

# Modellierung dynamischer Systeme mit zeitkontinuierlichen rekurrenten Fuzzy-Systemen

## Modeling of Dynamical Systems Using Continuous-time Recurrent Fuzzy Systems

Jürgen Adamy und Andreas Schwung

In diesem Beitrag wird die Anwendung zeitkontinuierlicher rekurrenter Fuzzy-Systeme (KRFS) zur Modellbildung technischer Systeme beschrieben. Ausgehend von der Systembeschreibung rekurrenter Fuzzy-Systeme werden Verfahren vorgestellt, mit denen das KRFS das Verhalten des Prozesses anhand von Messdaten und qualitativem Prozesswissen erlernen kann. Die Leistungsfähigkeit wird anhand der Modellierung eines biotechnologischen Prozesses gezeigt.

This paper presents the application of continuous-time recurrent fuzzy systems for modeling of dynamical systems. After a short introduction on recurrent fuzzy systems, we present methods for learning the dynamics using both measurement data and qualitative knowledge. The capability of the approach is shown by modeling of a biotechnological process.

**Schlagwörter:** Zeitkontinuierliche rekurrente Fuzzy Systeme, qualitative Modellierung dynamischer Systeme, Lernverfahren

**Keywords:** Continuous-time recurrent fuzzy systems, qualitative modeling of dynamical systems, learning algorithms

### 1 Einleitung

Seit ihrer Einführung werden die Theorie und die Anwendungsbereiche von Fuzzy-Systemen kontinuierlich erweitert. Eine vielbeachtete Anwendungsmöglichkeit der Fuzzy-Logik ist die Modellierung statischer und dynamischer Zusammenhänge. Die Verwendung von Fuzzy-Systemen zur Modellierung hat den Vorteil, dass linguistisches Vorwissen erfahrener Prozessanwender in den Modellbildungsprozess integriert werden kann. Somit eignen sich Fuzzy-Systeme insbesondere zur Modellierung sogenannter Grey-Box-Modelle, deren physikalische Beschreibung zwar nicht oder nur schwierig möglich ist, bei denen aber über reine Messdaten des Ein-/Ausgangsverhaltens hinaus qualitatives Wissen über das System vorhanden ist.

Eine weit verbreitete Möglichkeit zur Modellierung dynamischer Systeme mit Fuzzy-Logik geht auf Takagi, Sugeno und Kang (TSK) zurück [14]. Bei diesem Ansatz werden

Systemmodelle in  $i$  Arbeitspunkten in Form von Differentialgleichungen mithilfe von Fuzzy-Regeln gewichtet:

$$\begin{aligned} \text{Wenn } & x_1(t) = L_{j_1}^{x_1} \text{ und } \dots \text{ und } x_n(t) = L_{j_n}^{x_n} \\ \text{und } & u_1(t) = L_{q_1}^{u_1} \text{ und } \dots \text{ und } u_m(t) = L_{q_m}^{u_m}, \\ \text{dann } & \dot{\mathbf{x}} = \mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{u}. \end{aligned} \quad (1)$$

Ein gänzlich anderer Ansatz führt auf rekurrente Fuzzy-Systeme (RFS), bei denen die Systemdynamik linguistisch durch Regeln der Form

$$\begin{aligned} \text{Wenn } & x_1(t) = L_{j_1}^{x_1} \text{ und } \dots \text{ und } x_n(t) = L_{j_n}^{x_n} \\ \text{und } & u_1(t) = L_{q_1}^{u_1} \text{ und } \dots \text{ und } u_m(t) = L_{q_m}^{u_m}, \\ \text{dann } & \dot{x}_1(t) = L_{w_1}^{\dot{x}_1} \text{ und } \dots \text{ und } \dot{x}_n(t) = L_{w_n}^{\dot{x}_n} \end{aligned} \quad (2)$$

beschrieben wird. Hierbei sind  $L_{q_p}^{u_p}$ ,  $L_{j_i}^{x_i}$  und  $L_{w_i}^{\dot{x}_i}$  linguistische Werte der Eingänge, der Zustände und ihrer Ableitungen. Die Konklusion und damit die Zustandsableitungen

werden bei RFS im Gegensatz zu TSK-Fuzzy-Systemen aus linguistischen Werten und nicht aus Differentialgleichungen gebildet. Dies erlaubt eine Beschreibung der Systemdynamik allein auf Grundlage linguistischer Regeln.

