-Mengen erscheint schon wegen der Begrenzung auf zwei Referenzmengen für den Ausgang nicht sinnvoll. Wie jedoch bereits angesprochen, ist eine Rückführung solcher Modelle auf das hier verwendete jederzeit möglich. Eine Anpassung der Fuzzy-Mengen der Eingänge (Form, Lage) ist ebenfalls möglich (Sugeno und Yasukawa 1993). Erste Untersuchungen im Zusammenhang mit relationalen Fuzzy-Modellen finden sich in Suprijadi (1994).

Die erste Identifikation eines Gewichtstensors wird von Pedrycz (1984a,b) beschrieben. Sie basiert auf der Auflösung der relationalen Fuzzy-Gleichung

$$Y = X \circ \mathbf{R} \quad (3.1)$$

nach  $\mathbf{R}$ . Dies geschieht für jedes Meßtupel; die resultierenden Teilrelationalmatrizen werden anschließend überlagert. Dieses Verfahren wurde von Bertram und Schwarz (1993), Küpper (1993a,b) sowie Bertram, Küpper und Schwarz (1993) verbessert und angewendet. Jedoch liefert es keine sehr genauen Ergebnisse. Das ist insbesondere darauf zurückzuführen, daß das Verfahren nicht iterativ angewendet wird. Zwar liefern die Teilrelationalmatrizen für ihre jeweiligen Meßtupel exakte Werte (Bertram, Küpper und Schwarz

1993), jedoch nimmt die Genauigkeit durch ihre Überlagerung rapide ab. Eine Optimierung wie bei herkömmlichen Identifikationsmethoden findet nicht statt. Die erste Idee zur numerischen Optimierung findet sich schon bei Pedrycz (1983). Xu und Lu (1987) stellen ein selbstlernendes Modell vor, das eine gewisse Optimierung durch wiederholtes Lernen der Meßwerte erlaubt (vgl. auch Postlethwaite 1991, Angenendt 1993). Das Verfahren beruht auf einem heuristischen Lernansatz und liefert bereits bessere Ergebnisse. Weitere Verbesserungen lassen sich durch nichtlineare Optimierungsverfahren im off-line-Betrieb erreichen (Küpper 1993a,b). Der große zeitliche Aufwand und die mangelnde Eignung zur on-line-Anwendung dieser Verfahren legt die Entwicklung von on-line-fähigen lernenden oder adaptiven Modellen nahe. Verschiedene Ansätze finden sich bei Liu (1993), Pfeiffer (1993), Wang (1994), Wang und Mendel (1993), Brown und Harris (1991), Moore und Harris (1992) und Glorenne (1991).

In diesem Bericht wird der Gewichtstensor der relationalen Fuzzy-Modelle ausgehend von Gl. (2.29) mit dem Verfahren der stochastischen Approximation bestimmt, das 1951 von Robins und Monro (Ljung und Söderström 1983) eingeführt wurde. Die folgende Beschreibung lehnt sich eng an die sehr gute Darstellung von Ljung und Söderström (1983) an.

Gegeben sei eine Funktion  $Q(\mathbf{x}(k), \boldsymbol{\Theta})$ , wobei  $\boldsymbol{\Theta}$  die Parameter der Funktion sind und  $\mathbf{x}(k)$  Zufallszahlen, deren Verteilungsdichtefunktion unbekannt ist.  $Q(k)$  sei streng monoton fallend. Ziel ist die Lösung der Gleichung

$$E[Q(\mathbf{x}(k), \boldsymbol{\Theta})] = f(\mathbf{x}) = 0. \quad (3.2)$$

wobei  $E$  der Erwartungswert von  $Q$  über  $\mathbf{x}(k)$  ist. Die Lösung von (3.2) kann dann mit

$$\boldsymbol{\Theta}(k+1) = \boldsymbol{\Theta}(k) + \gamma(k) Q(\mathbf{x}(k), \boldsymbol{\Theta}(k-1)) \quad (3.3)$$

rekursiv berechnet werden. Die Folge von Lernfaktormatrizen  $\gamma(k)$  wird im folgenden zu einem Skalar  $\gamma(k)$  vereinfacht. Die Konvergenz des Verfahrens ist für

$$\gamma(k) \geq 0, \quad \sum_{k=1}^{\infty} \gamma(k) = \infty, \quad \sum_{k=1}^{\infty} \gamma^2(k) < \infty \quad (3.4)$$

nachgewiesen (Tsyplkin 1971). Genauere Konvergenzuntersuchungen, insbesondere bei schwächeren Annahmen, finden sich bei Ljung (1974) sowie Ljung und Söderström (1983).

Um eine Funktion  $f$  zu minimieren, wird  $Q$  in Gl. (3.2) durch ihren negativen Gradienten ersetzt:

$$Q(\mathbf{x}(k), \boldsymbol{\Theta}) = -\frac{\partial f(\mathbf{x}(k), \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}. \quad (3.5)$$

Es ist zu beachten, daß mit diesem Verfahren nur lokale Minima zu finden sind.

Im Fall der Parameteridentifikation ist der Ausgangspunkt der Überlegungen der Fehler zwischen dem Ausgang des Fuzzy-Modells  $\hat{y}$  und dem gemessenen Ausgang  $y$  des zu identifizierenden Systems:

$$e(k) = (y(k) - \hat{y}(\mathbf{x}(k), \boldsymbol{\Theta})). \quad (3.6)$$

Dabei enthält der Vektor  $\mathbf{x}(k)$  alle Eingänge des Fuzzy–Modells zum Zeitpunkt  $k$  und der Vektor  $\Theta$  alle Elemente der Relationalmatrix. Optimierungsziel ist es, die Varianz  $V$  des Fehlers über den Parametervektor  $\Theta$  zu minimieren:

$$\min_{\Theta} V(\epsilon(\mathbf{x}, \Theta)) = \min_{\Theta} \left( \frac{1}{2} E[(y - \hat{y})^2] \right) \quad . \quad (3.7)$$

Dabei ist  $E[\cdot]$  der Erwartungswert. Der Faktor  $1/2$  wird zur Rechenerleichterung eingeführt. Gemäß Gl. (3.5) und (3.2) ergibt sich

$$-\frac{\partial}{\partial \Theta} \frac{1}{2} E[(y - \hat{y})^2] = E[-\frac{\partial}{\partial \Theta} \frac{1}{2} (y - \hat{y})^2] = 0 \quad , \quad (3.8)$$

bei der Erwartungswert und Differentiation vertauscht wurden.

Die rekursive Gleichung zur Bestimmung des optimalen Parametervektors  $\Theta$  lautet

$$\begin{aligned} \Theta(k+1) &= \Theta(k) + \gamma(k) \left( -\frac{\partial}{\partial \Theta} \frac{1}{2} (y(k) - \hat{y}(k))^2 \right) \\ &= \Theta(k) + \gamma(k) (\hat{y}(k) - y(k)) \frac{\partial \hat{y}(k)}{\partial \Theta} \quad . \end{aligned} \quad (3.9)$$

Dieser Algorithmus wird auch stochastisches Gradientenverfahren genannt und findet im Bereich der Neuronalen Netze als Least–Mean–Squares–Verfahren (LMS) häufig Anwendung (Ljung und Söderström 1983).

Zur Berechnung der partiellen Ableitungen in Gl. (3.9) wird  $\hat{y}$  gemäß Gl. (2.29) getrennt nach  $g_{1p}$  und  $g_{2p}$ ,  $p \in [1, 2, \dots, d]$ , differenziert:

$$\frac{\partial \hat{y}}{\partial g_{1p}} = -2 \alpha_p \frac{\sum_{l=1}^d \alpha_l g_{2l}}{\left( \sum_{l=1}^d \alpha_l (g_{1l} + g_{2l}) \right)^2} \quad , \quad (3.10)$$

$$\frac{\partial \hat{y}}{\partial g_{2p}} = 2 \alpha_p \frac{\sum_{l=1}^d \alpha_l g_{2l}}{\left( \sum_{l=1}^d \alpha_l (g_{1l} + g_{2l}) \right)^2} \quad . \quad (3.11)$$

Mit diesen Gleichungen können die Änderungen von  $\Theta(k)$  nun berechnet werden. Der rechnerische Aufwand bei der Implementierung ist relativ gering, weil die einzelnen Summen in Gln. (3.10) und (3.11) bereits für die Auswertung von  $\hat{y}$  (Gl. 2.29) zu berechnen sind. Schließlich muß die Frage nach einer geeigneten Gewichtsfolge  $\gamma(k)$  beantwortet werden. Eigene Experimente ergaben, daß ein geeignet gewähltes, konstantes  $\gamma$  in fast allen Fällen die Konvergenz des Fehlers garantiert. Deshalb wurde zunächst auf eine zeitvariante Folge, wie etwa  $\gamma(k) = \frac{1}{k}$ , verzichtet.

Letzter, aber nicht unwichtigster Aspekt der Modellbildung ist die Verifikation des gefundenen Modells. An erster Stelle steht sicherlich der Vergleich zwischen den Trainingsdaten  $y(k)$  und dem Ausgang des Modells  $\hat{y}(k)$ . Wird dabei eine ausreichende Übereinstimmung erzielt, müssen außerdem andere Signale, im folgenden *Testsignale* genannt, auf das reale

System und das Modell gegeben werden. Dies ist insbesondere bei Fuzzy-Modellen wichtig, da hier wegen der teilweise sehr hohen Parameterzahl (sie steigt exponentiell mit der Zahl der Eingänge) die Gefahr der Überparametrisierung besteht. Das bedeutet, daß das Modell nicht die Modellstruktur, sondern vielmehr das Eingangssignal modellieren würde. Ein geeignetes Fehlermaß für den Vergleich zwischen Soll- und Ist-Verlauf stellt der mittlere quadratische Fehler dar:

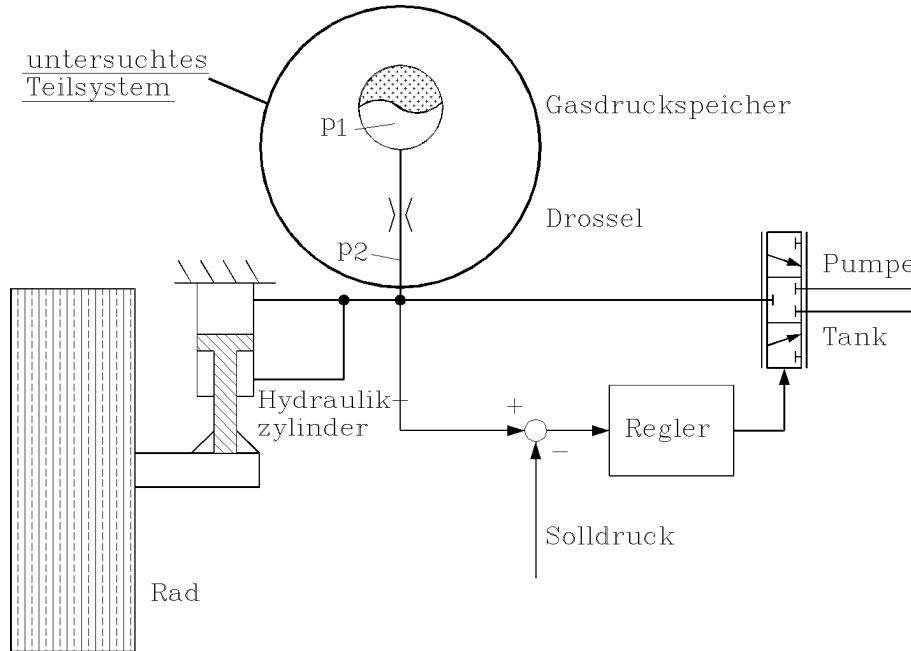
$$q_m = \frac{1}{N} \sum_{k=1}^N (y(k) - \hat{y}(k))^2 . \quad (3.12)$$

Es sei jedoch angemerkt, daß die Fehler, die den Fortschritt eines Lernvorgangs dokumentieren, nicht während des Lernens bestimmt werden. Das heißt, daß zunächst ein Lernschritt mit allen Trainingsdaten durchgeführt und dann, ohne Lernen, also mit  $\gamma(k) = 0$ , der Ausgang des Modells für die Trainingsdaten berechnet und der Fehler bestimmt wird.

## 4 Untersuchtes System

### 4.1 Beschreibung

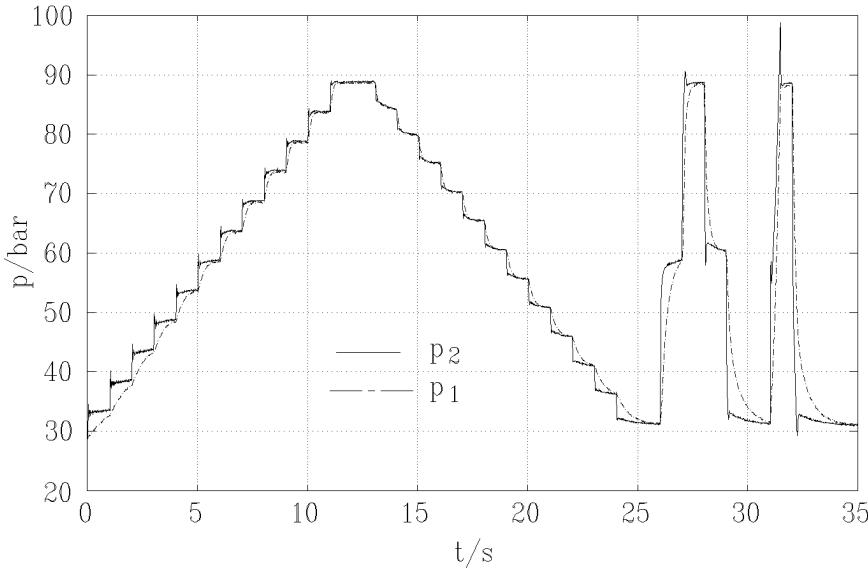
Das untersuchte hydraulische System ist Teil eines Prüfstandes, der aus einigen hydraulischen Komponenten einer aktiven hydropneumatischen Federung eines PKW besteht. Es umfaßt die Drossel und die Speichereinheit (Bild 4.1). Als einziges technisches Da-



**Bild 4.1:** Aktive hydropneumatische Federung

tum ist der Vorspanndruck des Stickstoffs in dem Speicher bekannt; er beträgt 17 bar. Als Meßgrößen stehen der Druck  $p_1$  im Speicher und der Systemdruck  $p_2$  zur Verfügung. Die Auflösung der Aufnehmer beträgt 2047 bit bei einem Meßbereich von 0 – 200 bar. Das ergibt eine Auflösung von  $0,0977 \frac{\text{bar}}{\text{bit}}$ . Durch Störungen im Meßsystem schwankt der Meßwert bei konstantem Druck um  $\pm 1 \text{ bit} \approx \pm 0,1 \text{ bar}$ . Der Systemdruck liegt wie der Speicherdruck zwischen 30 bar und 90 bar.

Auf eine genaue Modellierung des Systems wurde verzichtet, da dieses Wissen nicht Voraussetzung für die Modellierung als Fuzzy-System ist. Das Zeitverhalten des Systems soll anhand von Bild 4.2 betrachtet werden. Deutlich sind die unterschiedlichen Zeitkonstanten des Systems für Sprünge bei niedrigem und bei hohem Druckniveau zu erkennen. Auffällig ist auch, daß bei einem Sprung auf ein höheres Druckniveau ein anderer Verlauf als beim Rücksprung auftritt. Schließlich sei noch auf die physikalisch leicht erklärbare Tatsache hingewiesen, daß sich nach einer gewissen Zeit Druckgleichheit zwischen Speicher und System einstellt. Kleinere Abweichungen sind auf nicht ausgeglichene Offsets der Druckaufnehmer zurückzuführen.



**Bild 4.2:** Verlauf von System- und Speicherdruck (Trainingssignal)

## 4.2 Testdaten

Es ist eine charakteristische Eigenschaft des verwendeten Fuzzy-Modells, daß durch die gewählten dreieckigen Zugehörigkeitsfunktionen die Regeln im Raum der Eingangsgrößen nur lokal wirken. Liegen also während der Identifikation in einzelnen Unterräumen keine Daten vor, können dort später auch keine sinnvollen Ausgänge generiert werden. Das ist ein entscheidender Unterschied etwa zu linearen Modellen. Daher muß das Trainingssignal möglichst den gesamten Eingangsraum abdecken. Der für die Identifikation verwendete Verlauf des Systemdrucks wurde bereits in Bild 4.2 dargestellt. Durch die kleinen, mittleren und großen Sprünge auf allen Druckniveaus ist ein weiter Bereich des Eingangsraums abgedeckt. Ein ähnlicher Signalverlauf wird von Bertram, Küpper und Schwarz (1993) verwendet.

Zur Verifikation der gefundenen Modelle standen verschiedene Testsignale zur Verfügung. Das war zum einen ein Signal, das dem Trainingssignal entsprach, aber zu einem anderen Zeitpunkt gemessen wurde (Signal 1), ferner drei sinusförmige Signale mit unterschiedlichen Frequenzen (Signal 2 mit 0,75 Hz, Signal 3 mit 1,5 Hz und Signal 4 mit 2 Hz). Schließlich fand noch eine Folge von Sprüngen Verwendung, die im Trainingssignal nicht enthalten waren (Signal 5 und Signal 6), wobei das Signal 6 nur halb so große Haltezeiten wie Signal 5 besaß. Alle Signale sind im Anhang A dargestellt.

Die Abtastzeit beträgt bei allen Signalen  $T = 15$  ms. Diese waren zunächst mit  $T = 5$  ms aufgezeichnet worden, vor allem wegen der für die Regelung des Gesamtsystems notwendigen kurzen Abtastzeiten. Jedoch zeigte eine Abschätzung der optimalen Abtastzeit anhand der Sprungantworten im Trainingssignal (Reuter 1993b), daß die Verwendung einer dreifach größeren Abtastzeit ohne signifikanten Informationsverlust möglich ist.

## 5 Lösungsansätze

### 5.1 Festlegungen

Um die möglichen Variationen bei der Strukturfestlegung für die verschiedenen Modelle zu begrenzen, seien zunächst Form und Anzahl der Referenzfuzzy-Mengen für die Eingänge und die Normierungen festgelegt. Für jeden Eingang gibt es auf dem normierten Grundbereich  $[-1; 1]$  je fünf Fuzzy-Mengen mit dreieckigen Zugehörigkeitsfunktionen (Bild 5.1). Um eine sichere Erfassung aller Drücke zu gewährleisten, wird der Bereich von 20 bar bis

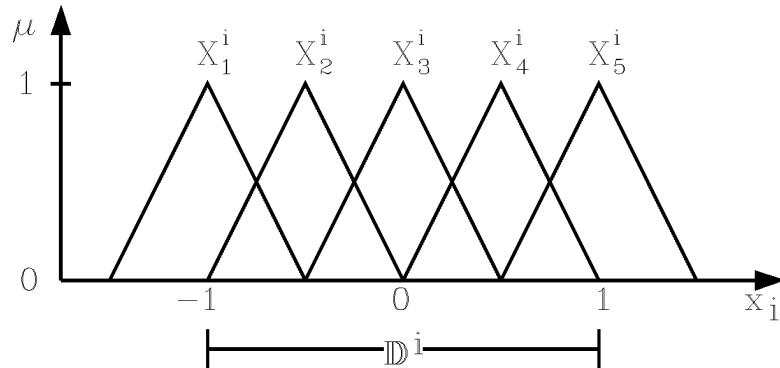


Bild 5.1: Zugehörigkeitsfunktionen für die Referenzfuzzy-Mengen des Eingangs

100 bar erfaßt. Die Normierung erfolgt mit

$$p_n = \frac{1}{40\text{bar}} (p - 60\text{bar}) \quad . \quad (5.1)$$

Werden Druckdifferenzen erfaßt, erfolgt die Normierung über

$$\Delta p_n = \frac{1}{30\text{bar}} \Delta p \quad . \quad (5.2)$$

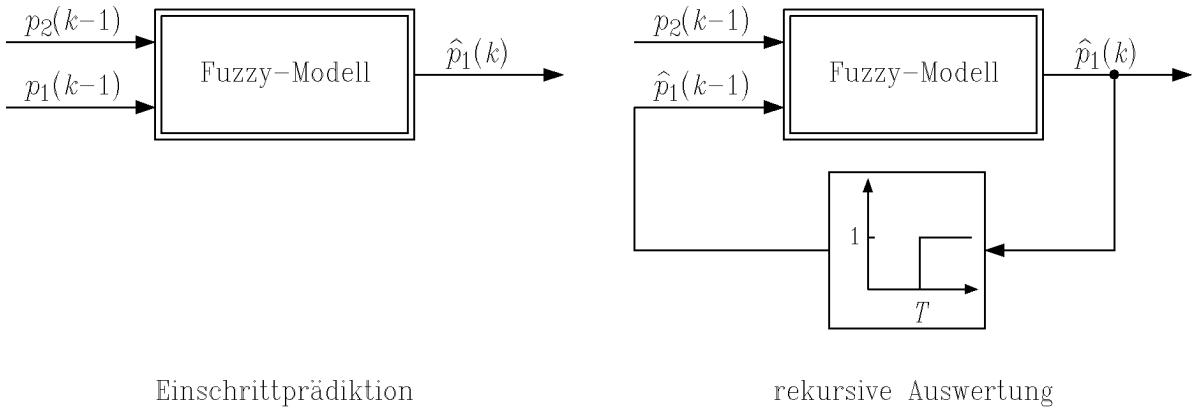
Es sei besonders darauf hingewiesen, daß bei der rekursiven Auswertung nicht direkt die prädizierte Fuzzy-Menge  $\tilde{Y}$  zurückgeführt wird, sondern diese zunächst defuzzifiziert und dann wieder fuzzifiziert wird. Das hat zum einen den Vorteil der leichteren Implementierung des rekursiven Modells, zum anderen wird die Umsetzung von den zwei Fuzzy-Mengen des Ausgangs (dargestellt durch den Vektor  $\tilde{Y}$ ) in die fünf Mengen des Eingangs ( $\tilde{X}$ ) erleichtert. Ebenso vereinfacht sich die bei einigen der untersuchten Modellstrukturen notwendige Summation der Signale. Schließlich wird eine immer stärkere Fuzzifizierung zweier Signale, die durch mehrfache Rückführung bis zur Wertlosigkeit führen kann, vermieden. Alle Werte des Relationaltensors waren mit dem Wert 0,5 initialisiert, der Lernfaktor  $\gamma(k)$  ist in allen Fällen konstant  $\gamma = 0,05$ .

### 5.2 Modell 1

Bei einer ersten Betrachtung des Trainingssignals erscheint das Systemverhalten näherungsweise als das eines  $P_{T1}$ -Systems mit einer vom Arbeitspunkt abhängigen Verstärkung. Dementsprechend erscheint es sinnvoll, die Eingangs- und Ausgangsvariablen analog

zu denen eines zeitdiskreten  $P_{T1}$ -Systems zu wählen:

Als Eingangsvariablen werden  $p_1(k-1)$  und  $p_2(k-1)$  gewählt, als Ausgangsvariable  $p_1(k)$ . Die Blockschaltbilder für eine Einschrittprädiktion und eine rekursive Auswertung sind in Bild 5.2 dargestellt. Zunächst soll überprüft werden, ob die gewählte zeitliche Differenz von  $k = 1$  zwischen den Eingangsvariablen und dem Ausgang wirklich die optimale Lösung darstellt. Zu diesem Zweck wurde die Modellstruktur zu



**Bild 5.2:** Struktur von Modell 1

renz von  $k = 1$  zwischen den Eingangsvariablen und dem Ausgang wirklich die optimale Lösung darstellt. Zu diesem Zweck wurde die Modellstruktur zu

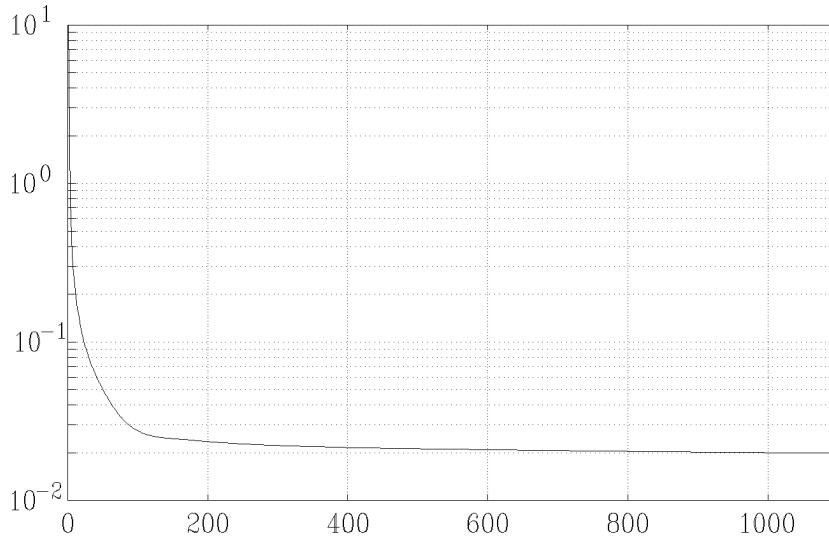
$$\hat{p}_1(k) = f(p_2(k - \tau_2), p_1(k - \tau_1)) \quad (5.3)$$

geändert. Die Berechnung des Fehlers nach 100 Lernschritten, jeweils mit allen Trainingssignalen, für verschiedene diskrete  $\tau_1$  und  $\tau_2$  erlaubt dann die Wahl der geeigneten Verzögerungen. Die Ergebnisse sind in Tabelle 5.1 zusammengefaßt. Der geringste Fehler ergibt sich für  $\tau_1 = \tau_2 = 1$  und bestätigt somit die Modellstruktur aus Bild 5.2. Mit dieser Struktur wurde während des Lernens der in Bild 5.3 dargestellte Verlauf des

**Tabelle 5.1:** Mittlerer quadratischer Fehler für verschiedene Verzögerungen  $\tau_1$  und  $\tau_2$  nach 100 Lernschritten

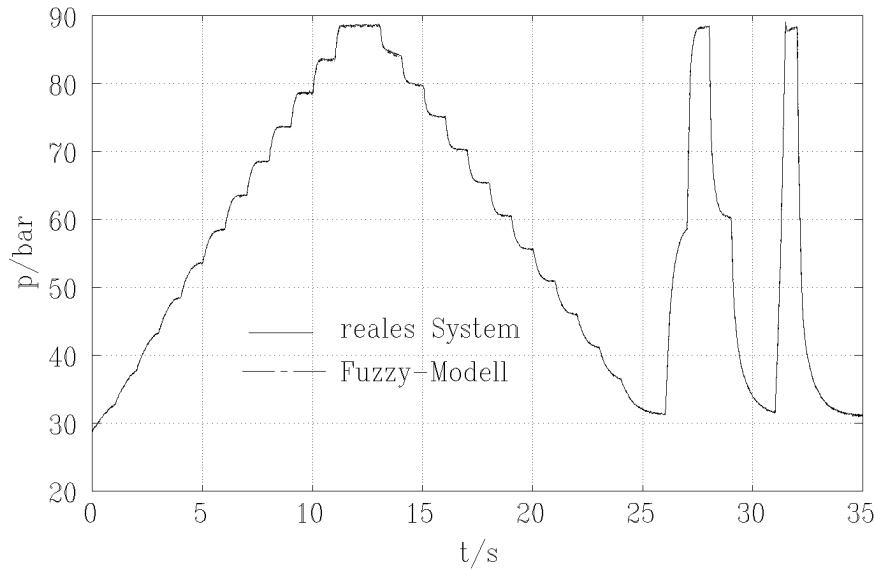
		$\tau_1$				
		1	2	3	4	5
$\tau_2$	1	1,1189931	3,0494782	5,3158722	6,9058264	8,0875009
	2	1,6699501	3,4300284	5,0704816	6,1642116	6,9475419
	3	4,1494426	4,3843221	5,4002957	6,1517117	6,6432140

mittleren quadratischen Fehlers gefunden. Ein Lernschritt entspricht dabei dem Lernen aller Daten des Trainingssignals. Zur besseren Ablesbarkeit wurde eine halblogarithmische Darstellung gewählt. Nach 1100 Lernschritten verbessert sich das Fuzzy-Modell nur noch unwesentlich, so daß an dieser Stelle der Lernvorgang abgebrochen wurde. Mit der dann vorliegenden Relationalmatrix sind alle weiteren in diesem Abschnitt vorgestellten



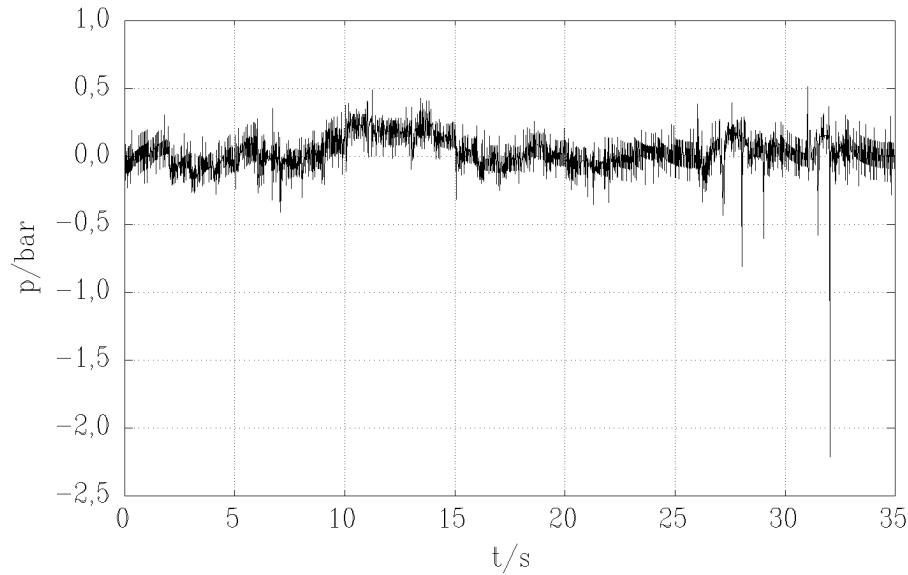
**Bild 5.3:** Verlauf des mittleren quadratischen Fehlers

Ergebnisse erzielt worden. Bild 5.4 zeigt den Verlauf des gemessenen Druckes  $p_1$  und den Ausgang  $\hat{p}_1$  des Fuzzy-Modells bei Einschrittprädiktion. Abweichungen sind praktisch nicht zu sehen. Erst der Graph der Differenz zwischen  $p_1$  und  $\hat{p}_1$  (Bild 5.5) läßt

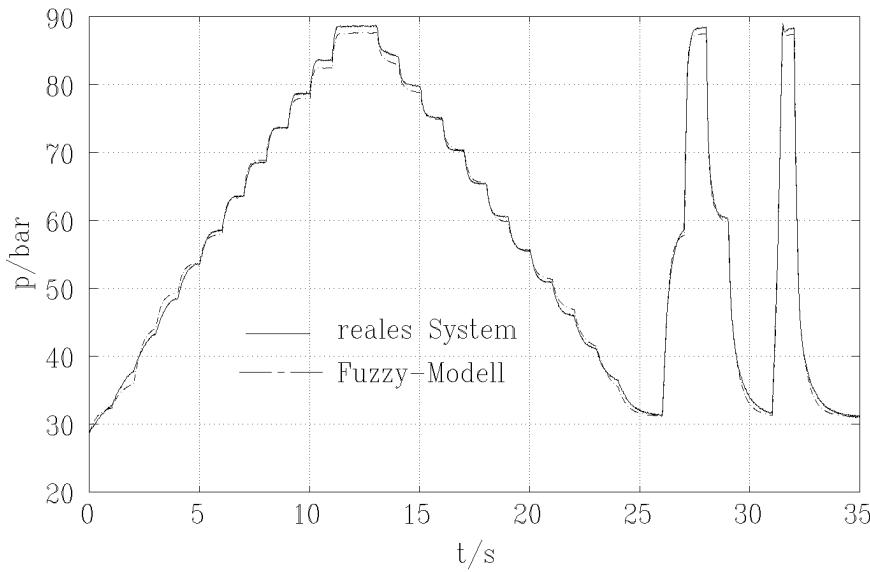


**Bild 5.4:** Speicherdruck  $p_1$  und Einschrittprädiktion  $\hat{p}_1$

den Fehler erkennen. Dabei wird allerdings bereits der Einfluß des Meßrauschen und des Quantisierungsfehlers deutlich. Die rekursive Auswertung (Bild 5.6) ergibt dagegen bereits sichtbare Abweichungen. Dennoch kann das Ergebnis als sehr gut gelten; die nichtlinearen Charakteristiken sind hier klar wiedergegeben. Die Qualität des gefundenen Modells wird allerdings noch deutlicher, wenn man es mit den Testsignalen, die eine andere Charakteristik besitzen, beaufschlagt und rekursiv ausgewertet.



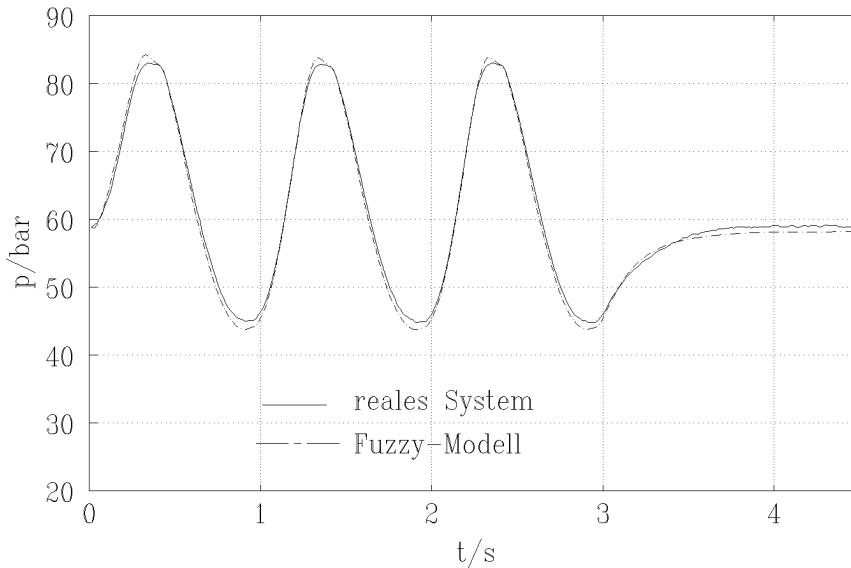
**Bild 5.5:** Differenz zwischen  $p_1$  und der Einschrittprädiktion  $\hat{p}_1$ , Modell 1



**Bild 5.6:** Speicherdruck  $p_1$  und rekursive Auswertung  $\hat{p}_1$ , Modell 1

Das Ergebnis für das Testsignal 2 ist als Beispiel in Bild 5.7 dargestellt. Auch hier wird eine sehr gute Übereinstimmung zwischen Soll- und Istwert erreicht. Insbesondere wird deutlich, daß keine Überparametrisierung eintritt und auch solche Signale, die nicht trainiert wurden, gut modelliert werden.

Die Ergebnisse in Form des mittleren quadratischen Fehlers für das Trainingssignal und alle Testsignale sind in Tabelle 5.2 angegeben. Neben der Einschrittprädiktion und der rekursiven Auswertung ist auch der Fehler eines sogenannten Minimalmodells gegeben (Kluge 1992). Dabei ist der Ausgang  $\hat{y}(k)$  des Modells der Ausgangswert des realen Systems vom vorherigen Tastschritt  $y(k-1)$ . Hier ist eine rekursive Auswertung selbstverständlich nicht möglich. Die Plots für einige der in Tabelle 5.2 aufgeführten Fälle finden sich im



**Bild 5.7:** Speicherdruck  $p_1$  und rekursive Auswertung  $\hat{p}_1$ , Modell 1, Testsignal 2

Anhang B. Abschließend sei auf einen weiteren interessanten Aspekt hingewiesen: Bekanntermaßen kann ein Fuzzy-System als Kennfeld dargestellt werden. Bei zwei Eingängen, die das betrachtete Modell aufweist, ist eine graphische Darstellung möglich (Bild 5.8). Wie erwartet, stellt sich das Kennfeld in weiten Bereichen als nahezu linear dar. Erst

**Tabelle 5.2:** Mittlerer quadratischer Fehler für verschiedene Signale, Modell 1

	$q_m$ (Einschritt)	$q_m$ (rekursiv)	$q_m$ (Minimalmodell)
Trainingssignal	0,0199	0,4983	0,1843
Testsignal 1	0,0217	0,6013	0,1718
Testsignal 2	0,0302	0,7336	1,1293
Testsignal 3	0,0308	0,9787	0,3143
Testsignal 4	0,0534	0,5511	3,4162
Testsignal 5	0,0218	0,6452	0,1475
Testsignal 6	0,0262	0,7612	0,2102

bei genauerer Untersuchung findet man auch in diesen Bereichen Nichtlinearitäten. Die starken Abweichungen an den Ecken erklären sich daraus, daß sich hier keine Werte des Trainingssignals finden, also die Initialwerte für die Relationalmatrix unverändert blieben. Zur Verdeutlichung sei das Bild 5.9 betrachtet, auf dem neben den Höhenlinien des Kennfeldes die Eingangsgrößen des Modells für das Trainingssignal eingezeichnet sind. In den genannten Bereichen gibt es keine Werte. Die zunächst schwach scheinende Nichtlinearität legt es nahe, als konventionelles Vergleichssystem zu dem Fuzzy-Modell ein lineares  $P_{T1}$ -System zu betrachten. Dies geschieht im folgenden Abschnitt.