RFS wurden zunächst für zeitdiskrete Systeme entwickelt [1; 3; 6; 9] und in der Mustererkennung industriell eingesetzt [1; 10]. In [2] werden RFS dann auf den Fall zeitkontinuierlicher Größen erweitert und systemtheoretische Eigenschaften wie Ruhelagen und Stabilität untersucht. In [5] wird außerdem am Beispiel einer Brennstoffzelle gezeigt, dass sich zeitkontinuierliche rekurrente Fuzzy-Systeme (KRFS) sehr gut zur Modellierung nichtlinearer Systeme eignen. Allerdings wurde die Modellbildung und Optimierung weitestgehend händisch durchgeführt.

Um die Modellierung zu automatisieren, müssen geeignete Optimierungsverfahren entwickelt werden, mit denen das KRFS an das zu modellierende System angepasst werden kann.

Hierbei kann das bereits vorhandene Wissen über das System in den Modellierungsprozess einbezogen werden. Im Regelfall wird dieses Prozesswissen qualitativer Natur sein, sodass nur die Struktur des Fuzzy-Systems vom Anwender vorgegeben werden kann. Die quantitative Ermittlung der Parameter des KRFS allein aus Prozesswissen ist dagegen nahezu unmöglich.

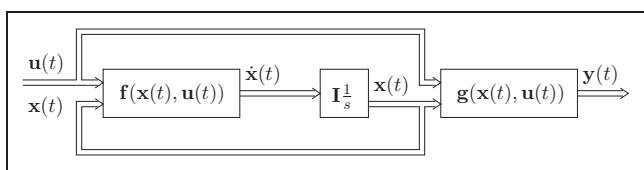
Deshalb werden im Folgenden einerseits Verfahren vorgestellt, die auf Basis der bereits vorgegebenen Struktur die Parameter des KRFS an das Systemverhalten anpassen. Andererseits kann bei geeigneter Auslegung des KRFS mit einem dieser Verfahren auch die Struktur, d. h. die Regelbasis, bestimmt werden.

Der Beitrag gliedert sich wie folgt: Abschnitt 2 stellt zunächst die Systembeschreibung kontinuierlicher rekurrenter Fuzzy-Systeme vor. In Abschnitt 3 werden dann Lernverfahren für KRFS entwickelt, deren Anwendung Abschnitt 4 anhand der Modellierung eines biotechnologischen Prozesses zeigt.

## 2 Zeitkontinuierliche rekurrente Fuzzy-Systeme

Zunächst soll eine kurze Einführung in KRFS gegeben werden. Eine ausführlichere Beschreibung findet sich in [2].

Die Struktur zeitkontinuierlicher rekurrenter Fuzzy-Systeme ist in Bild 1 dargestellt. Die Funktionen  $f(\mathbf{x}(t), \mathbf{u}(t))$  und  $\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t))$  sind jeweils vollständige Fuzzy-Systeme mit



**Bild 1:** Blockschaltbild eines zeitkontinuierlichen rekurrenten Fuzzy-Systems.

Fuzzifizierung, Inferenz und Defuzzifizierung. Somit nehmen sowohl die Eingangsvektoren  $\mathbf{u}(t)$  und  $\mathbf{x}(t)$  als auch die Ausgangsvektoren  $\dot{\mathbf{x}}(t)$  bzw.  $\mathbf{y}(t)$  crisper Werte an. Da die Ausgangsfunktion  $\mathbf{g}$  für die Dynamik keine Rolle spielt, wird sie im Weiteren ohne Beschränkung der Allgemeinheit vernachlässigt.