**Bild 5.8:** Kennfeld des identifizierten Fuzzy–Modells 1

**Bild 5.9:** Trainingsdaten im Eingangsraum (\*) und Höhenlinien des Ausgangs

### 5.3 Modellierung als $P_{T_1}$ –System

Wird die untersuchte Strecke als  $P_{T_1}$ –System modelliert, erhält man die folgende Systemgleichung

$$p_1(k) = \Theta_1 p_1(k-1) + \Theta_2 p_2(k-1) \quad . \quad (5.4)$$

Zur Bestimmung der Parameter wird ein einfaches Least–Squares–(LS)–Verfahren eingesetzt (Isermann 1988). Besteht das Trainingssignal aus  $(n+1)$  Werten, so errechnen sich die im Sinne des Fehlerquadrates optimalen Parameter mit

$$\hat{\Theta} = [\psi^T \ \psi]^{-1} \psi^T \mathbf{Y} \quad , \quad (5.5)$$

mit

$$\mathbf{Y} = \begin{bmatrix} p_1(k-n) \\ \vdots \\ p_1(k) \end{bmatrix} \quad , \quad \Theta = \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} \quad (5.6)$$

und

$$\psi = \begin{bmatrix} p_1(k-1-n) & p_2(k-1-n) \\ \vdots & \vdots \\ p_1(k-1) & p_2(k-1) \end{bmatrix} \quad . \quad (5.7)$$

Diese sind für das verwendete Trainingssignal

$$\hat{\Theta} = \begin{bmatrix} 0,9085 \\ 0,0908 \end{bmatrix} \quad . \quad (5.8)$$

Das  $P_{T_1}$ –System besitzt mit der Tastzeit  $T = 15$  ms also die Verstärkung

$$K = \frac{\Theta_2}{1 - \Theta_1} = 0,9923 \quad (5.9)$$

und die Zeitkonstante

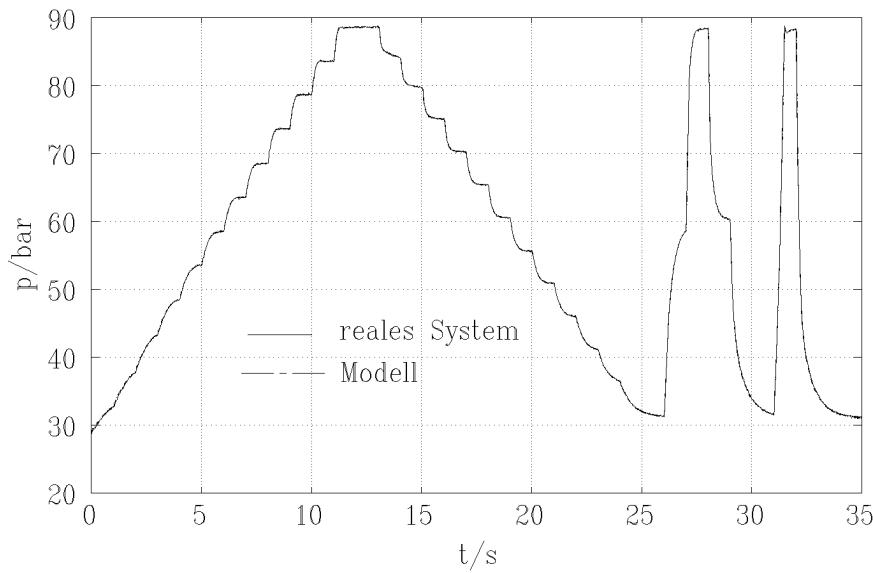
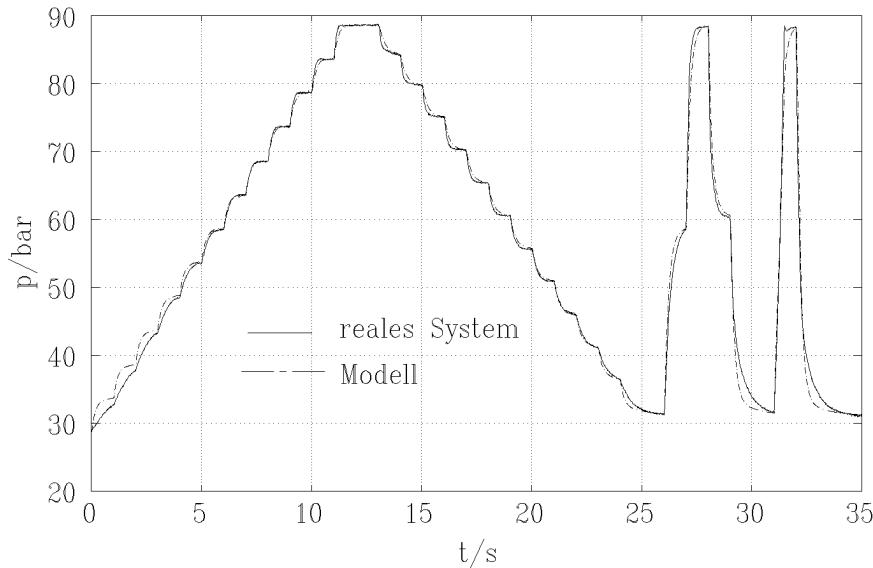
$$T_1 = \frac{-T}{\ln \Theta_1} = 10,43T = 157 \text{ ms} \quad . \quad (5.10)$$

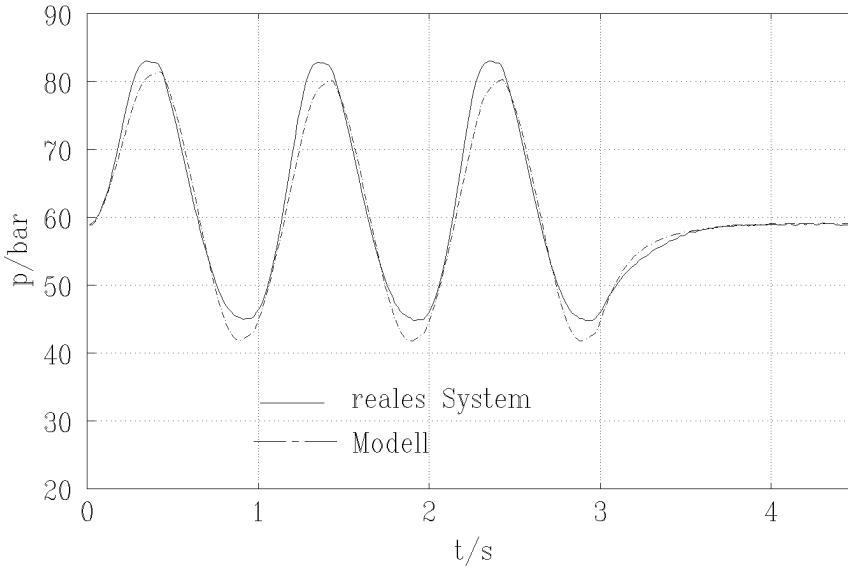
Damit erweist sich auch die gewählte Abtastzeit  $T \approx 0,1T_1$  als günstig. Weiterhin erreicht der Speicherdruck wegen der Verstärkung, die nahezu eins ist, fast exakt den Systemdruck (Abweichung < 0,8 %).

Die Einschrittprädiktion (Bild 5.10) zeigt gute Ergebnisse. Jedoch ergeben sich bereits bei der rekursiven Auswertung (Bild 5.11), wo echtes  $P_{T_1}$ –Verhalten zu erkennen ist, relativ große Abweichungen, die klar auf das nichtlineare Verhalten des technischen Systems zurückzuführen sind. Dafür treten, wie bereits angesprochen, auch hier keine bleibenden Abweichungen bei konstantem Eingang zwischen System und Modell auf. Noch deutlicher wird der Fehler aus der Linearisierung bei rekursiver Auswertung für das Testsignal 2 (Bild 5.12). Der Grund ist wiederum in der dynamischen Nichtlinearität zu suchen. Die Ergebnisse für alle Signale sind in Tabelle 5.3 zusammengefaßt. Das Fuzzy–Modell ist in allen Fällen dem linearen Modell deutlich überlegen. Das wird weiterhin auch in den Plots für andere Testsignale deutlich, die in Anhang C zu finden sind.

**Tabelle 5.3:** Mittlerer quadratischer Fehler für verschiedene Signale,  $P_{T_1}$ -System

	$q_m$ (Einschritt)	$q_m$ (rekursiv)
Trainingssignal	0,0465	1,9323
Testsignal 1	0,0454	2,0191
Testsignal 2	0,1167	4,4769
Testsignal 3	0,0523	3,0752
Testsignal 4	0,3485	7,9153
Testsignal 5	0,0247	0,9162
Testsignal 6	0,0298	1,1106

**Bild 5.10:** Speicherdruck  $p_1$  und Einschrittprädiktion  $\hat{p}_1$ ,  $P_{T_1}$ -System**Bild 5.11:** Speicherdruck  $p_1$  und rekursive Auswertung  $\hat{p}_1$ ,  $P_{T_1}$ -System



**Bild 5.12:** Speicherdruck  $p_1$  und Einschrittprädiktion  $\hat{p}_1$ ,  $P_{T1}$ -System, Signal 2

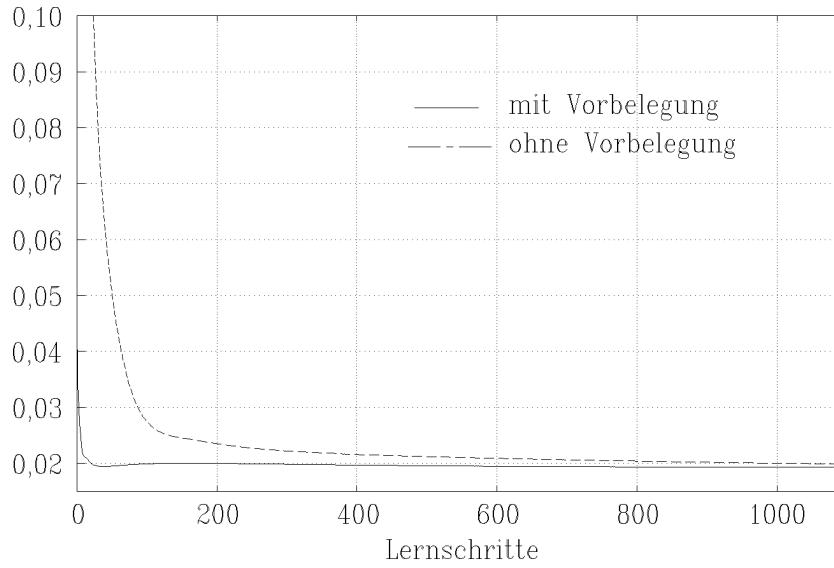
## 5.4 Modell 2 mit Vorbelegung

Wie im vorhergehenden Abschnitt gezeigt wurde, reicht die Güte eines  $P_{T1}$ -Modells zur Approximation des Drossel–Speicher–Systems nicht aus. Jedoch erscheint es sinnvoll, die Modellierung als  $P_{T1}$ -System zur Vorinformation für die nachfolgende Identifikation der Parameter zu nutzen. Dies kann auf zwei Arten geschehen.

### 5.4.1 Vorbelegung durch Lernen

Die erste Möglichkeit besteht darin, sich zunächst unter Verwendung des mit dem LS–Verfahren identifizierten  $P_{T1}$ -Modells ideale, d. h. über dem Eingangsraum gleichverteilte, Daten zu beschaffen und mit diesen das Fuzzy–Modell zu trainieren. Dadurch verhält sich dieses zunächst wie das  $P_{T1}$ -System, ist also bereits eine relativ gute Näherung des realen Systems. Die Trainingsdaten dienen dann der Feinabstimmung des Modells.

Bild 5.13 zeigt den Verlauf des quadratischen Fehlers während des Lernens für ein derartig vorbelegtes Fuzzy–Modell und den des Fuzzy–Modells 1. Das vorbelegte Modell erreicht bereits nach weniger als 50 Schritten einen Fehler, der geringer ist als der des anderen Modells nach 1100 Lernschritten. Dies dokumentiert den großen Vorteil der sinnvollen Vorbelegung. Eine Bewertung der Ergebnisse ist aber wiederum erst anhand der Testsignale möglich. Die Tabelle 5.4 zeigt die Fehler für die verschiedenen Signale bei Einschrittprädiktion und rekursiver Auswertung. Die Werte wurden nach 38 Lernschritten, also dem ersten Minimum der Fehlers, nach 150 Lernschritten und nach 1100 Lernschritten bestimmt. Dabei fallen zwei unterschiedliche Trends auf: Die Fehler der Testsignale 1, 5 und 6 folgen in etwa dem des Trainingssignals: Nach dem Minimum bei 38 Lernschritten steigen sie etwas an (150 Lernschritte), um langfristig wiederum einen kleinen Wert anzunehmen. Diese Ähnlichkeit erscheint plausibel, da es sich in allen Fällen um prinzipiell



**Bild 5.13:** Fehlerverläufe für vorbelegtes und nicht vorbelegtes Fuzzy–Modell

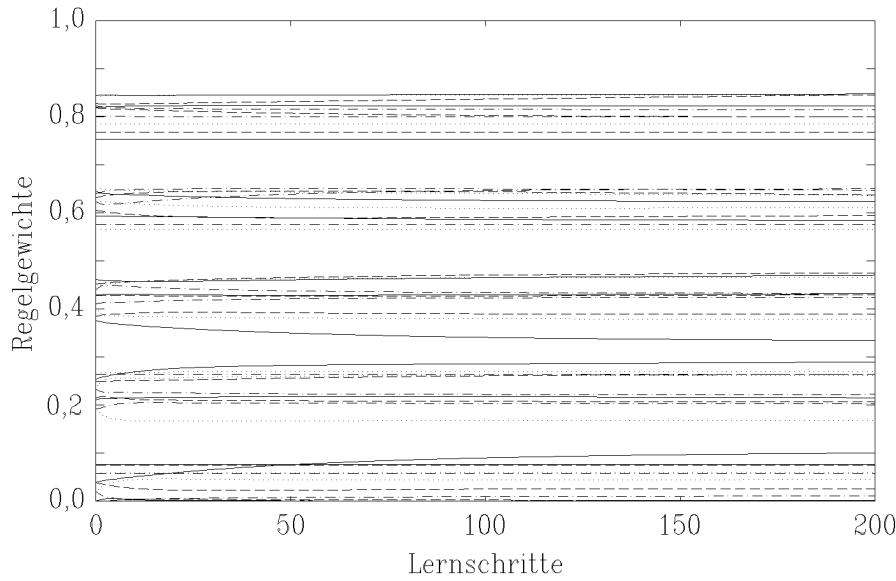
**Tabelle 5.4:** Mittlerer quadratischer Fehler für verschiedene Signale, Fuzzy–Modell mit  $P_{T1}$ –System vorbelegt

Lernschritte	$q_m$ (Einschritt)			$q_m$ (rekursiv)		
	38	150	1100	38	150	1100
Trainingssignal	0,0194	0,0200	0,0193	0,4511	0,5010	0,4558
Testsignal 1	0,0212	0,0219	0,0212	0,5590	0,6147	0,5673
Testsignal 2	0,0519	0,0425	0,0375	1,7331	1,2261	1,0371
Testsignal 3	0,0487	0,0405	0,0365	2,1200	1,5029	1,2995
Testsignal 4	0,0659	0,0587	0,0540	0,8435	0,6177	0,5818
Testsignal 5	0,0262	0,0254	0,0237	0,6233	0,8459	0,7678
Testsignal 6	0,0337	0,0316	0,0287	0,8354	1,0463	0,9206

ähnliche Signale handelt, nämlich eine Folge von Sprüngen. Dagegen nimmt der Fehler bei den Signalen 2, 3 und 4 kontinuierlich ab. Zur Erinnerung: Hierbei handelt es sich um sinusförmige Signale.

Neben dem Verlauf des mittleren quadratischen Fehlers während des Lernens und der Betrachtung der Testsignale ist die Entwicklung der identifizierten Parameter ein dritter Aspekt bei der Beurteilung des vorbelegten Fuzzy–Modells. Das Bild 5.14 zeigt den Verlauf der Regelgewichte des vorbelegten Modells für die ersten 200 Lernschritte. Es ist auffällig, daß die Parameter sich langsam und stetig ändern. Dieser Verlauf läßt darauf schließen, daß die gefundenen Startparameter bereits nahe bei einem lokalen Minimum liegen.

Es bleibt festzuhalten, daß die Ergebnisse insgesamt sehr gut sind. Sie erreichen zwar nicht ganz die Qualität des Modells 1, sind aber für ein Simulationsmodell ausreichend.



**Bild 5.14:** Verlauf der Regelgewichte während der Identifikation mit Vorbelegung

Einige Signalverläufe von Modell und System nach 150 Lernschritten sind im Anhang D dargestellt.

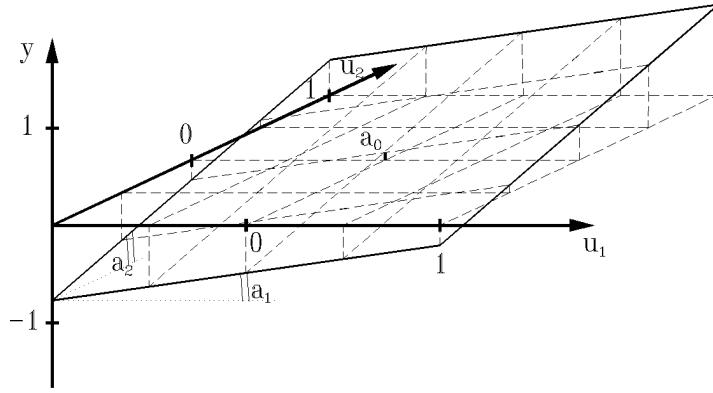
#### 5.4.2 Analytische Vorbelegung

Die zweite Möglichkeit zur Vorbelegung eines Fuzzy-Modells besteht darin, die exakte Relationalmatrix für ein bestimmtes Ein-/Ausgangsverhalten zu berechnen. Voraussetzung ist dabei, daß zu diesem Problem eine Lösung existiert und auch gefunden werden kann. Für den Fall eines Fuzzy-Systems mit zwei Eingängen und einem Ausgang wird im folgenden gezeigt, wie ein beliebiger linearer Zusammenhang mit diesem dargestellt werden kann.

Es werde ein lineares statisches System mit den zwei Eingängen  $u_1$  und  $u_2$  und dem Ausgang  $y$  betrachtet. Die Eingänge sind auf das Intervall  $[-1; 1]$  normiert. Der Ausgang  $y$  ist so normiert, daß er für diese Eingänge immer innerhalb des Intervalls  $[-1; 1]$  liegt. Das System wird durch die Gleichung

$$y = a_0 + a_1 u_1 + a_2 u_2 \quad (5.11)$$

beschrieben. Die Werte von  $a_0$ ,  $a_1$  und  $a_2$  sind durch die erwähnte Begrenzung von  $y$  ebenfalls beschränkt. Eine graphische Darstellung von Gl. (5.11) ist in Bild 5.15 gegeben. Durch die genannte Normierung kann jedes Fuzzy-Modell auf dem gleichen Grundbereich  $u_1 \times u_2 \rightarrow [0; 1]$  definiert werden. Für die Eingänge werden gleichverteilte, symmetrische, dreiecksförmige Fuzzy-Mengen gewählt, die ein sogenanntes Fuzzy-Informationssystem bilden. Ihre Anzahl  $c_i$ ,  $i = 1, 2$ , ist beliebig  $> 1$ . Zur Veranschaulichung wird im Folgenden  $c_1 = c_2 = 3$  gewählt. Für den Ausgang wird der gleiche Typ von Fuzzy-Mengen verwendet, jedoch nur  $c_3 = 2$ . Damit hat das Modell  $c_1 c_2 c_3 = 2d$  Regeln. Es ergibt sich bei der

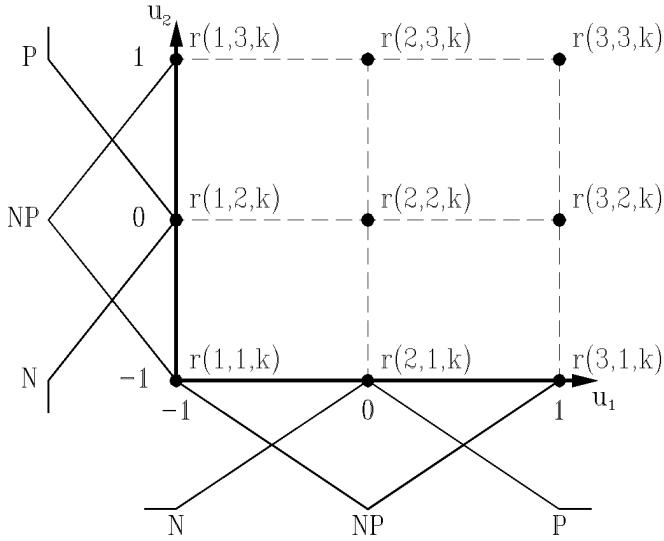


**Bild 5.15:** Graphische Darstellung des linearen Systems

Verwendung des Summen- und des Produktoperators als Ausgang des Fuzzy-Modells analog zu Gl. (2.29)

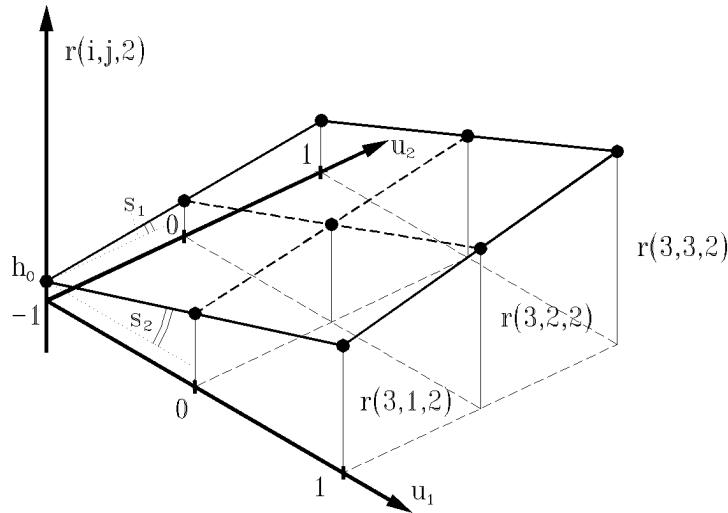
$$y = \frac{\sum_{l=1}^d \alpha_l (-g_{1l} + g_{2l})}{\sum_{l=1}^d (g_{1l} + g_{2l})}. \quad (5.12)$$

In Bild 5.16 ist der Eingangsraum und seine Unterteilung durch die Fuzzy-Mengen des Eingangs dargestellt. Den Knotenpunkten der Maxima der Fuzzy-Mengen sind jeweils

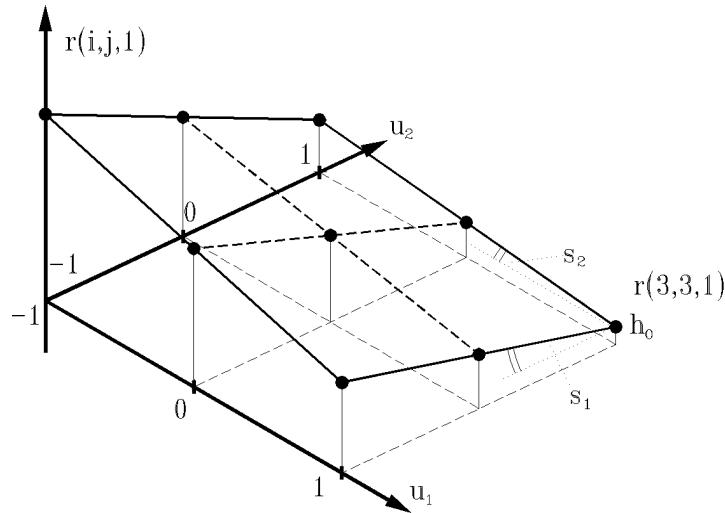


**Bild 5.16:** Unterteilung des Eingangsraumes

die zugehörigen Gewichte  $r(i, j, k)$ , also die Elemente der Relationalmatrix, zugeordnet. Zunächst werden nur die Elemente  $r(i, j, 2)$  betrachtet. Sie werden so gewählt, daß  $r(1, 1, 2) = h_0$  ist und die anderen eine Ebene mit den Steigungen  $s_1$  und  $s_2$  aufspannen. Zur Verdeutlichung ist dies im Bild 5.17 dargestellt: Die Elemente  $r(i, j, 1)$  werden so gewählt, daß  $r(3, 3, 2) = h_0$  ist und die aufgespannte Ebene die Steigung  $-s_1$  und  $-s_2$  hat. Das Ergebnis ist im Bild 5.18 dargestellt. Die Steigungen  $s_1$  und  $s_2$  sind mit



**Bild 5.17:** Graphische Darstellung der Regelgewichte  $r(i,j,2)$



**Bild 5.18:** Graphische Darstellung der Regelgewichte  $r(i,j,1)$

$$\begin{aligned} s_1 &= -\frac{a_2}{a_1 + a_2 - 1} h_0, \\ s_2 &= -\frac{a_1}{a_1 + a_2 - 1} h_0. \end{aligned} \quad (5.13)$$

gegeben.  $h_0$  ist so zu wählen, daß die Gewichte eine geeignete Größe haben, also z. B.  $r(i,j,k) \leq 1$ . Die Werte in der Mitte der von  $r(i,j,1)$  und  $r(i,j,2)$  aufgespannten Ebene sind gleich. Sie ergeben sich zu

$$r_m = h_0 + s_1 + s_2. \quad (5.14)$$

Nun werden die Werte  $r(i,j,1)$  um einen Wert  $\delta$  erniedrigt, die  $r(i,j,2)$  und  $\delta$  erhöht:

$$\begin{aligned} r(i,j,1) &= r(i,j,1) - \delta \\ r(i,j,2) &= r(i,j,1) + \delta \end{aligned} \quad (5.15)$$

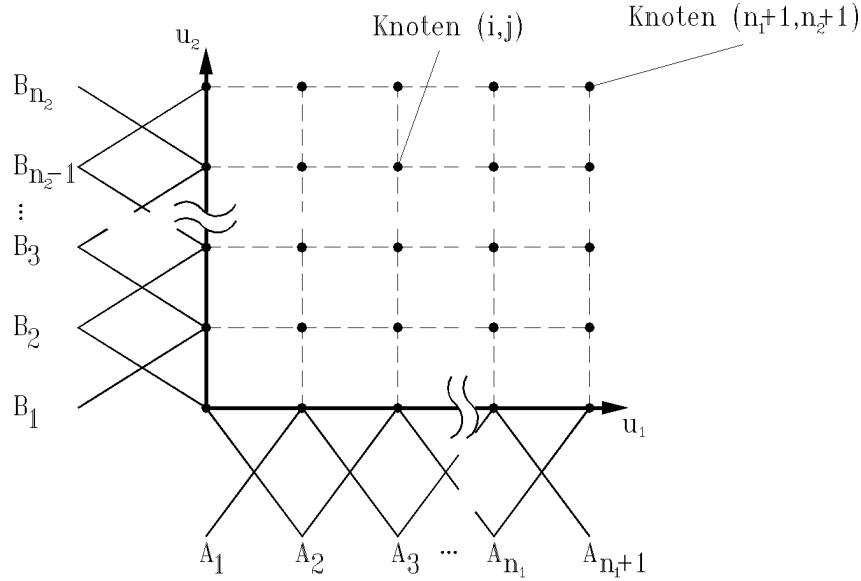
Der Wert  $\delta$  ergibt sich zu

$$\delta = a_0 r_m \quad (5.16)$$

Der Beweis der obigen Formel kann durch Nachrechnen geführt werden:

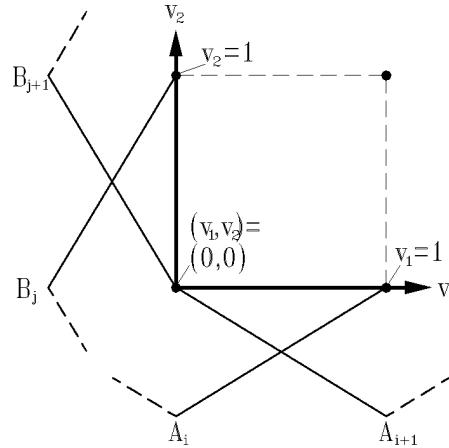
**Beweis:**

Der Grundbereich  $U_1 \times U_2 : [-1, 1] \times [-1, 1]$  wird gemäß Bild 5.19 durch  $i = 1 \dots (n_1 + 1)$  Fuzzy-Mengen für  $u_1$  und  $j = 1 \dots (n_2 + 1)$  Fuzzy-Mengen für  $u_2$  in  $(n_1 + 1)(n_2 + 1)$  Bereiche unterteilt. Nun wird ein beliebiger Teilbereich (Bild 5.20) herausgegriffen und



**Bild 5.19:** Unterteilung des Eingangsraums

näher betrachtet. Die Hilfskoordinaten  $v_1$  und  $v_2$  sind so festgelegt, daß sie im betrachteten



**Bild 5.20:** Teilbereich des Eingangsraums

Bereich von 0 bis 1 liegen:

$$v_1 = (u_1 + 1) \frac{n_1}{2} - i + 1, \quad$$

$$v_2 = (u_2 + 1) \frac{n_2}{2} - j + 1. \quad (5.17)$$

Nun kann man die Zugehörigkeitsgrade von  $v_1$  und  $v_2$  zu den gezeigten Fuzzy-Mengen angeben:

$$\begin{aligned} A_i(v_1) &= 1 - v_1 & A_{i+1}(v_1) &= v_1, \\ B_j(v_2) &= 1 - v_2 & B_{j+1}(v_2) &= v_2. \end{aligned} \quad (5.18)$$

Die Regelgewichte ergeben sich aus den vorher angegebenen Überlegungen zu

$$\begin{aligned} r(i, j, 1) &= -\delta + h_0 + \frac{2s_2}{n_1}(n_1 - i + 1) + \frac{2s_1}{n_1}(n_2 - j + 1), \\ r(i, j, 2) &= +\delta + h_0 + \frac{2s_2}{n_1}(i - 1) + \frac{2s_1}{n_1}(j - 1). \end{aligned} \quad (5.19)$$

Nach der Anpassung der Notation von Gl. (5.12) erhält man

$$y = \frac{\sum_{i=1}^2 \sum_{j=1}^2 A_i B_j [-r(i, j, 1) + r(i, j, 2)]}{\sum_{i=1}^2 \sum_{j=1}^2 A_i B_j [r(i, j, 1) + r(i, j, 2)]}. \quad (5.20)$$

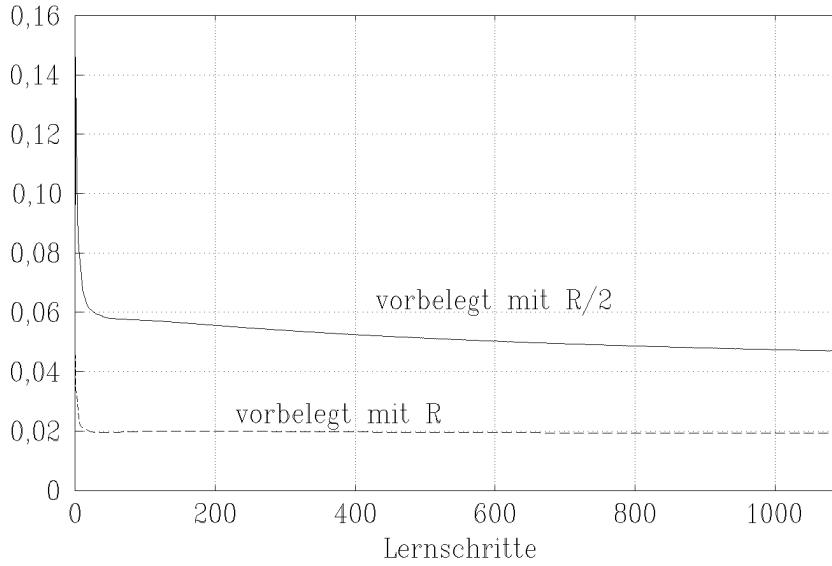
Durch Einsetzen der Gln. (5.19), (5.18), (5.17), (5.15), (5.14) und (5.13) in Gl. (5.20) erhält man nach Auflösung und Vereinfachungen den Ausdruck

$$y = a_0 + a_1 u_1 + a_2 u_2. \quad (5.21)$$

Dies ist genau Gl. (5.11), womit der Beweis abgeschlossen ist.  $\square$

Eine Erweiterung auf Fuzzy-Systeme höherer Ordnung erscheint möglich, wurde aber noch nicht durchgeführt.

Im Gegensatz zu der im vorhergehenden Abschnitt dargestellten Vorbelegung durch Lernen liegt hier eine schnelle und exakte Lösung vor. Mit der so gefundenen Relationalmatrix werden dann auch die gleichen guten Konvergenzeigenschaften erzielt, so daß auf eine Wiedergabe der Ergebnisse hier verzichtet wird. Dies gilt jedoch nur dann, wenn beide Relationalmatrizen gleich groß sind. Das kann durch den Normierungsfaktor  $h_0$  erreicht werden, mit dem sich die Ergebnisse aus Gl. (5.19) beliebig skalieren lassen. Wird aber  $h_0$  falsch gewählt, ergibt sich eine deutlich schlechtere oder überhaupt keine Konvergenz der Ergebnisse. Als Beispiel wurde die Relationalmatrix, die aus den Gln. (5.19) für das untersuchte System gewonnen wurde, so skaliert, daß ihre Elemente etwa halb so groß wie die der gelernten Matrix waren. Die Fehlerentwicklung für die beiden Matrizen als Anfangswerte bei anschließendem Lernen mit den Trainingsdaten zeigt Bild 5.21. Die Erklärung für die unterschiedliche Fehlerentwicklung ergibt sich bei näherer Betrachtung der Ausgangsgleichung für das Fuzzy-Modell (Gl. 2.29) und des Lernalgorithmus (Gl. 3.9).



**Bild 5.21:** Fehlerentwicklung bei unterschiedlicher Vorbelegung der Relationalmatrix

Zunächst ist anhand von Gl. (2.29) noch einmal zu sehen, daß eine Skalierung  $\beta$  aller Elemente  $g_{jl}$  der Relationalmatrix sich nicht auf das Ergebnis auswirkt:

$$\hat{y} = \frac{\sum_{l=1}^d \alpha_l (-g_{1l}\beta + g_{2l}\beta)}{\sum_{l=1}^d \alpha_l (g_{1l}\beta + g_{2l}\beta)} = \frac{\sum_{l=1}^d \alpha_l (-g_{1l} + g_{2l})}{\sum_{l=1}^d \alpha_l (g_{1l} + g_{2l})} . \quad (5.22)$$

Dagegen ist die Größe der Ableitungen (3.10) und (3.11) umgekehrt proportional zu der Skalierung  $\beta$ :

$$\frac{\partial \hat{y}}{\partial \Theta_i} \sim \frac{1}{\beta} . \quad (5.23)$$

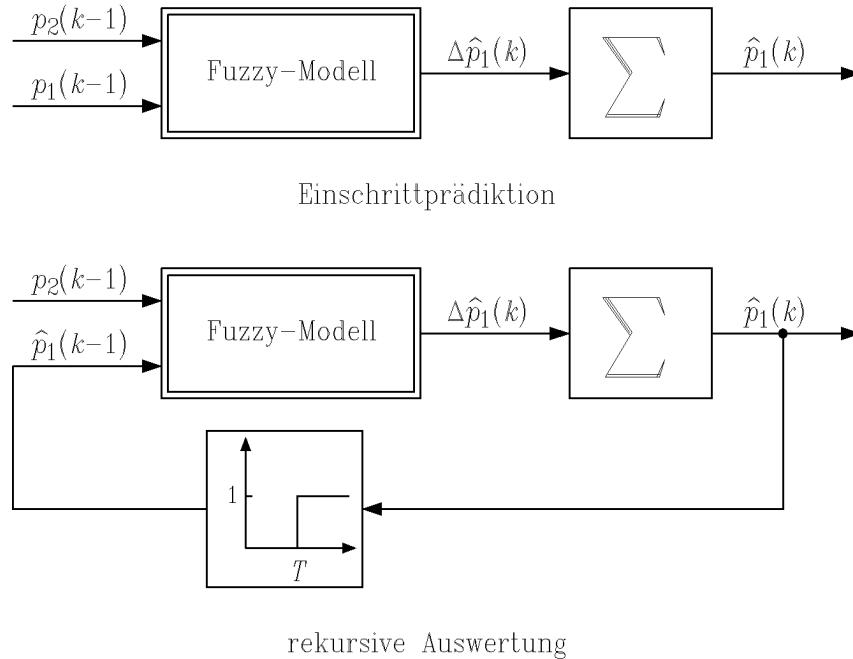
Damit ist die Änderung von  $\Theta$ , die sich aus Gl. (3.9) ablesen läßt, ebenfalls proportional zu  $1/\beta$ . Wird also  $\Theta$  z. B. mit dem Faktor 2 multipliziert, ist seine Änderung bei ansonsten unveränderten Bedingungen nur noch halb so groß. Dieser Effekt kann ausgeglichen werden, indem neben  $\mathbf{R}$  auch der Lernfaktor  $\gamma$  entsprechend skaliert wird:

$$\mathbf{R}^* = \mathbf{R}\beta \implies \gamma^* = \beta^2\gamma . \quad (5.24)$$

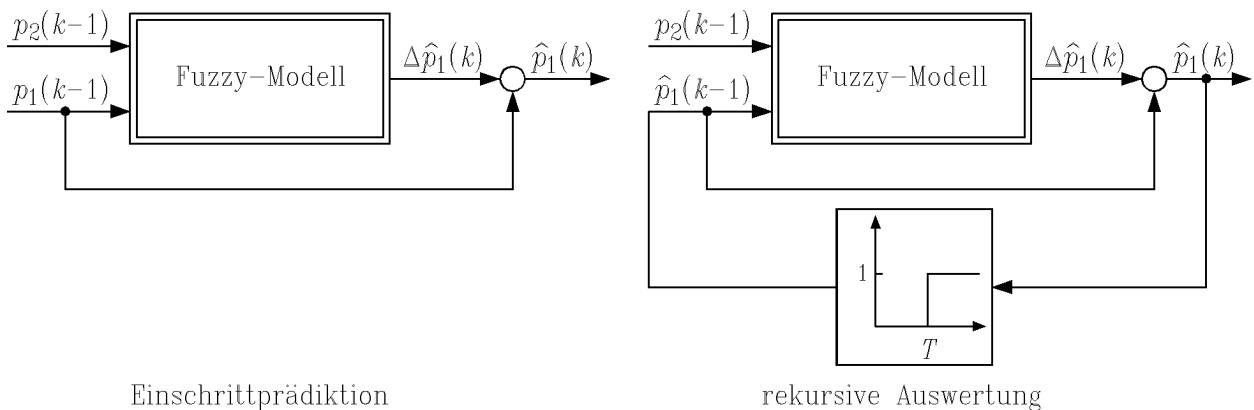
Es besteht jedoch weiterhin das Problem, die Initialrelationalmatrix bzw. den Lernfaktor  $\gamma$  zu wählen. Darauf wird in diesem Bericht nicht weiter eingegangen. Im Ausblick finden sich hierzu aber einige Hinweise.

## 5.5 Modell 3: Inkrementales Modell

Bei den bisher vorgestellten Modellen wurde immer der Speicherdruck zum nächsten Abtastzeitpunkt vorhergesagt. Ein anderer Ansatz besteht darin, nur die Änderung des Speicherdrucks vorherzusagen. Für dieses Modell bestehen zwei Möglichkeiten der Einschrittprädiktion und analog zwei zur rekursiven Auswertung. Diese sind jeweils in Bild 5.22 und 5.23 dargestellt; in diesem Bericht wird nur die Integration der Druckänderung

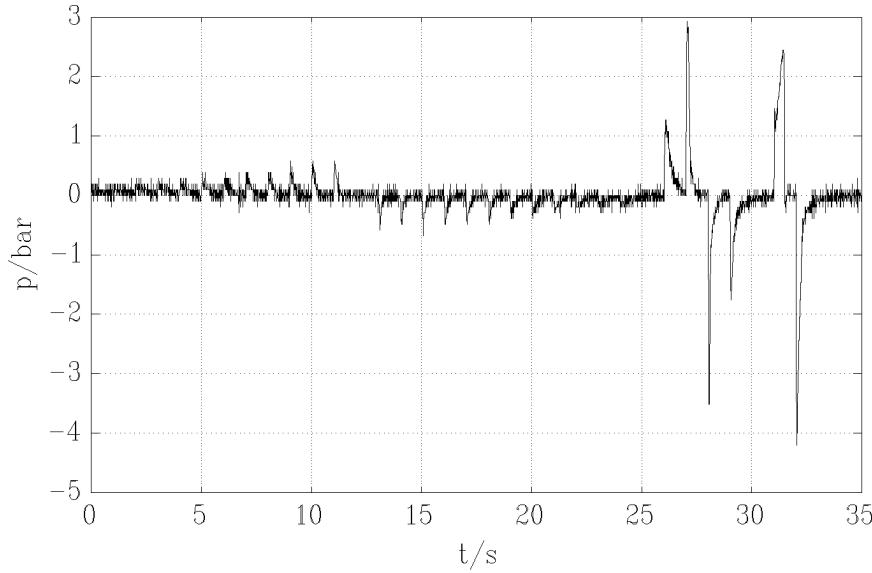


**Bild 5.22:** Integration der Druckänderung



**Bild 5.23:** Addition von Ausgangsdruck und Druckänderung

untersucht (Bild 5.22). Besonders problematisch bei der Schätzung der Druckänderung zwischen zwei Abtastzeitpunkten sind die starken Störungen der Messung. Dies wird bei der Betrachtung von  $\Delta p_1(k)$  aus dem Trainingssignal deutlich (Bild 5.24). In diesem Fall

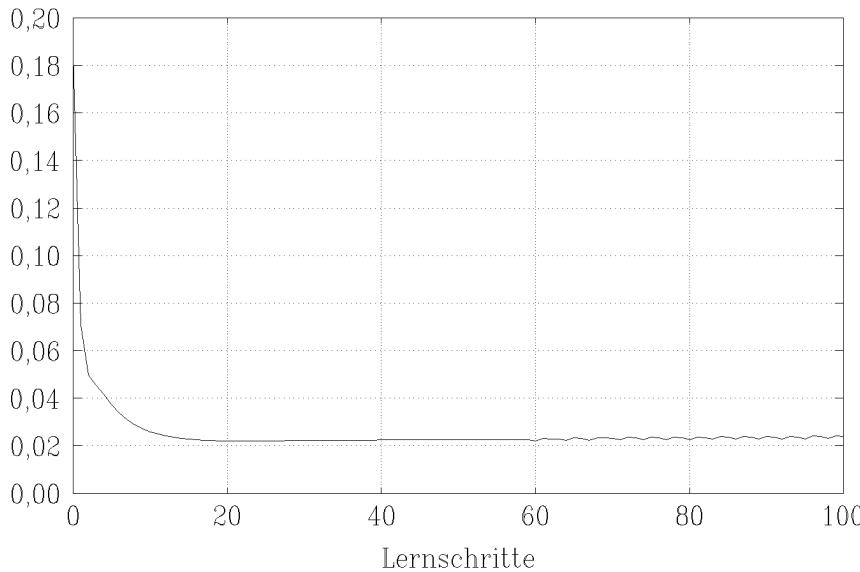


**Bild 5.24:**  $\Delta p_1(k)$  des Trainingssignals

wird auch die Normierung der Ausgangsgröße mit

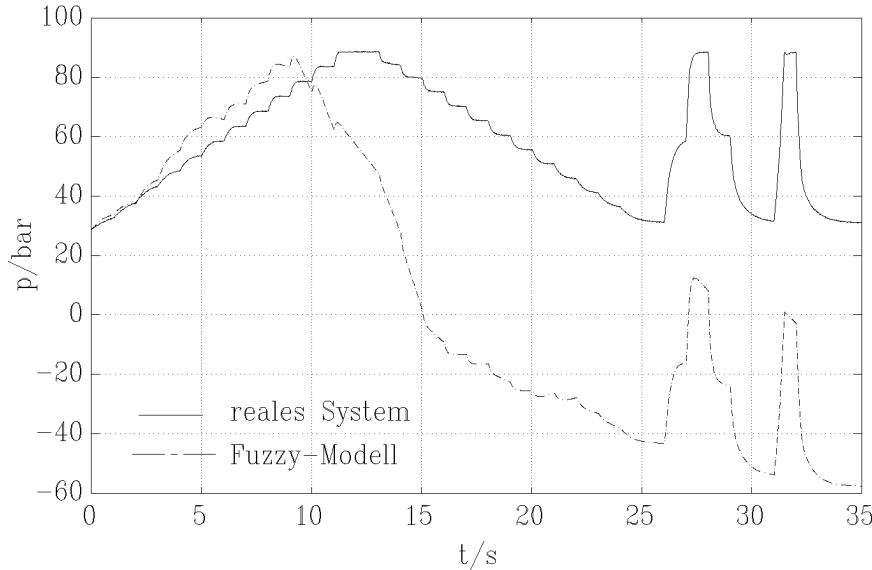
$$\Delta p_{1_n} = \frac{\Delta p_1}{5} \quad (5.25)$$

anders gewählt. Der Verlauf des mittleren quadratischen Fehlers von  $\Delta p_1(k)$  während des Identifikationsvorganges ist in Bild 5.25 dargestellt. Nach anfänglich guter Konvergenz gibt es keine weitere Verbesserung, was höchstwahrscheinlich auf die besagten Störungen zurückzuführen ist. Aus diesem Grund wurde die Relationalmatrix nach 100 Lernschritten



**Bild 5.25:** Verlauf des mittleren quadratischen Fehlers für  $\Delta p_1$  (100 Schritte)

bestimmt und damit die Testsignale ausgewertet. Zunächst sei jedoch das Identifikationssignal bei Einschrittprädiktion und rekursiver Auswertung betrachtet (Bild 5.26 und 5.27).