Die Dynamik des KRFS, die durch  $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$  gegeben ist, kann linguistisch durch Regeln der Form

$$\begin{aligned} \text{Wenn } & x_1(t) = L_{j_1}^{x_1} \text{ und } \dots \text{ und } x_n(t) = L_{j_n}^{x_n} \\ \text{und } & u_1(t) = L_{q_1}^{u_1} \text{ und } \dots \text{ und } u_m(t) = L_{q_m}^{u_m}, \\ \text{dann } & \dot{x}_1(t) = L_{w_1(j,q)}^{\dot{x}_1} \text{ und } \dots \text{ und } \dot{x}_n(t) = L_{w_n(j,q)}^{\dot{x}_n} \end{aligned} \quad (3)$$

beschrieben werden. Hierbei sind  $L_{j_i}^{x_i}$ ,  $L_{q_p}^{u_p}$  und  $L_{w_i(j,q)}^{\dot{x}_i}$  die linguistischen Werte der linguistischen Zustände  $x_i$ , der linguistischen Eingänge  $u_p$  und der linguistischen Zustandsableitungen  $\dot{x}_i$ . Letztere beschreibt die Ableitung der Zustandsvariable in sprachlicher Form, beispielsweise durch „positiv“, „null“ und „negativ“.

Jede Regel der Form (3) wird über die Index-Vektoren  $\mathbf{j}$  (Indizes der linguistischen Werte  $L_{j_i}^{x_i}$  der Zustände) und  $\mathbf{q}$  (Indizes der linguistischen Werte  $L_{q_p}^{u_p}$  der Eingänge) sowie  $\mathbf{w}$  (Indizes der linguistischen Werte  $L_{w_i}^{\dot{x}_i}$  der Zustandsableitungen) vollständig durch die Abbildung  $(\mathbf{j}, \mathbf{q}) \rightarrow \mathbf{w}(\mathbf{j}, \mathbf{q})$  definiert. Für ein System mit zwei Zuständen und einem Eingang mit jeweils zwei linguistischen Werten ( $L_1^{x_1}$  und  $L_2^{x_1}$ ) und den linguistischen Zustandsableitungen ( $L_1^{\dot{x}_1}$ ,  $L_2^{\dot{x}_1}$  und  $L_3^{\dot{x}_1}$ ) lässt sich beispielsweise die Regel

$$\begin{aligned} \text{Wenn } & x_1(t) = L_1^{x_1} \text{ und } x_2(t) = L_2^{x_2} \\ \text{und } & u(t) = L_2^u, \\ \text{dann } & \dot{x}_1(t) = L_1^{\dot{x}_1} \text{ und } \dot{x}_2(t) = L_3^{\dot{x}_2} \end{aligned} \quad (4)$$

durch die Abbildung  $(\mathbf{j}, \mathbf{q}) = (1 \ 2 \ 2) \rightarrow (1 \ 3) = \mathbf{w}(\mathbf{j}, \mathbf{q})$  beschreiben.

Fasst man obige linguistische Werte und Ableitungen jeweils in gemeinsamen linguistischen Vektoren  $\mathbf{L}_j^x$ ,  $\mathbf{L}_q^u$  und  $\mathbf{L}_{\mathbf{w}(\mathbf{j}, \mathbf{q})}^{\dot{x}}$  zusammen, so ergibt sich die folgende äquivalente Kurzform der Regeln (3)

$$\text{Wenn } \mathbf{x}(t) = \mathbf{L}_j^x \text{ und } \mathbf{u}(t) = \mathbf{L}_q^u, \quad (5)$$

$$\text{dann } \dot{\mathbf{x}}(t) = \mathbf{L}_{\mathbf{w}(\mathbf{j}, \mathbf{q})}^{\dot{x}} \quad (6)$$

in Form einer *linguistischen Differentialgleichung* mit  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{u} \in \mathbb{R}^m$ .

Einen geschlossenen mathematischen Ausdruck für (5), (6) gewinnt man, wenn jedem linguistischen Wert  $L_{j_i}^{x_i}$  jeder Zustandsvariablen  $x_i$  eine Zugehörigkeitsfunktion  $\mu_{j_i}^{x_i}(x_i)$  zugeordnet wird. Gleiches Vorgehen führt zu den Zugehörigkeitsfunktionen  $\mu_{q_p}^{u_p}(u_p)$  der Eingangsgrößen  $u_p$ . Die Abszisse des Maximalwerts der Zugehörigkeitsfunktion  $s_{j_i}^{x_i}$  bzw.  $s_{q_p}^{u_p}$  wird als Kernposition bezeichnet. Als Zugehörigkeitsfunktionen der linguistischen Ableitungen, also der

Ausgänge von  $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$ , werden Singletons  $s_{w_i(j,q)}^{x_i}$  verwendet, die auch als Kernpositionsableitungen bezeichnet werden. Ähnlich wie bei den linguistischen Werten lassen sich auch die Kernpositionen und Kernpositionsableitungen in Vektorform  $\mathbf{s}_j^x$ ,  $\mathbf{s}_q^u$  und  $\mathbf{s}_{w(j,q)}^x$  zusammenstellen.