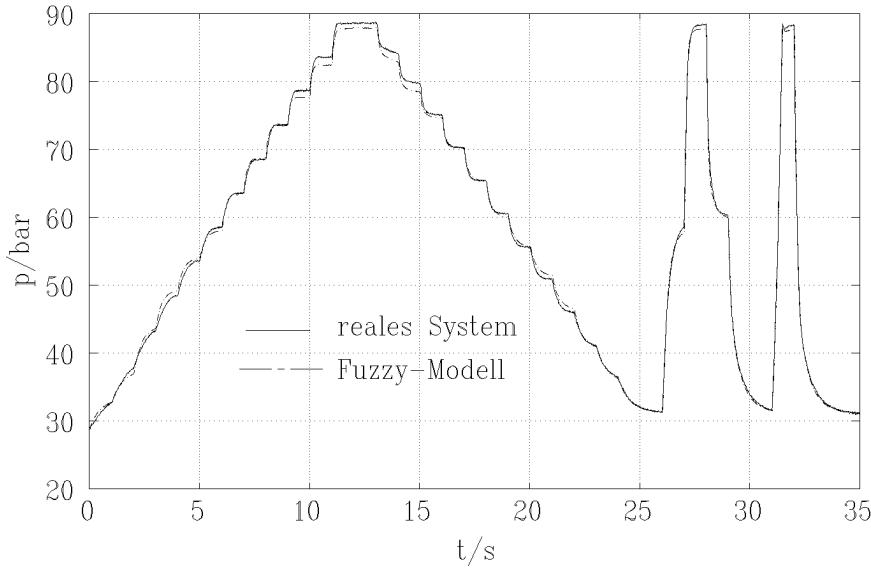


**Bild 5.26:** Einschrittprädiktion von  $\Delta p_1(k)$ , Darstellung von  $\sum \Delta p_1(k)$

**Tabelle 5.5:** Mittlerer quadratischer Fehler für verschiedene Signale, Modell 3

$q_m$ (rekursiv)	
Trainingssignal	0,3631
Testsignal 1	0,4280
Testsignal 2	1,4072
Testsignal 3	1,5094
Testsignal 4	0,8712
Testsignal 5	0,4097
Testsignal 6	0,5481

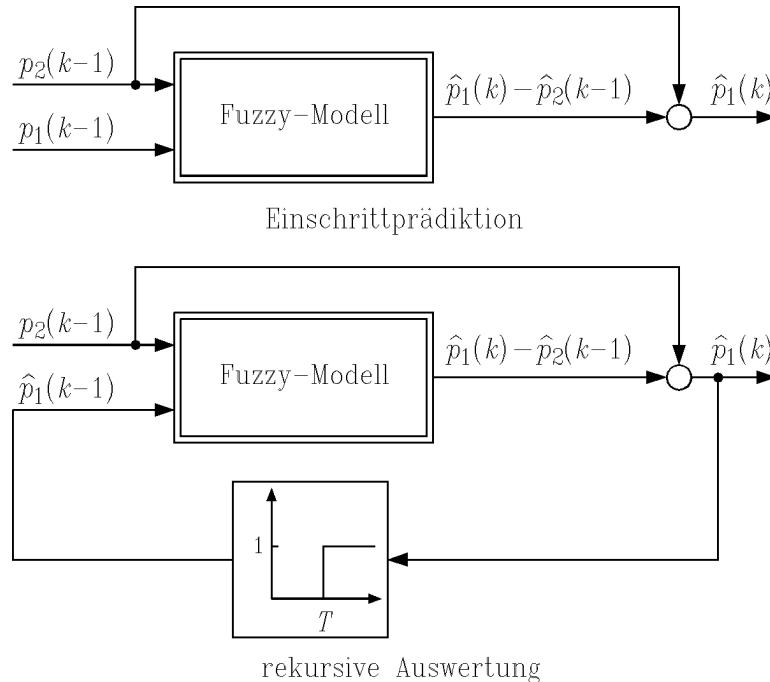
Bei der Einschrittprädiktion tritt der große Fehler des Modells deutlich zu Tage: Durch die Summation wird er weiter addiert und die eigentlich interessierende Größe  $p_1(k)$  driftet weg. So ist wahrscheinlich eine Einschrittprädiktion mit Addition von Ausgangsdruck und Druckänderung (Bild 5.23) hier deutlich vorteilhafter. Sehr interessant ist aber das Ergebnis bei Rückführung der summierten Druckdifferenzen: Das Modell stabilisiert sich selbst und erreicht so eine hervorragende Genauigkeit. Dabei ist zu betonen, daß hier zum erstenmal das rekursive Modell besser ist als die Einschrittprädiktion. Die Betrachtung des quadratischen Fehlers für  $p_1(k)$  (die Einschrittprädiktion wurde wegen der großen Abweichungen nicht berücksichtigt) zeigt die gute Leistung des Modells auch im Vergleich zum Modell 1 (Tabelle 5.5). Diese Ergebnisse bieten sicherlich Ansatzpunkte für Untersuchungen der Stabilität bei dynamischen Fuzzy-Systemen und der Wirkung von Rückführungen. Auffällig ist in jedem Fall die Analogie zwischen der Einschrittprädiktion und einem rückgekoppelten Integrator.



**Bild 5.27:** Rekursive Auswertung von  $\Delta p_1(k)$ , Darstellung von  $\sum \Delta p_1(k)$

## 5.6 Modell 4

Die hier untersuchte Modellstruktur schätzt ebenfalls eine Druckdifferenz; in diesem Fall jedoch die zukünftige Differenz zwischen Speicher- und Systemdruck. Die Modellstruktur ist in Bild 5.28 dargestellt. Als Eingänge werden wiederum  $p_2(k-1)$  und  $p_1(k-1)$



**Bild 5.28:** Struktur von Modell 4

verwendet. Durch den größeren Wertebereich der Differenz  $p_1(k) - p_2(k-1)$  verbessert

**Tabelle 5.6:** Mittlerer quadratischer Fehler für verschiedene Signale, Modell 4

	$q_m$ (Einschritt)	$q_m$ (rekursiv)
Trainingssignal	0,0211	0,6229
Testsignal 1	0,0233	0,7521
Testsignal 2	0,0261	0,4923
Testsignal 3	0,0343	1,0257
Testsignal 4	0,0479	0,4017
Testsignal 5	0,0252	3,8855
Testsignal 6	0,0285	0,9586

sich im Vergleich zum Modell 3 das Verhältnis von Nutz- zu Störsignal. Gleichzeitig ist zu erwarten, daß der positive Effekt der Rückkopplung immer noch auftritt.

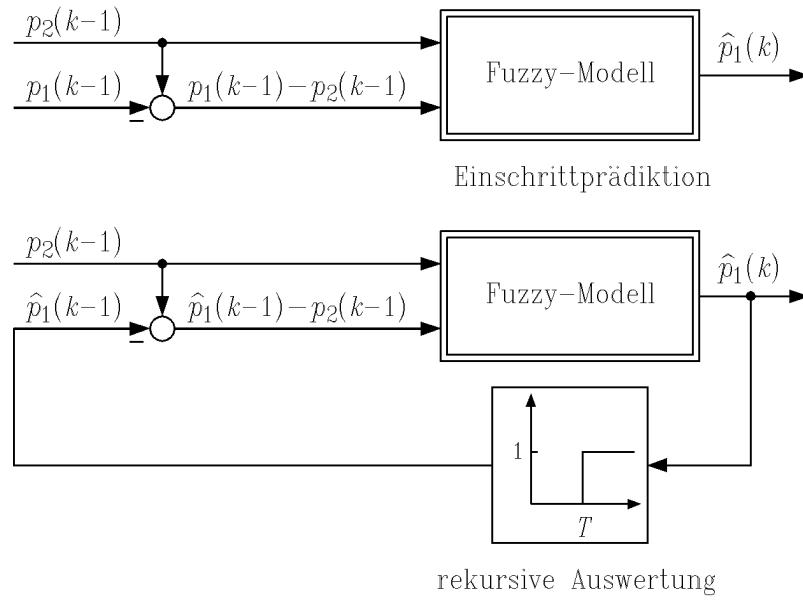
Die quadratischen Fehler für die verschiedenen Signale und Auswertungsmethoden nach wiederum 1100 Lernschritten sind in Tabelle 5.6 zusammengefaßt. Einige Signalverläufe finden sich im Anhang E. Es ist festzuhalten, daß die Ergebnisse sich kaum von denen des Modells 1 unterscheiden. Sie sind sogar teilweise schlechter. Das gilt auch für die rekursive Auswertung. Der erwartete positive Effekt durch die Rückführung des Modellausgangs — und damit implizit des Modellfehlers — trat nicht auf.

## 5.7 Modell 5

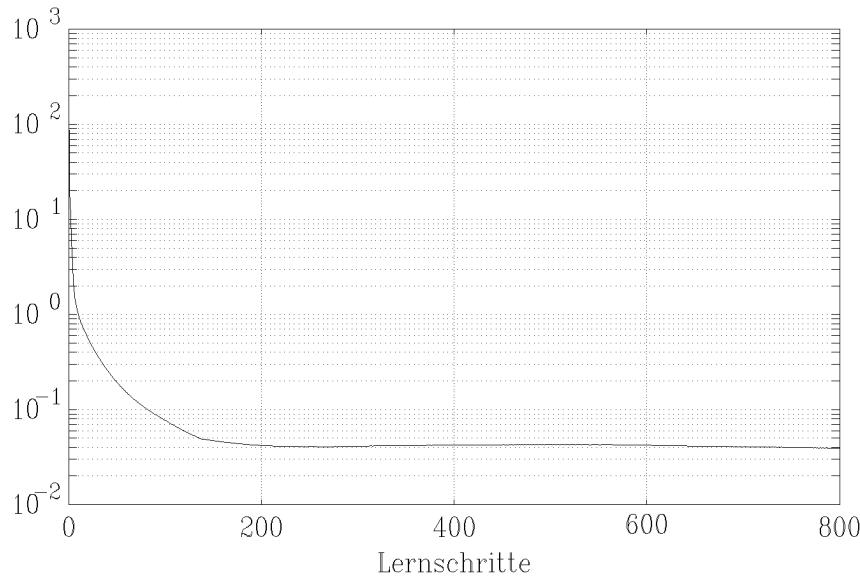
Auf der Basis von Modell 1 wurde schließlich noch eine weitere Modellstruktur untersucht: Bereits bei der Betrachtung der Eingangswerte für das Fuzzy-Modell aus dem Trainingsignal (Bild 5.9) wurde deutlich, daß die meisten Werte auf der Diagonalen liegen,  $p_1$  und  $p_2$  also nur um einen relativ kleinen Wert differieren. Um eine bessere Abdeckung des Eingangsraumes und damit eine bessere Nutzung der verwendeten Referenzfuzzy-Mengen zu erzielen, wird nun neben dem Systemdruck  $p_2(k-1)$  die Druckdifferenz  $p_2(k-1) - p_1(k-1)$  als Eingang verwendet. Wichtig ist dabei, daß dem Fuzzy-Modell so über  $p_2(k)$  weiterhin Informationen über den aktuellen Arbeitspunkt zur Verfügung stehen und damit Nichtlinearitäten weiterhin berücksichtigt werden können. Die sich so ergebenden Blockschaltbilder sind in Bild 5.29 dargestellt.

Die Konvergenz des mittleren quadratischen Fehlers bei diesem Modell verläuft deutlich langsamer als bei dem Modell 1. Auch scheint der zu erzielende minimale Fehler kleiner. Deshalb wurde der Lernvorgang nach 800 Schritten abgebrochen. Der Fehlerverlauf ist in Bild 5.30 zu sehen. In Tabelle 5.7 werden die Fehler für das Trainingssignal und die Testsignale angegeben. Die Ergebnisse sind deutlich schlechter als bei Modell 1. Auch findet sich keine Verbesserung bei rekursiver Auswertung. Dieses Ergebnis ist überraschend, da das Modell wegen der anders gewählten Eingangs eine bessere Leistung erwarten ließ.

Eine Erklärung für diese Resultate konnte nicht gefunden werden.



**Bild 5.29:** Struktur von Modell 5



**Bild 5.30:** Verlauf des mittleren quadratischen Fehlers, Modell 5

**Tabelle 5.7:** Mittlerer quadratischer Fehler für verschiedene Signale, Modell 5

	$q_m$ (Einschritt)	$q_m$ (rekursiv)
Trainingssignal	0,0394	3,1085
Testsignal 1	0,0418	3,2386
Testsignal 2	0,0496	0,8526
Testsignal 3	0,0411	1,4896
Testsignal 4	0,0903	0,5221
Testsignal 5	0,0419	12,7961
Testsignal 6	0,0477	8,1002

## 6 Zusammenfassung und Ausblick

In diesem Bericht wird die Fuzzy-Modellbildung eines technischen Systems, bestehend aus einem Hydrospeicher und einer Drossel, beschrieben. Die fünf betrachteten Modellstrukturen unterscheiden sich durch die Wahl ihrer Ein- und Ausgangsgrößen. Die zu identifizierenden Parameter der verwendeten Fuzzy-Relationalmodelle sind die Gewichte der einzelnen Regeln (Pedrycz 1984). Zu ihrer Bestimmung wurde das Verfahren der stochastischen Approximation (Tsyplkin 1969), auch stochastisches Gradientenverfahren genannt (Ljung und Söderström 1983), verwendet. Dieses Verfahren arbeitet rekursiv, ändert also die Modellparameter nach jedem Meßwert und ist damit on-line fähig. Das Trainingssignal war eine Folge verschiedener Sprünge, die einen weiten Bereich des Eingangsraumes abdecken. Nach der Identifikation wurden die Ergebnisse mittels mehrerer Testsignale verifiziert.

Es konnte gezeigt werden, daß die verwendeten Modellstrukturen grundsätzlich das Systemverhalten gut beschreiben und die Parameter geeignet identifiziert werden. Die Modellstruktur zeichnet sich mit nur zwei Eingängen durch ihre Einfachheit aus. Die Modellgüte ist sogar gut genug, um von einer Einschrittprädiktion, wie sie bei Bertram und Schwarz (1993) verwendet wird, auf die rekursive Auswertung der Modellgleichungen überzugehen. Damit ist es möglich, das Modell ohne das parallel laufende reale System auszuwerten. Dies ist der erste Schritt zum Einsatz von Fuzzy-Modellen in der Simulation. Insbesondere Echtzeitsimulationen für Anwendungen von Hardware-in-the-loop erscheinen möglich. Dazu ist allerdings eine weitere Beschleunigung der Fuzzy-Simulation notwendig. Diese soll in weiteren Arbeiten untersucht werden.

Als lineares Vergleichssystem wurde ein  $P_{T_1}$ -System mit dem LS-Verfahren identifiziert. Wie erwartet, zeigte jedoch das Fuzzy-Modell eine deutlich bessere Übereinstimmung mit dem realen System. Dennoch konnte das  $P_{T_1}$ -System dazu verwendet werden, das Fuzzy-Modell vorzubelegen, um die Konvergenzgeschwindigkeit zu erhöhen. Dazu wurde ein Verfahren vorgestellt, mit dem die Gewichte für ein lineares Fuzzy-Modell mit zwei Eingängen bestimmt werden können.

Eine weitere Möglichkeit zur Nutzung linearer Systeme ist in Bild 6.1 dargestellt. Bei diesem Hybrid-System wird zunächst ein lineares Modell der Strecke als eine erste Näherung bestimmt. Die dann noch verbleibenden relativ kleinen Abweichungen beschreibt

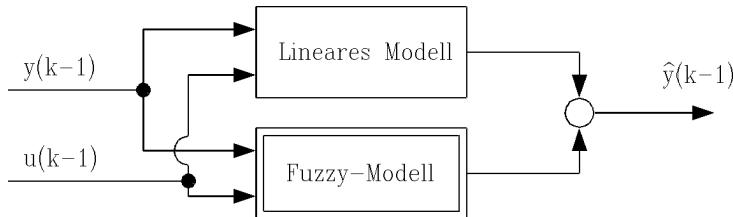
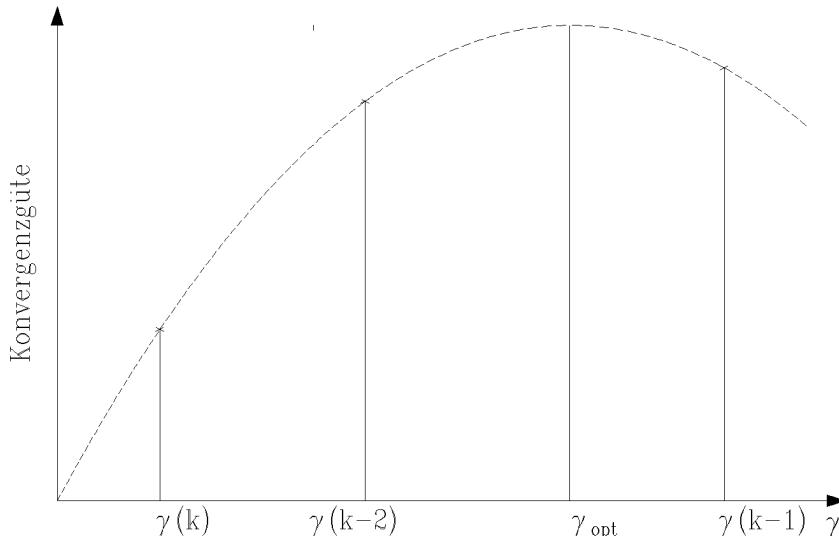


Bild 6.1: Hybrides Modell

ein Fuzzy-Modell. Zu dessen Bestimmung eignet sich das in diesem Bericht beschriebene

Verfahren. Eine weitere Verbesserung der Modelle läßt sich durch die Verwendung von zusätzlichen Regeln erreichen. Sie könnten etwa physikalische Gesetzmäßigkeiten oder auch Expertenwissen enthalten, z. B. „Der Systemdruck kann nicht über dem Versorgungsdruck liegen“ oder „Der Druck kann nicht negativ werden“.