Zusätzlich werden an die Zugehörigkeitsfunktionen die folgenden Bedingungen gestellt:

1. Beschränkung:  $\mu_{j_i}^{x_i}(x_i) \in [0, 1]$  für alle  $x_i \in X_i$ ,
2. Konvexität:  $\begin{cases} \mu_{j_i}^{x_i}(x_i) \text{ monoton steigend } \forall x_i < s_{j_i}^{x_i} \\ \mu_{j_i}^{x_i}(x_i) \text{ monoton fallend } \forall x_i > s_{j_i}^{x_i} \end{cases}$ ,
3. Partitionierung:  $\sum_{j_i} \mu_{j_i}^{x_i}(x_i) = 1$  für alle  $x_i \in X_i$  und  $\mu_{j_i}^{x_i}(s_{j_i}^{x_i}) = 1$  und  $\mu_{j_i}^{x_i}(s_{l_i}^{x_i}) = 0$  für  $j_i \neq l_i$ ,
4. Stetigkeit:  $\mu_{j_i}^{x_i}(x_i)$  ist stetig in  $X_i$ .

Verwendet man nun Dreieck- und Rampenfunktionen als Zugehörigkeitsfunktionen, dann lässt sich das KRFS vollständig durch die Lage der Kernpositionen und Kernpositionsableitungen beschreiben (Bild 2a).

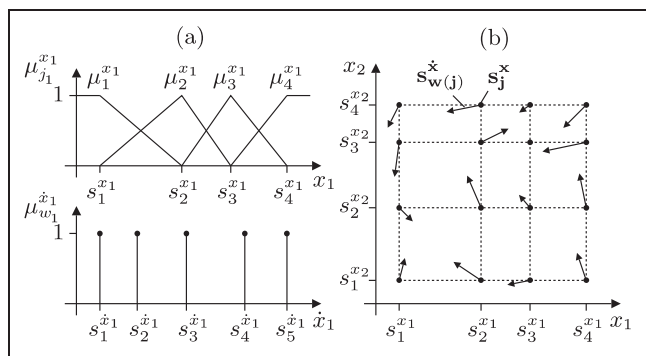
Als Operatoren für die Aggregation und Implikation wird das algebraische Produkt, für die Akkumulation die einfache Summe verwendet. Die Defuzzifizierung erfolgt mit der Center of Singletons Methode.

Mit den Bedingungen 1.–4. und gemäß [2;3] ergibt sich schließlich die mathematische Beschreibung der Dynamik zu

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) = \sum_{j,q} \mathbf{s}_{w(j,q)}^x \prod_{i=1}^n \mu_{j_i}^{x_i}(x_i) \prod_{p=1}^m \mu_{q_p}^{u_p}(u_p). \tag{7}$$

Anschaulich formen die Kernpositionsvektoren  $\mathbf{s}_j^x$  und  $\mathbf{s}_q^u$  ein Gitter im Eingangsraum  $X \times U$ . Die Kernpositionsableitungen  $s_{w_i(j,q)}^{x_i}$  geben an, in welche Richtung und mit welcher Geschwindigkeit die Systemtrajektorie durch den zugehörigen Punkt führt (Bild 2b).

Den im folgenden Abschnitt vorgestellten Lernverfahren für KRFS werden zur Vereinfachung der Darstellung noch zwei Definitionen vorangestellt:



**Bild 2:** (a) Beispiele für Zugehörigkeitsfunktionen mit entsprechenden Kernpositionen  $s_{j_i}^{x_i}$  und Kernpositionsableitungen  $s_{w_i}^{x_i}$ . (b) Die Zugehörigkeitsfunktionen bilden ein Gitter im Zustands-/Eingangsraum [5].