Auch das Konvergenzverhalten des benutzten stochastischen Gradientenverfahrens muß noch verbessert werden. Dabei ist die optimale Wahl des Lernfaktors  $\gamma$  entscheidend. Obwohl in diesem Bericht  $\gamma$  für alle Fälle konstant war, ist eine Optimierung für jedes einzelne Modell sicherlich ratsam. Dies wurde bereits in Abschnitt 5.4.2 deutlich, wo die Abhängigkeit des Konvergenzverhaltens von der Startparametern bzw. dem Lernfaktor aufgezeigt werden konnte. Denkbar ist sowohl eine a-priori Wahl des optimalen Lernfaktors, etwa anhand einiger Testläufe, als auch eine Adaption, d. h. einer kontinuierlichen Anpassung des Lernfaktors. Letzteres kann entweder nach jedem einzelnen Meßwert oder nach einer gewissen Zahl von Meßwerten, also etwa nach jedem Abarbeiten der Trainingsdaten erfolgen. Ein möglicher Algorithmus zur Anpassung des Lernfaktors ist in Bild 6.2 dargestellt: Es werden die Konvergenzraten der letzten drei Lernfaktoren ermittelt. Daraus kann dann etwa mittels einer quadratischen Funktion ein optimaler Lernfaktor ermittelt werden. Diese einfachen Rechnungen sind auch on-line möglich. Ein Test dieses Verfahrens steht noch aus. Es muß auch untersucht werden, ob die bisher ausschließlich betrachtete Konvergenz des mittleren quadratischen Fehlers ein geeignetes Maß ist. Es sollte auch die Änderung der einzelnen Gewichte der Relationalmatrix betrachtet werden, was wegen ihrer großen Zahl allerdings Schwierigkeiten bereitet. In diesem Zusammenhang ist auch interessant,



**Bild 6.2:** Optimale Bestimmung des Lernfaktors

daß der verwendete Lernalgorithmus Ähnlichkeiten zu den im Bereich der künstlichen Neuronalen Netze verwendeten Algorithmen hat. Diese zeigt z.B. Suprijadi (1994) auf. So lassen sich in diesem Forschungsgebiet sicherlich viele Arbeiten zu Konvergenzverhalten und -verbesserung finden. Dies ist eine der Motivation, die zur Zusammenführung beider Gebiete zur Neuro-Fuzzy-Logik führten.

Ein weiterer Vorteil der Neuronalen Netze liegt in deren massiv parallelen Struktur mit relativ einfachen Rechenoperationen. Dadurch sind insbesondere auf dedizierten Hardwarestrukturen extrem hohe Verarbeitungsgeschwindigkeiten möglich, die unter dem Aspekt der Echtzeitanwendung von Fuzzy-Logik wichtig sind. Ein weiteres wichtiges Gebiet ist die Frage nach der Stabilität der gefundenen Modelle. Wie bereits angesprochen, wurden die Fuzzy-Modelle durch die Rückkopplung der Ausgangsgrößen zu dynamischen Systemen, genauer zu zeitdiskreten, nichtlinearen dynamischen Systemen. Es muß in Zukunft geklärt werden, ob und unter welchen Umständen ihre Stabilität gesichert ist. Gerade bei Anwendungen von Hardware-in-the-loop kann diese Frage extrem kritisch sein.

Die Ergebnisse dieses Berichtes entstanden im Rahmen des Projektes „Echtzeitsimulation mittels Fuzzy-Systemmodellen“, das vom Ministerium für Wissenschaft und Forschung des Landes Nordrhein-Westfalen gefördert wurde.

## 7 Literatur

- Angenendt, G.** 1993. *Identifikation eines selbsteinstellenden Fuzzy-Modells*. Diplomarbeit MSRT. Universität –GH– Duisburg.
- Bertram, T.** 1991. *Einführung in die Fuzzy-Regelung*. Forschungsbericht Nr. 4/91 MSRT. Universität –GH– Duisburg.
- Bertram, T., K. Küpper and H. Schwarz.** 1993. Fuzzy modelling for the control of mechatronic systems. *Proceedings of the 2nd Conference on Mechatronics and Robotics, Duisburg/Moers*. 86-106.
- Bertram, T. und H. Schwarz.** 1993. Fuzzy identification of hydraulic systems. *12th IFAC World Congress*. Sydney, Australia. Vol. VII. 489-492.
- Bertram, T. u.a.** 1994. Fuzzy Control. Zusammenstellung und Beschreibung wichtiger Begriffe. *Automatisierungstechnik* **42**.
- Bezdek, J.** 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press.
- Böhme, G.** 1990. *Algebra*. Berlin: Springer.
- Brown, M. and C.J. Harris.** 1991. A nonlinear adaptive controller: a comparison between fuzzy logic control and neurocontrol. *IMA Journal of Mathematical Control and Information* **3**. 239-265.
- Driankow, D., H. Hellendoorn and M. Reinfrank.** 1993. *An Introduction to Fuzzy Control*. Berlin Heidelberg: Springer-Verlag.
- Glorennec, P.Y.** 1991. Adaptive fuzzy control. *IFSA '91*. 33-36.
- Isermann, R.** 1988. *Identifikation dynamischer Systeme I, II*. Berlin: Springer.
- Jelali, M.** 1994. *Zur Modellbildung nichtlinearer Prozesse durch quadratische Systeme mit linearer Steuerung (QLS)*. Forschungsbericht Nr. 5/94 MSRT. Universität –GH– Duisburg.
- Kosko, B.** 1992. Fuzzy systems as universal approximators. *Proc. IEEE International Conference on Fuzzy Systems, San Diego*. 1153-1163.
- Küpper, K.** 1993a. *Numerisch optimierte Fuzzy-Modelle*. Forschungsbericht Nr. 5/93 MSRT. Universität –GH– Duisburg.
- Küpper, K.** 1993b. *Fuzzy-Modellierung eines Gasofens*. Forschungsbericht Nr. 7/93 MSRT, Universität –GH– Duisburg.

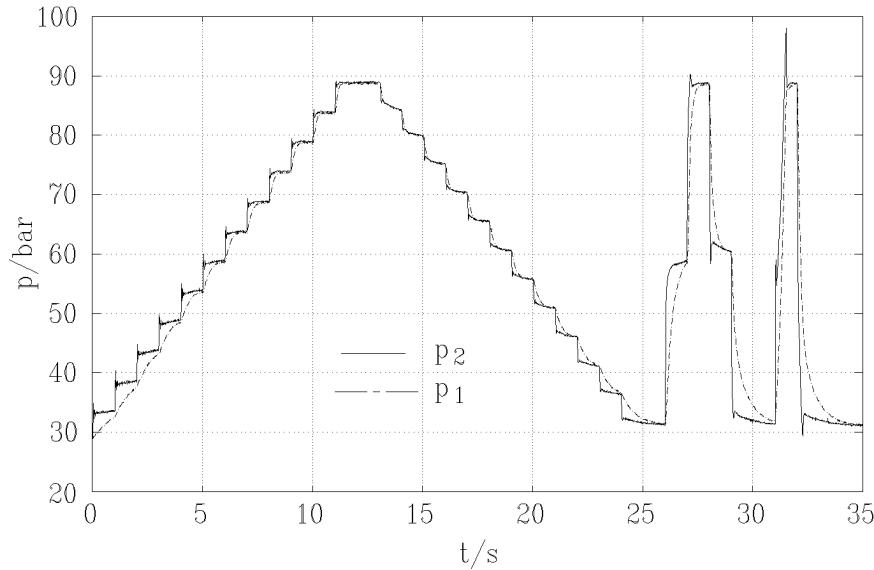
- Larsen, P. M.** 1980. Industrial applications of fuzzy logic control. *International Journal of Man-Machine-Studies* 12. 3-10.
- Lee, C. C.** 1990. Fuzzy logic in control systems: fuzzy logic controller. *IEEE Transactions on Systems, Man, and Cybernetics* 20. 404-418 und 419-435.
- Liu, M.-H.** 1993. Fuzzy-Modellbildung für automatisiertes Roboterentgraten. *3. Workshop Fuzzy Control GMA-UA 1.4.2 Dortmund*. 141-156.
- Ljung, L.** 1974. *Convergence of Recursive Stochastic Algorithms*. Report 7403. Division of Automatic Control. Lund Institute of Technology.
- Ljung, L. and T. Söderström.** 1987. *Theory and Practice of Recursive Identification*. 1st Paperback edition. Cambridge, MA.: The MIT Press.
- Meyer-Gramann, K.D. and E.-W. Jüngst.** 1993. Fuzzy Control - schnell und kostengünstig implementiert mit Standard-Hardware. *Automatisierungstechnik* 41. 166-172.
- Mizumoto, M.** 1987. Min-max-gravity method versus product-sum-gravity method for fuzzy controls. *Proceedings of the fourth IFSA World Congress*. Brüssel/Belgien. Engineering 127-130.
- Moore, C. G. and C. J. Harris.** 1992. Indirect adaptive fuzzy control. *International Journal of Control* 56. 441-468.
- Pedrycz, W.** 1983. Numerical and applicational aspects of fuzzy relational equations. *Fuzzy Sets and Systems* 11. 1-18.
- Pedrycz, W.** 1984a. An identification algorithm in fuzzy relational systems. *Fuzzy Sets and Systems* 13. 153-167.
- Pedrycz, W.** 1984b. Identification in fuzzy systems. *IEEE Transactions on Systems, Man, and Cybernetics*. 14. 361-366.
- Pedrycz, W.** 1985. Design of fuzzy control algorithms with the aid of fuzzy models. *Industrial Applications of Fuzzy Control*. Hrsg. M. Sugeno. Amsterdam: North-Holland.
- Pedrycz, W.** 1989. *Fuzzy Control and Fuzzy Systems*. Taunton: Research Studies Press.
- Pedrycz, W.** 1991. Fuzzy modeling: Fundamentals, construction, and evaluation. *Fuzzy Sets and Systems* 41. 1-15.
- Pfeiffer, B.-M.** 1993. Identifikation von Fuzzy-Reglern aus Lerndaten. *3. Workshop Fuzzy Control GMA-UA 1.4.2 Dortmund*. 238-250.
- Postlethwaite, B.** 1991. Empirical comparison of methods of fuzzy relational identification. *IEE Proceedings Part D*, Vol. 138, No. 3. 199-206.

- Reuter, H.** 1993a. *Zur Identifikation bilinierarer Modelle in kanonischer Form*. Forschungsbericht Nr. 8/93 MSRT. Universität –GH– Duisburg.
- Reuter, H.** 1993b. *Ein neues Verfahren zur Abschätzung einer günstigen Abtastzeit mittels Entropieanalyse*. Forschungsbericht Nr. 9/93 MSRT. Universität –GH– Duisburg.
- van der Rhee, F. and H. R. van der Nauta Lemke.** 1987. Applying fuzzy set theory to modeling processes. *10th IFAC World Congress on Automatic Control*, München/Germany. 338-343.
- van der Rhee, F. and H. R. van der Nauta Lemke.** 1990. Knowledge based fuzzy modeling of systems. *11th IFAC World Congress on Automatic Control, Tallin/USSR*. 186-191.
- Schwarz, H.** 1991. Nichtlineare Regelungssysteme: Systemtheoretische Grundlagen. München: Oldenbourg.
- Sugeno, M. and G. T. Kang.** 1986. Fuzzy modeling and control of multilayer incinerator. *Fuzzy Sets and Systems* 18. 329-345.
- Sugeno, M. and G. T. Kang.** 1988. Structure identification of fuzzy models. *Fuzzy Sets and Systems* 28. 15-33.
- Sugeno, M. und K. Tanaka.** 1991. Successive identification of a fuzzy model and its applications to prediction of a complex system. *Fuzzy Sets and Systems* 42. 315-334.
- Sugeno, M. und T. Yasukawa.** 1993. A fuzzy logic based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems* 1. 1-34.
- Suprijadi, E.B.** 1994. *Identifikation von Fuzzy-Modellen mittels stochastischer Approximation*. Studienarbeit MSRT. Universität –GH– Duisburg.
- Takagi, T. and M. Sugeno.** 1983. Derivation of fuzzy control rules from human operator's control actions. *IFAC Symposium Fuzzy Information, Knowledge Representation and Decision Analysis, Marseille*. 55-60.
- Takagi, T. and M. Sugeno.** 1985. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* 15. 116-132.
- Tong, R. M.** 1978. Synthesis of fuzzy models for industrial processes-some recent results. *International Journal of General Systems* 4. 142–162.
- Tong, R. M.** 1980. The evaluation of fuzzy models derived from experimental data. *Fuzzy Sets and Systems* 4. 1-12.

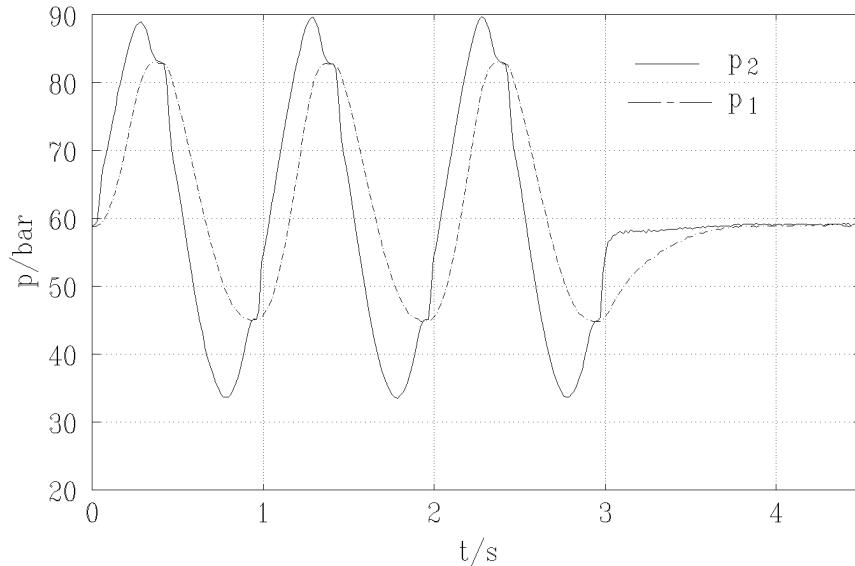
- Tsyplkin, Y. Z.** 1971. *Adaption and Learning in Automatic Systems*. New York, London: Academic Press.
- Unbehauen, H.** 1988. *Regelungstechnik III*. Braunschweig: Vieweg.
- Wang, X. L.** 1992. Fuzzy systems are universal approximators. *Proc. IEEE International Conference on Fuzzy Systems, San Diego*. 1163-1170.
- Wang, L. X.** 1994. *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Englewood Cliffs, NJ: Prentice Hall.
- Wang, L. X. and J.M. Mendel.** 1993. Fuzzy adaptive filters, with application to non-linear channel equalization. *IEEE Transactions on Fuzzy Systems, Vol. 1*. 161-170.
- Xu, C.-W.** 1989. Fuzzy system identification. *IEE Proceedings 136*. 146-150.
- Xu, C.-W. and Y. Z. Lu.** 1987. Fuzzy model identification and self-learning for dynamic systems. *IEEE Transactions on Systems, Man, and Cybernetics 17*. 683-689.
- Yin, X.** 1994. *Zur Identifikation zeitkontinuierlicher nichtlinearer Systeme*. Dissertation. Universität –GH– Duisburg. VDI Verlag. Fortschrittsberichte. Reihe 8. Nr. 385. Düsseldorf: VDI.
- Zadeh, L.** 1973. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics 3*. 28-44.

## A Testsignale

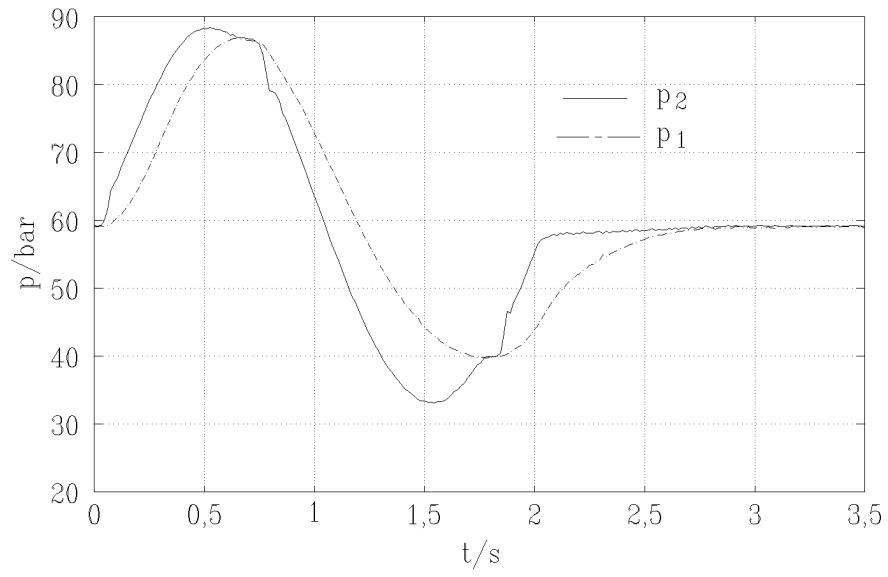
In diesem Anhang sind die Testsignale abgebildet, die zur Verifizierung der Fuzzy-Modelle benutzt wurden.