**Definition 1** Prämisse  $\Xi$  einer Regel:

$$\Xi_{(j,q)}(\mathbf{x}, \mathbf{u}) = \prod_{i=1}^n \mu_{j_i}^{x_i}(x_i) \prod_{p=1}^m \mu_{q_p}^{u_p}(u_p). \tag{8}$$

**Definition 2** Menge der zur Kernpositionsableitung  $s_{w^*}^{x_i}$  gehörenden Regeln:

$$\Lambda_i(s_{w^*}^{x_i}) = \{(\mathbf{j}, \mathbf{q}) | s_{w_i(j,q)}^{x_i} = s_{w^*}^{x_i}\}. \tag{9}$$

$\Lambda_i(s_{w^*}^{x_i})$  enthält alle Indexvektoren der Prämisse  $(\mathbf{j}, \mathbf{q})$ , deren Konklusion  $w_i(\mathbf{j}, \mathbf{q})$  durch den Index  $w^*$  gegeben ist, d. h.  $\Lambda_i(s_{w^*}^{x_i})$  beschreibt alle Regeln der Form (3), die denselben linguistischen Ausgangswert besitzen.

### 3 Lernverfahren für KRFS

Der folgende Abschnitt soll genauer beleuchten, wie dynamische Systeme durch rekurrente Fuzzy-Systeme nachgebildet werden können. Grundsätzlich sind hier die zwei folgenden Probleme zu lösen.

Zum einen müssen die Struktur des Fuzzy-Systems, insbesondere die Zahl der Zugehörigkeitsfunktionen für die Zustands- und Eingangsgrößen, sowie die Regeln festgelegt werden. Dies kann bei vielen Systemen auf Basis qualitativen Prozesswissens erfolgen.

Zum anderen müssen die freien Parameter des KRFS, d. h. die Parameter der Zugehörigkeitsfunktionen, so angepasst werden, dass das KRFS den dynamischen Prozess gut nachbilden kann. Diese Optimierung der Systemparameter ist allerdings allein auf Basis von Expertenwissen nahezu unmöglich.

Stattdessen verwendet man  $N$  Messdatensätze der Eingangs- und Zustandsgrößen sowie der Zustandsableitungen

$$\left( \hat{u}_p^{(k)}, \hat{x}_i^{(k)}, \dot{\hat{x}}_i^{(k)} \right)$$

mit  $i = 1, \dots, n, p = 1, \dots, m, k = 1, \dots, N$ , die das Verhalten des Prozesses möglichst repräsentativ wiedergeben. Mit diesen Messdaten ist es dann möglich, das KRFS auch quantitativ an das zu modellierende System anzupassen. Das KRFS lernt also anhand von Beispieldaten das generelle Systemverhalten.

Zunächst wird nun die Optimierung der Ausgangszugehörigkeitsfunktionen (Kernpositionsableitungen) und anschließend die Optimierung der Eingangszugehörigkeitsfunktionen (Kernpositionen) beschrieben.

#### 3.1 Optimierung der Ausgangszugehörigkeitsfunktionen

Zur Optimierung der Kernpositionsableitungen werden, wie oben, zwei Fälle unterschieden, die unterschiedliche Anforderungen an das vorhandene Vorwissen stellen:

- 1) Optimierung ohne Beschränkungen (Abschnitt 3.1.1): Jeder Regel wird eine eigene Kernpositionsableitung  $s_{w_i(j,q)}^{x_i}$  zugeordnet (Bild 3 links). Vorwissen wird hier nur für die Festlegung der Zahl der Eingangszugehörig-

		$x_2$						$x_2$			
		n	ns	ps	p			n	ns	ps	p
$x_1$	$z$	1	2	3	4	$x_1$	$z$	2	1	2	3
	$p$	5	6	7	8		$p$	4	1	2	4
	$n$	9	10	11	12						

**Bild 3:** Regelbasis für  $\hat{x}_i$  ohne Beschränkung (links) und mit Beschränkung (rechts) der Zahl an Kernpositionsableitungen. Die Zahlen von 1–12 kennzeichnen jeweils unterschiedliche linguistische Werte der Zustandsableitungen  $\mu_{w(j,q)}^x$ .

rigkeitsfunktionen benötigt, die Festlegung einer Regelbasis ist dagegen nicht erforderlich. So können die Struktur und die Parameter des Systems bestimmt werden. Nachteilig ist allerdings die große Anzahl von Kernpositionsableitungen.

- 2) Optimierung mit Beschränkungen (Abschnitt 3.1.2): Die Zahl der Kernpositionsableitungen wird eingeschränkt. Dies setzt voraus, dass anhand von Expertenwissen bereits die Struktur, d. h. eine Regelbasis, festgelegt wurde (Bild 3 rechts), wodurch die linguistische Interpretierbarkeit des KRFS gegeben ist. Man bestimmt also nur die Parameter des Systems. Während der Optimierung sind hier Nebenbedingungen einzuhalten.

Zunächst wird Fall 1 betrachtet.

### 3.1.1 Optimierung ohne Beschränkungen

Zur Herleitung wird zunächst die allgemeine Systemdarstellung rekurrenter Fuzzy-Systeme (7) in die Komponenten der Zustandsableitung aufgespalten:

$$\begin{aligned} \dot{x}_i &= \sum_{j,q} s_{w_i(j,q)}^{x_i} \prod_{l=1}^n \mu_{j_l}^{x_l}(x_l) \prod_{p=1}^m \mu_{q_p}^{u_p}(u_p) \\ &= \sum_{j,q} \Xi_{(j,q)}(\mathbf{x}, \mathbf{u}) \cdot s_{w_i(j,q)}^{x_i} = \Xi^T(\mathbf{x}, \mathbf{u}) \cdot \mathbf{s}^{x_i}, \end{aligned} \quad (10)$$

wobei im letzten Schritt die Summe über alle Regeln durch das Vektorprodukt dargestellt wird. Im Vektor  $\Xi(\mathbf{x}, \mathbf{u})$  sind die Prämissen (8) aller Regeln zusammengefasst. Der Vektor  $\mathbf{s}^{x_i}$  enthält die der jeweiligen Prämisse (Regel) zugeordneten Kernpositionsableitungen in  $x_i$ -Richtung. Somit enthält  $\mathbf{s}^{x_i}$  alle Kernpositionsableitungen der linguistischen Zustandsableitung  $\dot{x}_i$ .

Offensichtlich ist Gleichung (10) linear in den zu optimierenden Parametern  $\mathbf{s}^{x_i}$ , sodass sich das Verfahren der kleinsten Quadrate (Least Squares) zur Lösung anbietet. Man minimiert also den quadratischen Fehler zwischen gemessenem  $\hat{x}_i$  und berechnetem Ausgang  $\dot{x}_i$  der  $N$  Datenpaare:

$$\begin{aligned} J_i(\mathbf{s}^{x_i}) &= \frac{1}{2} \sum_{k=1}^N \left( e_i^{(k)} \right)^2 = \frac{1}{2} \sum_{k=1}^N \left( \hat{x}_i^{(k)} - \dot{x}_i^{(k)} \right)^2 \\ &= \frac{1}{2} \sum_{k=1}^N \left( \hat{x}_i^{(k)} - \Xi^T(\hat{\mathbf{x}}^{(k)}, \hat{\mathbf{u}}^{(k)}) \cdot \mathbf{s}^{x_i} \right)^2 \rightarrow \min. \end{aligned} \quad (11)$$

Insgesamt ergeben sich somit  $i = 1, \dots, n$  konvexe Optimierungsprobleme, die unabhängig voneinander gelöst werden können.

Zur Vereinfachung der Schreibweise werden die Regressionsmatrix  $\Psi$  und Vektoren  $\mathbf{z}_i$

$$\Psi = \begin{pmatrix} \Xi^T(\hat{\mathbf{x}}^{(1)}, \hat{\mathbf{u}}^{(1)}) \\ \vdots \\ \Xi^T(\hat{\mathbf{x}}^{(N)}, \hat{\mathbf{u}}^{(N)}) \end{pmatrix}, \quad \mathbf{z}_i = \begin{pmatrix} \hat{x}_i^{(1)} \\ \vdots \\ \hat{x}_i^{(N)} \end{pmatrix} \quad (12)$$

eingeführt. Mittels dieser Notation ergibt sich aus Gl. (11) das Optimierungsproblem zu

$$J_i(\mathbf{s}^{x_i}) = (\mathbf{z}_i - \Psi \cdot \mathbf{s}^{x_i})^T (\mathbf{z}_i - \Psi \cdot \mathbf{s}^{x_i}) \rightarrow \min. \quad (13)$$

Die Lösung dieses Optimierungsproblems in der  $i$ -ten Komponente lautet [7]

$$\mathbf{s}^{x_i} = [\Psi^T \cdot \Psi]^{-1} \Psi^T \mathbf{z}_i. \quad (14)$$

Nach der Optimierung kann man eng beieinanderliegende Komponenten von  $\mathbf{s}^{x_i}$  zu jeweils einer zusammenfassen und so deren Anzahl reduzieren. Nun widmen wir uns Fall 2.