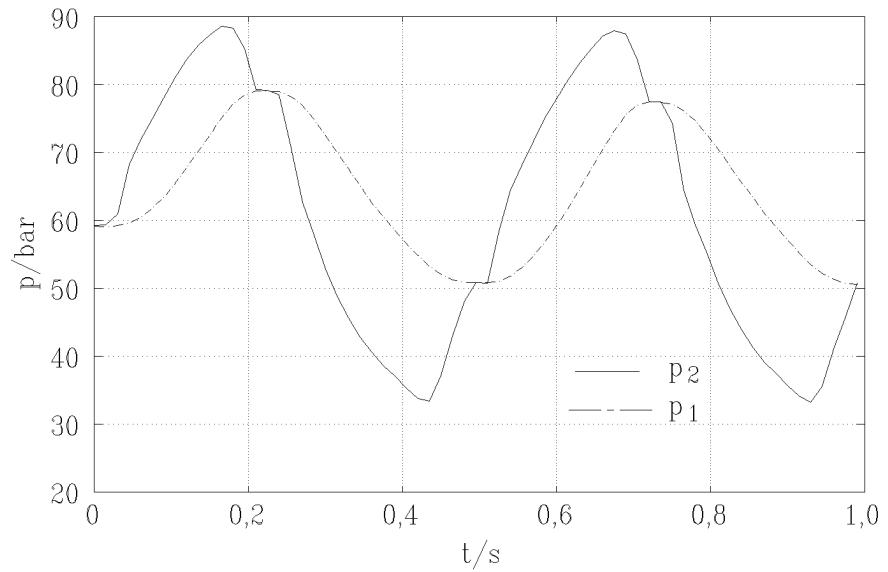
**Bild A.1:** Testsignal 1



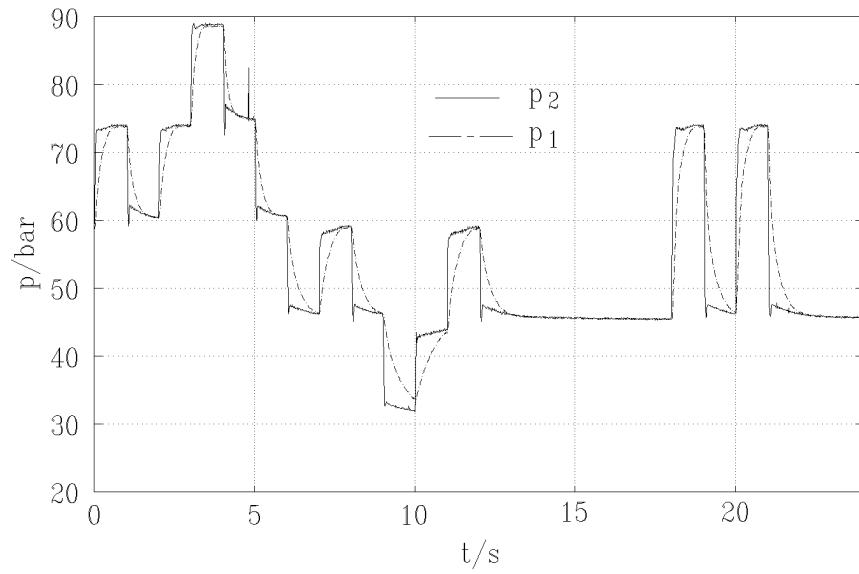
**Bild A.2:** Testsignal 2



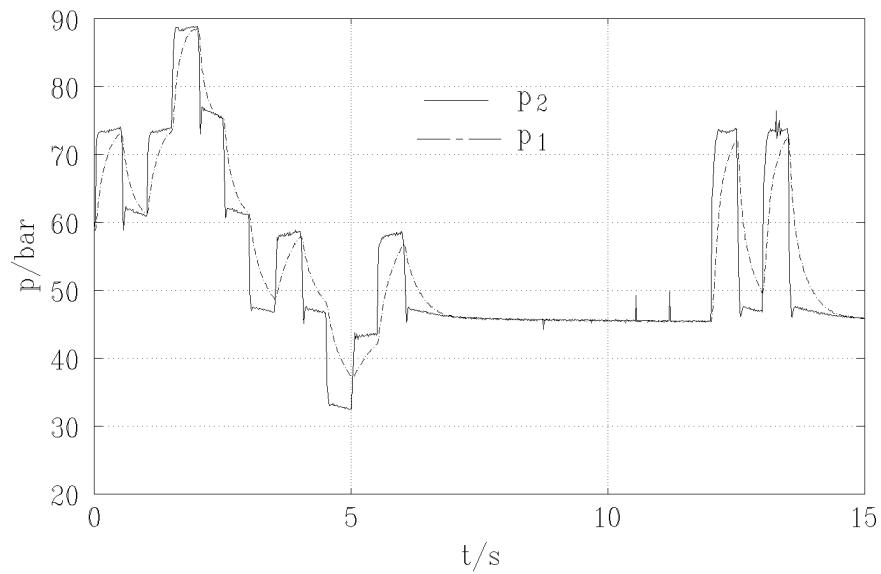
**Bild A.3:** Testsignal 3



**Bild A.4:** Testsignal 4



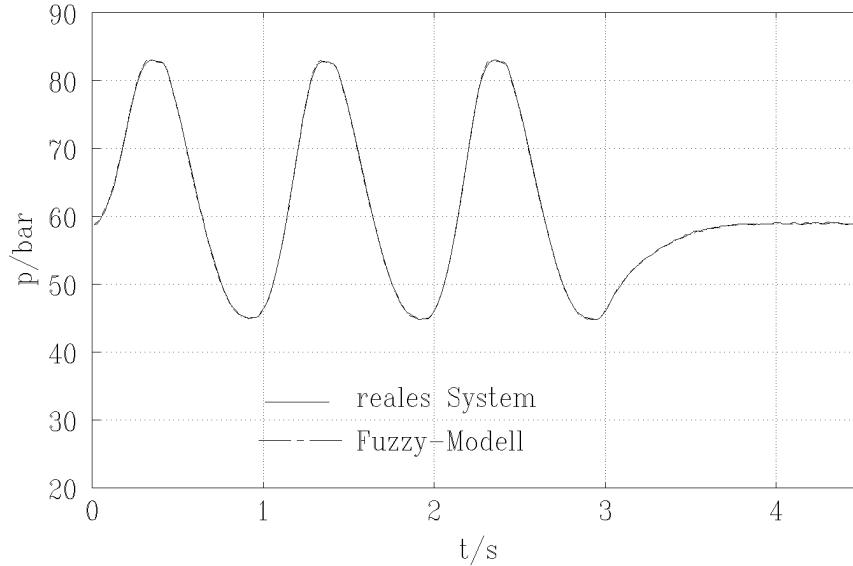
**Bild A.5:** Testsignal 5



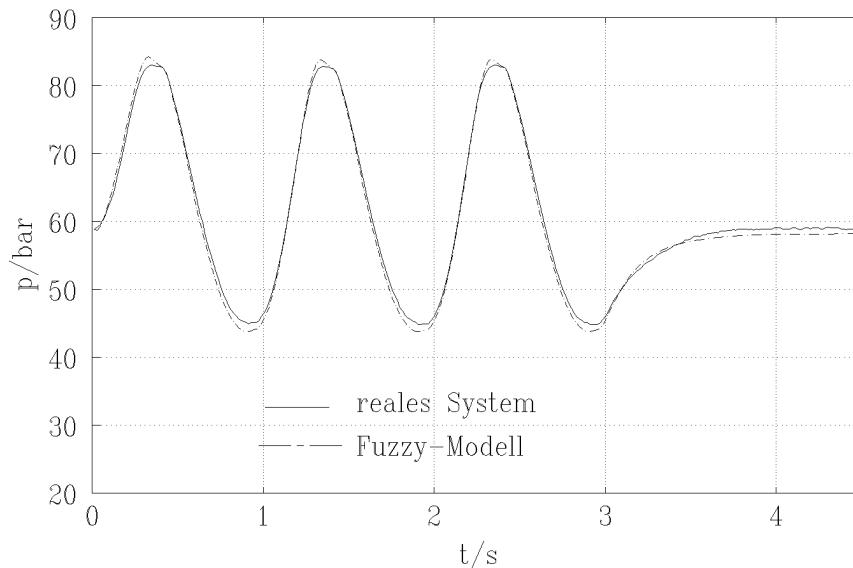
**Bild A.6:** Testsignal 6

## B Ergebnisse für Modell 1

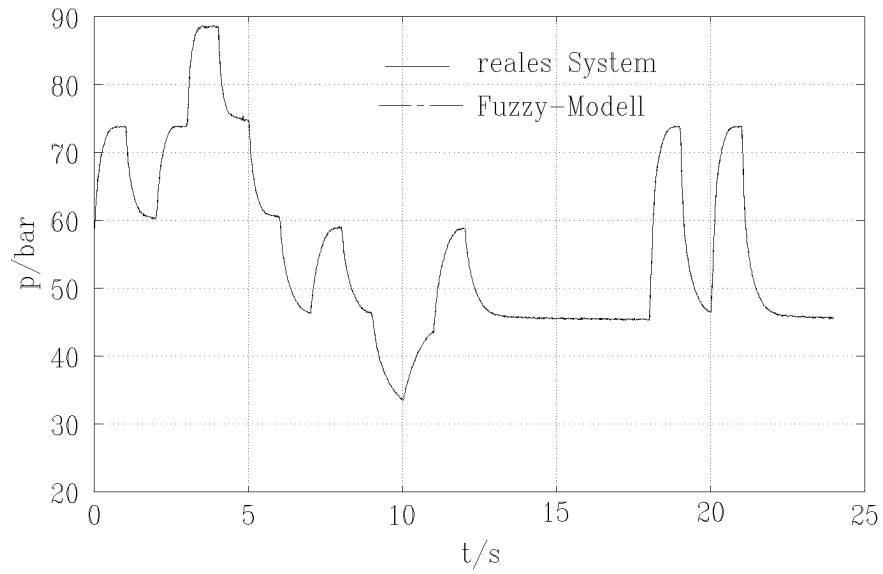
In diesem Anhang werden die Ergebnisse des Modells 1 für die verschiedenen Testsignale und Auswertungsarten gezeigt.



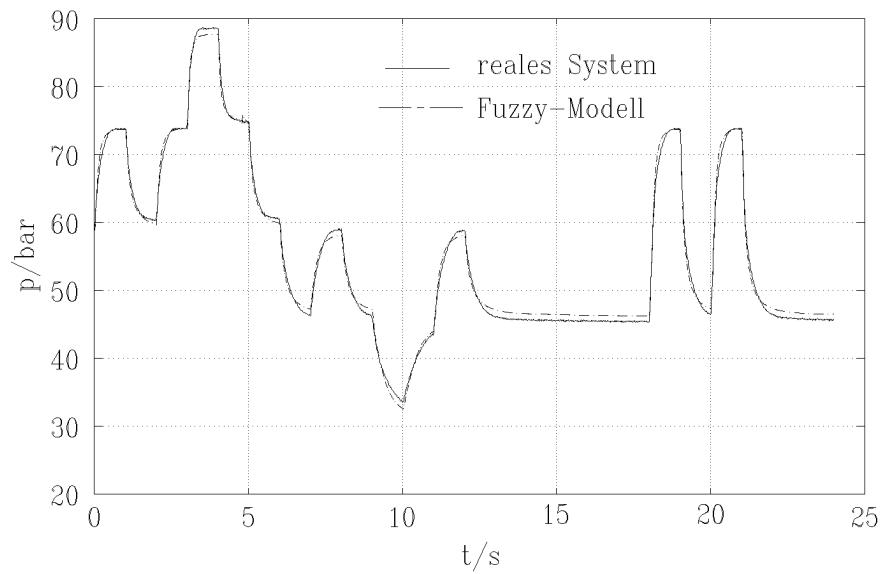
**Bild B.1:** Testsignal 2, Einschrittprädiktion



**Bild B.2:** Testsignal 2, rekursive Auswertung



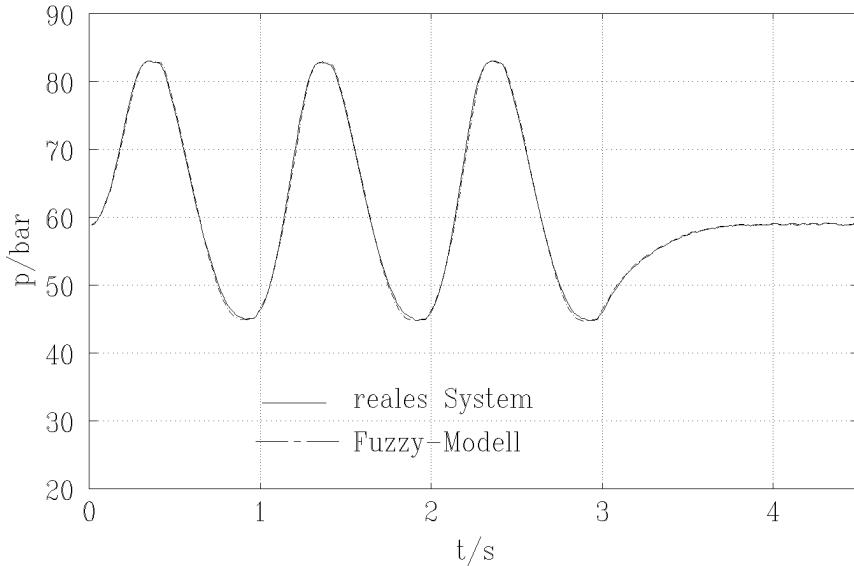
**Bild B.3:** Testsignal 5, Einschrittprädiktion



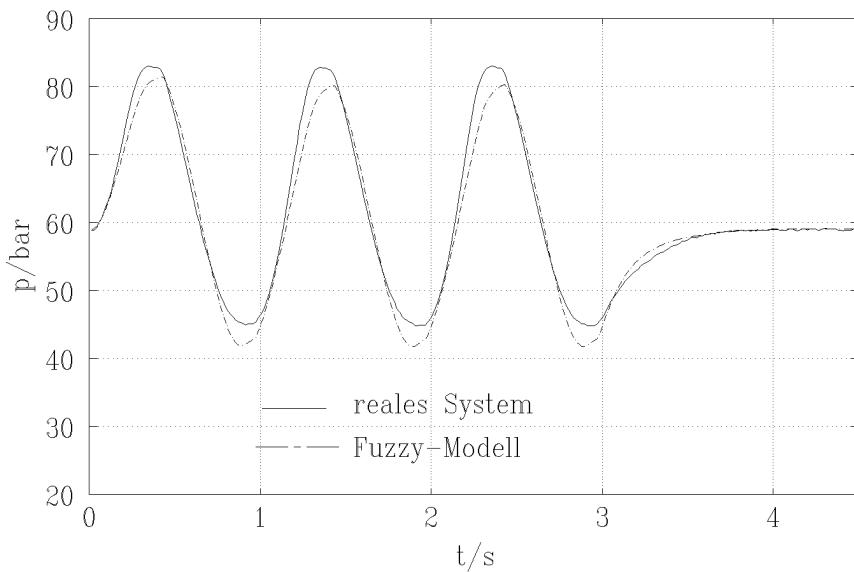
**Bild B.4:** Testsignal 5, rekursive Auswertung

## C Ergebnisse für die Modellierung mit einem $P_{T1}$ -System

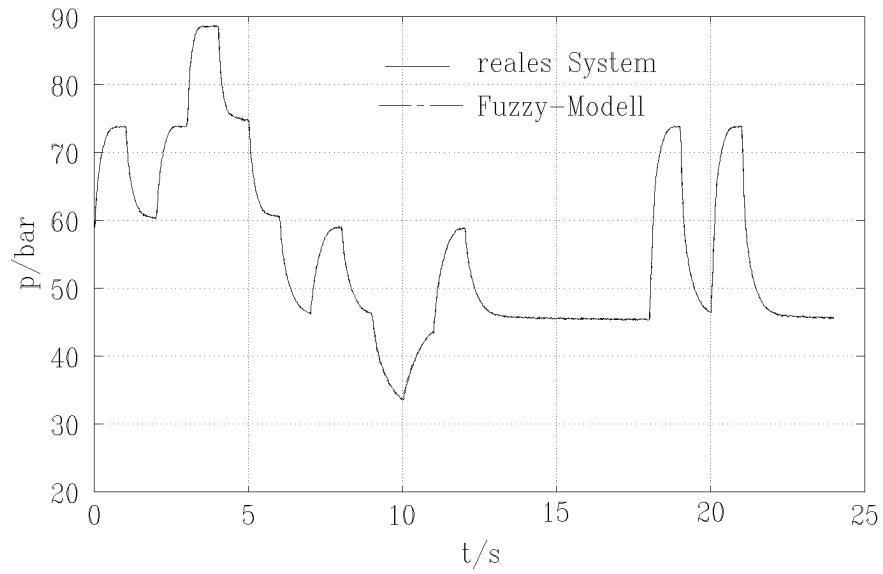
In diesem Anhang werden die Ergebnisse gezeigt, die für die verschiedenen Testsignale und Auswertungsarten bei der Modellierung der Strecke als  $P_{T1}$ -System erzielt wurden.



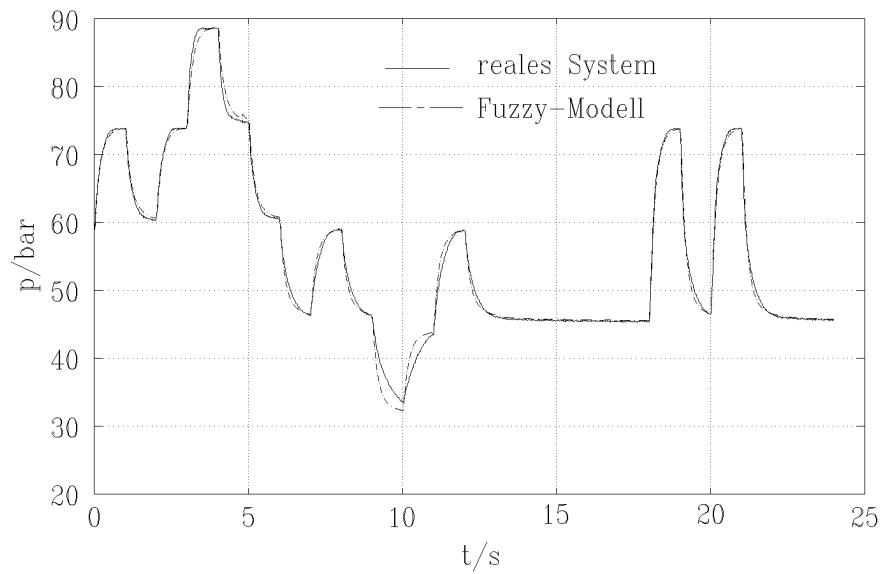
**Bild C.1:** Testsignal 2, Einschrittprädiktion



**Bild C.2:** Testsignal 2, rekursive Auswertung



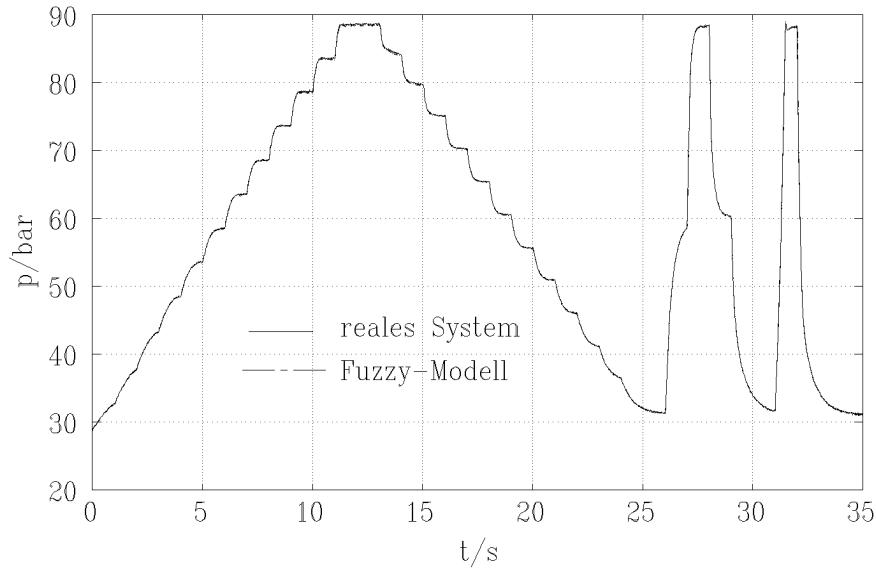
**Bild C.3:** Testsignal 5, Einschrittprädiktion



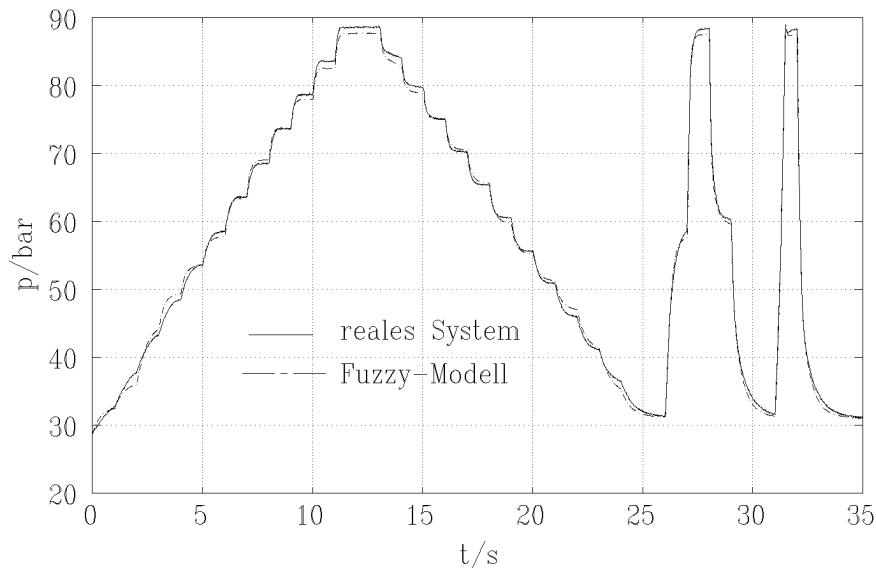
**Bild C.4:** Testsignal 5, rekursive Auswertung

## D Ergebnisse für Modell 2

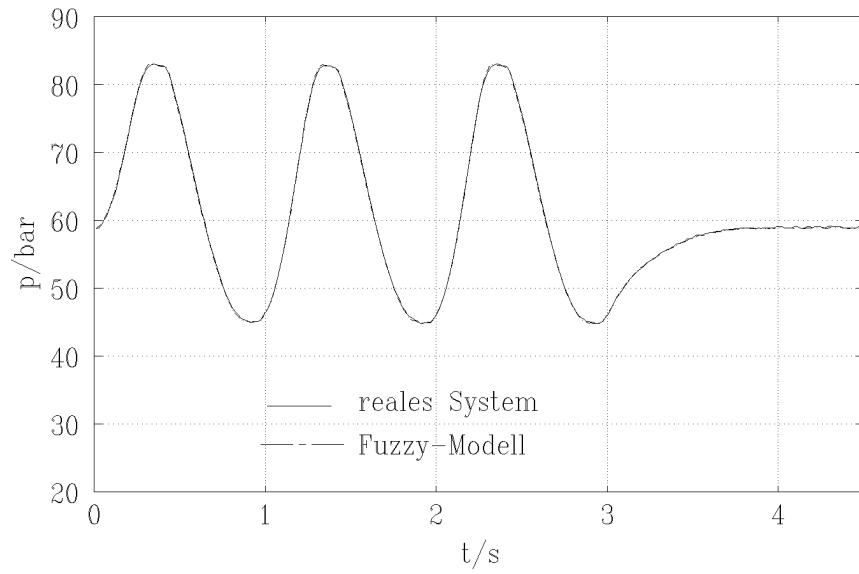
In diesem Anhang werden die Ergebnisse des Modells 2 für die verschiedenen Signale und Auswertungsarten gezeigt.



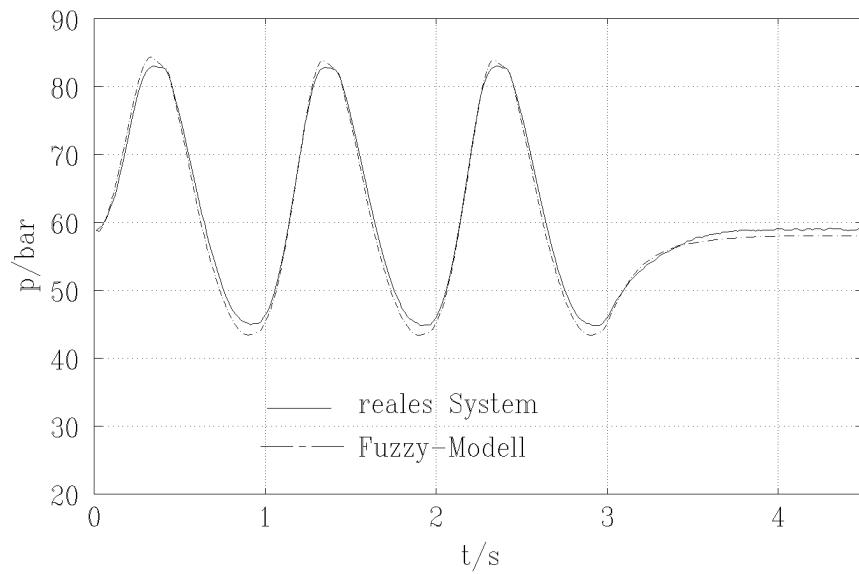
**Bild D.1:** Trainingssignal, Einschrittprädiktion



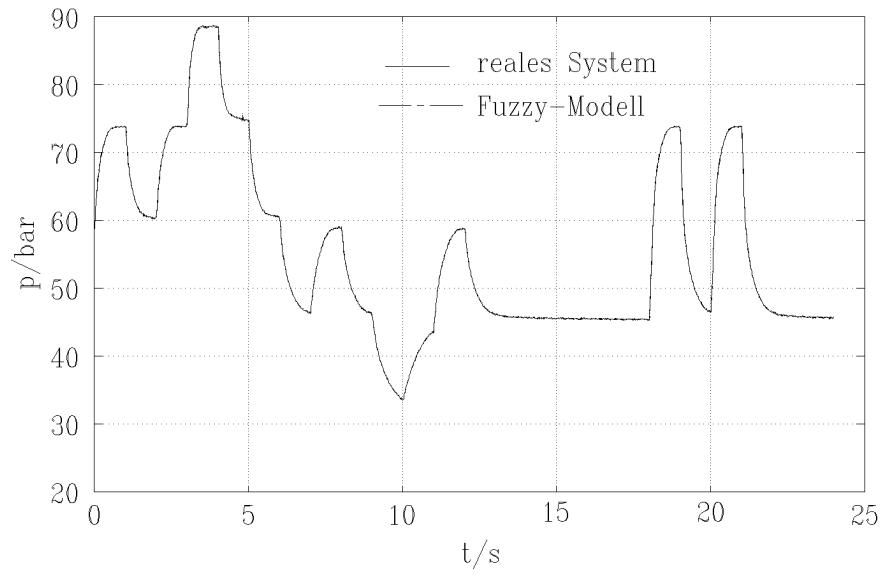
**Bild D.2:** Trainingssignal, rekursive Auswertung



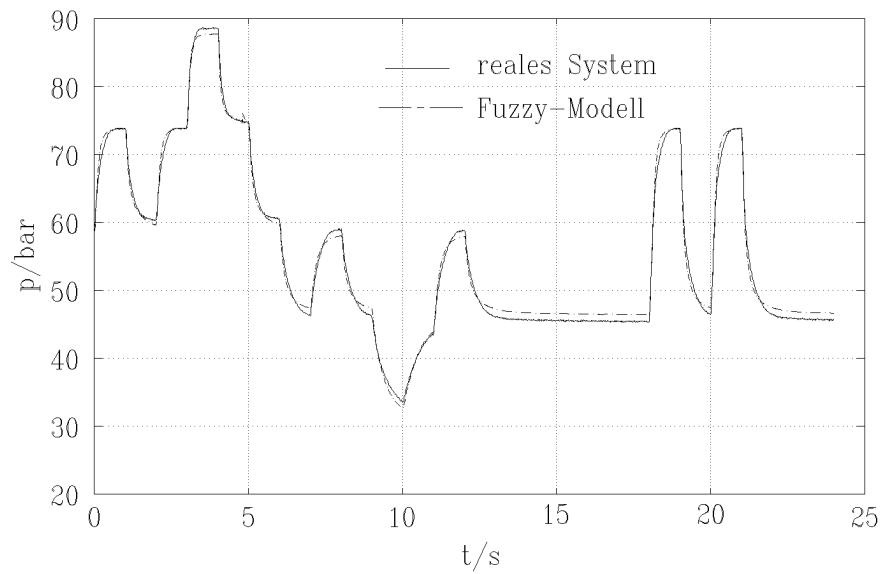
**Bild D.3:** Testsignal 2, Einschrittprädiktion



**Bild D.4:** Testsignal 2, rekursive Auswertung



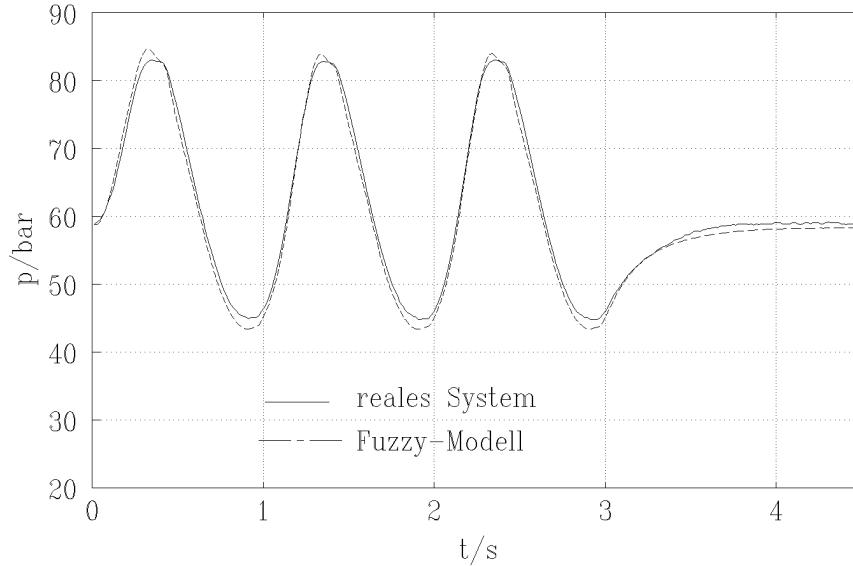
**Bild D.5:** Testsignal 5, Einschrittprädiktion



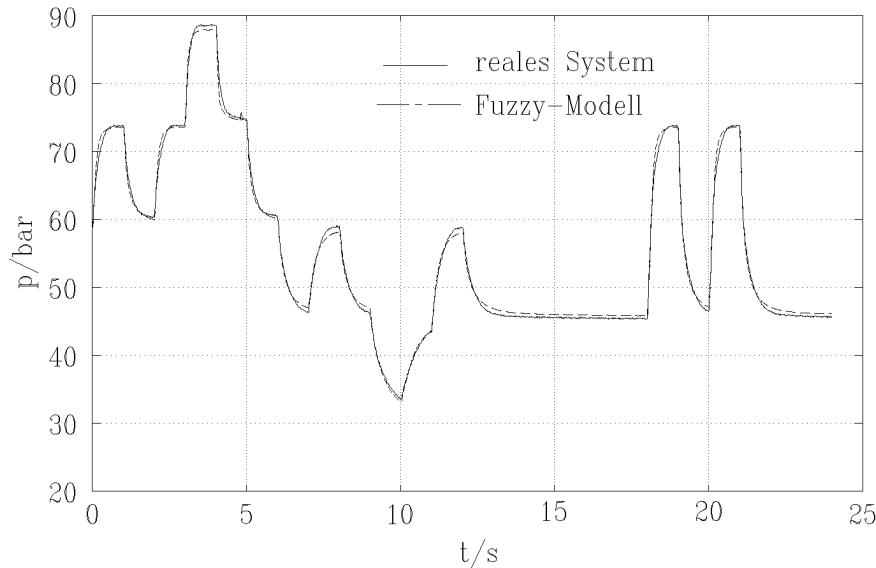
**Bild D.6:** Testsignal 5, rekursive Auswertung

## E Ergebnisse für das inkrementale Modell

In diesem Anhang werden die Ergebnisse des inkrementalen Modells für die verschiedenen Testsignale bei rekursiver Auswertung gezeigt.



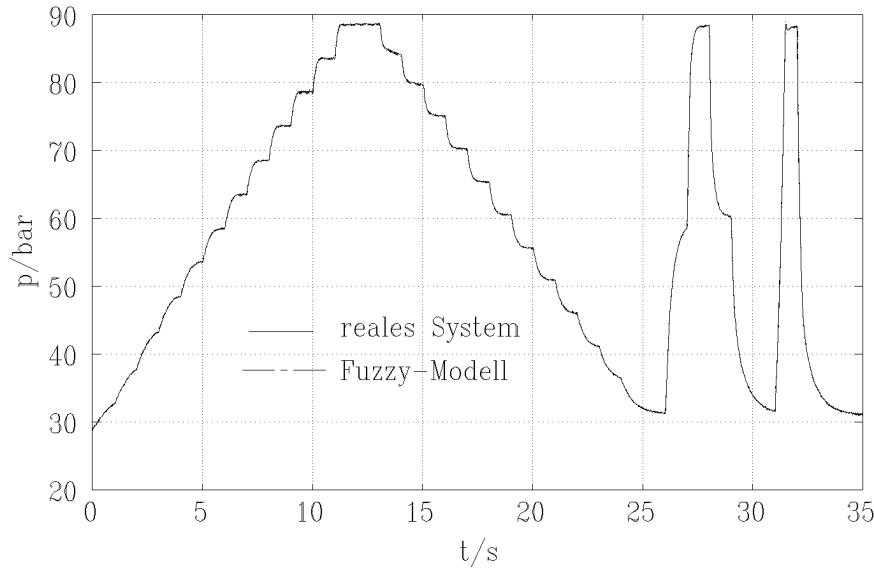
**Bild E.1:** Testsignal 2, rekursive Auswertung



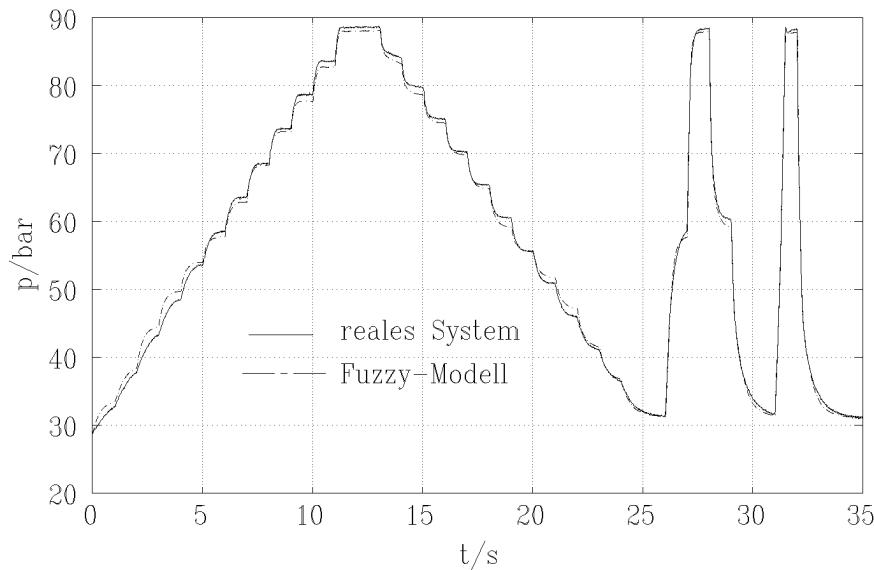
**Bild E.2:** Testsignal 5, rekursive Auswertung

## F Ergebnisse für Modell 4

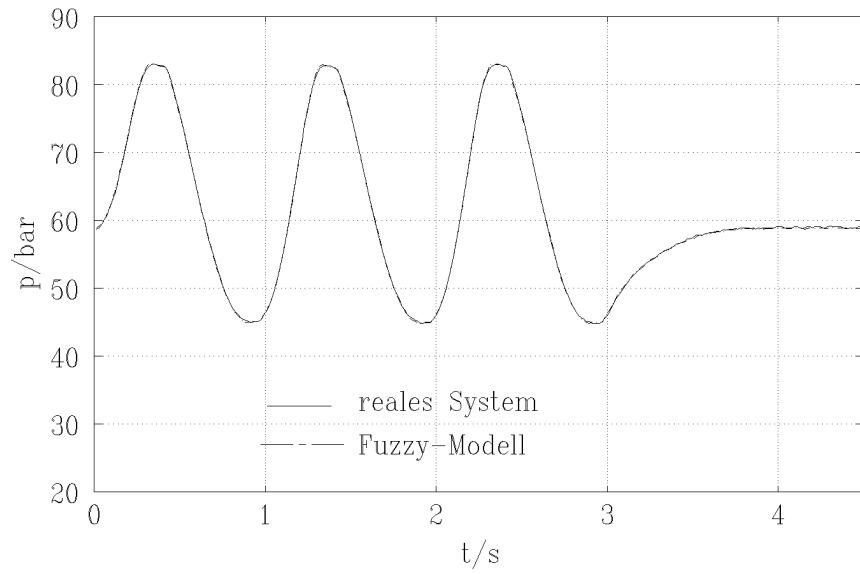
In diesem Anhang werden die Ergebnisse des Modells 4 für die verschiedenen Signale und Auswertungsarten gezeigt.



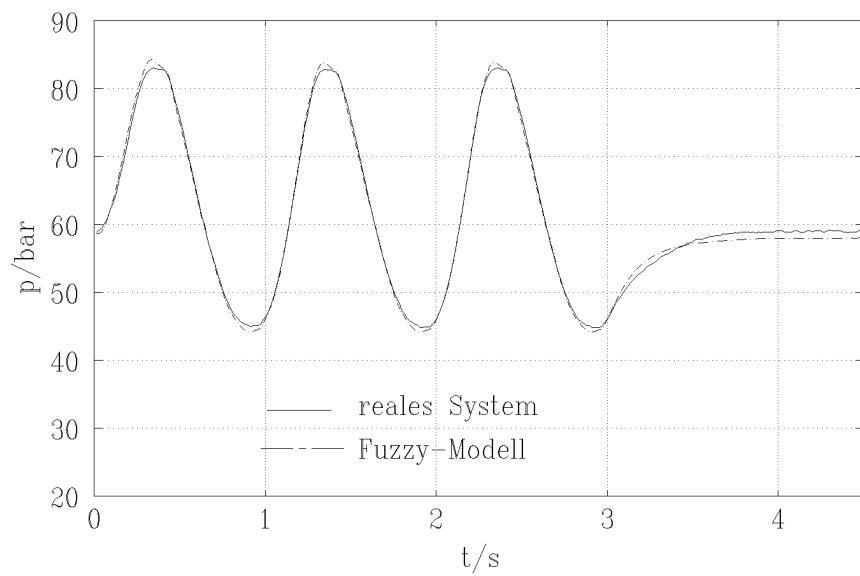
**Bild F.1:** Trainingssignal, Einschrittprädiktion



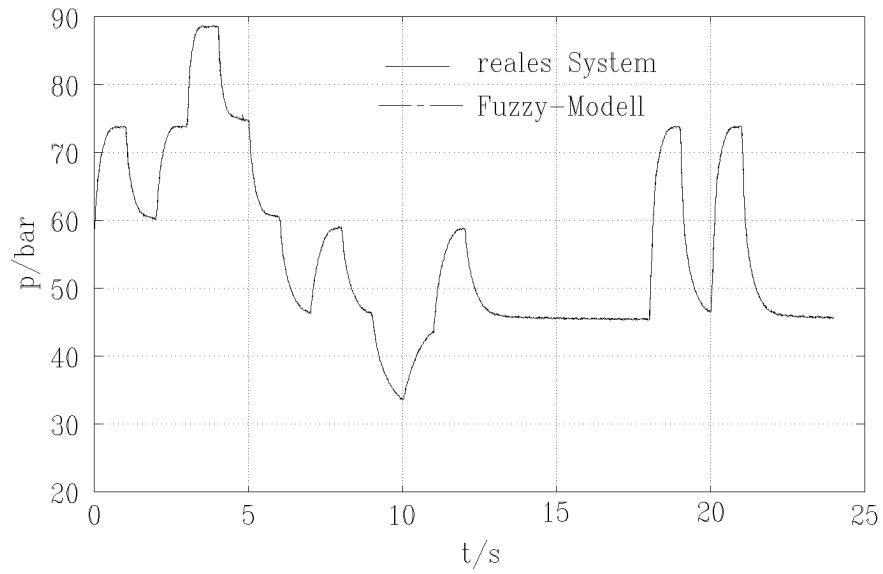
**Bild F.2:** Trainingssignal, rekursive Auswertung



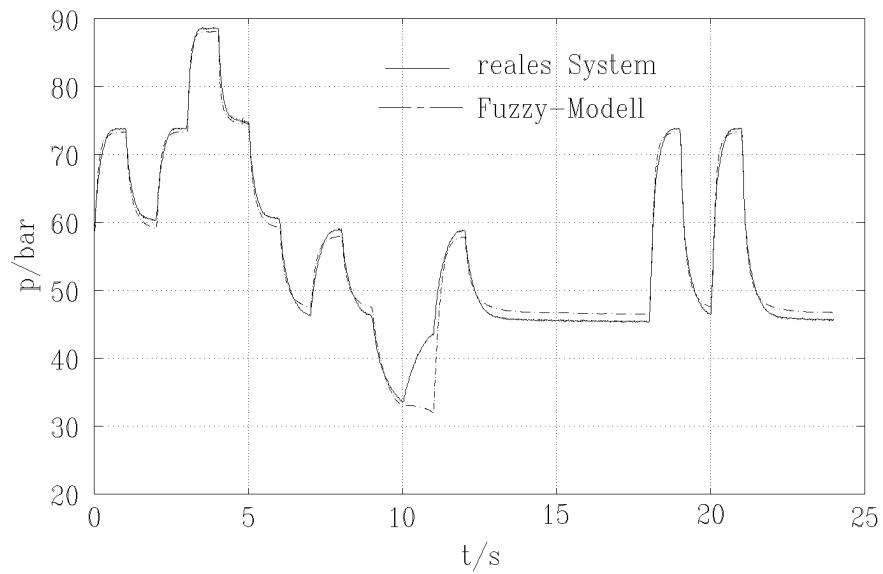
**Bild F.3:** Testsignal 2, Einschrittprädiktion



**Bild F.4:** Testsignal 2, rekursive Auswertung



**Bild F.5:** Testsignal 5, Einschrittprädiktion



**Bild F.6:** Testsignal 5, rekursive Auswertung