### 3.1.2 Optimierung mit Beschränkungen

Ein Nachteil des Vorgehens in Abschnitt 3.1.1 ist, dass die Interpretierbarkeit des KRFS verloren geht, da jeder Regel eine eigene Kernpositionsableitung zugeordnet wird. Da die Zahl der Regeln  $R$  exponentiell mit der Dimension  $n + m$  des Fuzzy-Systems und der Zahl an Zugehörigkeitsfunktionen in jeder Dimension  $r(\alpha)$  gemäß

$$R = \prod_{\alpha=1}^{n+m} r(\alpha) \quad (15)$$

anstiegt, wächst auch die Zahl der Kernpositionsableitungen exponentiell.

Will man die Interpretierbarkeit beibehalten, so muss man die Zahl der linguistischen Werte der Kernpositionsableitungen begrenzen. Wie bereits erwähnt, muss hierzu die Regelbasis vorher festgelegt werden, es muss also einer bestimmten Prämisse eine Konklusion aus der Menge der möglichen Konklusionen zugeordnet werden. Die Erstellung der Regelbasis erfolgt hierbei mithilfe qualitativen Wissens über das Systemverhalten.

Um auch in diesem Fall eine Optimierungsaufgabe zu entwickeln, geht man wieder von der Systembeschreibung (10) aus. Allerdings sind die Vektoren  $\Xi(\mathbf{x}, \mathbf{u})$  und  $\mathbf{s}^{x_i}$  von denen in Abschnitt 3.1.1 verschieden.

Aufgrund der eingeschränkten Zahl linguistischer Werte gilt jetzt

$$\mathbf{s}^{x_i} = [s_1^{x_i} \dots s_{m_i}^{x_i}]^T, \quad (16)$$

wobei  $m_i$  der Anzahl an linguistischen Werten der  $i$ -ten Zustandsableitung entspricht und die Kernpositionsableitungen aufsteigend bezüglich ihrer linguistischen Werte geordnet sind.

Der Vektor  $\Xi_i(\mathbf{x}, \mathbf{u})$  enthält wiederum die zur jeweiligen Konklusion gehörenden Prämissen und ist definiert durch

$$\Xi_i(\mathbf{x}, \mathbf{u}) = \begin{pmatrix} \sum_{(j,q) \in \Lambda_i(s_1^{\hat{x}_i})} \Xi_{(j,q)}(\mathbf{x}, \mathbf{u}) \\ \vdots \\ \sum_{(j,q) \in \Lambda_i(s_m^{\hat{x}_i})} \Xi_{(j,q)}(\mathbf{x}, \mathbf{u}) \end{pmatrix} \quad (17)$$

mit  $\Lambda_i(s_w^{\hat{x}_i})$  nach Definition 2. Im Gegensatz zum unbeschränkten Fall (Abschnitt 3.1.1) ist  $\Xi_i(\mathbf{x}, \mathbf{u})$  nun von der  $i$ -ten Komponente des Zustandsvektors abhängig, da sich die Beschränkungen in jeder Komponente  $\hat{x}_i$  unterscheiden können.

Mit (16), (17) und ähnlicher Zuordnung wie in (12) kann wiederum das Least-Squares-Problem (13) formuliert werden. Allerdings entstehen Nebenbedingungen, die bei der Optimierung eingehalten werden müssen. So soll ein linguistischer Wert „niedrig“ kleiner als ein linguistischer Wert „hoch“ sein. Es muss also gelten:

$$s_v^{\hat{x}_i} < s_{v+1}^{\hat{x}_i} \quad \forall v = 1, \dots, m-1. \quad (18)$$

Fasst man diese Bedingungen zusammen, so folgt

$$\mathbf{A} \cdot \mathbf{s}^{\hat{x}_i} < \mathbf{0} \quad (19)$$

mit

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & -1 \end{pmatrix}, \quad \mathbf{A} \in \mathbb{R}^{(m-1) \times m}. \quad (20)$$

Insgesamt ergeben sich damit  $i = 1, \dots, n$  konvexe quadratische Optimierungsprobleme der Gestalt

$$J_i(\mathbf{s}^{\hat{x}_i}) = (\mathbf{z}_i - \Psi \cdot \mathbf{s}^{\hat{x}_i})^T (\mathbf{z}_i - \Psi \cdot \mathbf{s}^{\hat{x}_i}) \longrightarrow \min, \quad (21)$$

$$\mathbf{A} \cdot \mathbf{s}^{\hat{x}_i} < \mathbf{0}, \quad (22)$$

zu deren Lösung effiziente Verfahren der quadratischen Programmierung (QP) genutzt werden können [12].

### 3.2 Optimierung der Eingangszugehörigkeitsfunktionen

Will man neben den Kernpositionsableitungen  $\mathbf{s}_{w(j,q)}^{\hat{x}}$ , d.h. den Ausgangszugehörigkeitsfunktionen, auch die Lage der Kernpositionen  $\mathbf{s}_j^x, \mathbf{s}_q^u$ , d.h. die Eingangszugehörigkeitsfunktionen, optimieren, entsteht ein multimodales Optimierungsproblem, zu dessen Lösung verschiedene Suchverfahren [13] existieren.

Eine Gruppe sind gradientenbasierte Verfahren [13], bei denen sich die Parameter

$$\xi = \begin{bmatrix} \mathbf{s}_j^x \\ \mathbf{s}_q^u \end{bmatrix}, \quad \xi \in \mathbb{R}^{l \times 1} \quad (23)$$

mit

$$l = \sum_{\alpha=1}^{n+m} r(\alpha) \quad (24)$$

im aktuellen Iterationsschritt  $\kappa$  aus den Parametern im vorangegangenen Iterationsschritt  $\kappa - 1$  und einem Korrekturterm gemäß

$$\xi(\kappa) = \xi(\kappa - 1) - \eta(\kappa - 1)\mathbf{g}(\kappa - 1) \quad (25)$$

berechnen, wobei  $\eta$  die Schrittweite und  $\mathbf{g}$  den Gradienten der Gütefunktion angibt.

Als Gütefunktion dient der quadratische Fehler zwischen gemessenem  $\hat{x}_i$  und dem anhand von  $\hat{\mathbf{x}}^{(k)}, \hat{\mathbf{u}}^{(k)}$  berechneten Ausgangswert  $\hat{x}_i$ :

$$J_i(\xi) = \frac{1}{2} \sum_{k=1}^N (e_i^{(k)})^2 = \frac{1}{2} \sum_{k=1}^N (\hat{x}_i^{(k)} - \hat{x}_i^{(k)})^2. \quad (26)$$

Der Index  $i$  weist darauf hin, dass bei KRFS insgesamt  $n$  Gütefunktionen  $J_i$  auftreten, die jeweils für eine Komponente der Zustandsableitung  $\hat{x}_i$  gelten. Entsprechend handelt es sich bei der Optimierung der Kernpositionen grundsätzlich um ein Mehrzieloptimierungsproblem, das zusammenfassend in der Vektorform

$$\mathbf{J}(\xi) = \begin{bmatrix} J_1(\xi) \\ \vdots \\ J_n(\xi) \end{bmatrix} \rightarrow \min \quad (27)$$

geschrieben werden kann. Es sei darauf hingewiesen, dass im Gegensatz zur Optimierung der Kernpositionen die Optimierung der Kernpositionsableitungen (Abschnitt 3.1) kein Mehrzieloptimierungsproblem darstellt, da die Gütefunktionen bezüglich der Kernpositionsableitungen entkoppelt sind.

#### 3.2.1 Lösung des Mehrzieloptimierungsproblems

Zur Lösung von Mehrzieloptimierungsproblemen existieren verschiedene Ansätze [11]. Das Mehrzieloptimierungsproblem (27) ist etwas einfacher zu handhaben als allgemeine Mehrzieloptimierungsprobleme, da alle Kriterien die gleiche Form aufweisen.

Deshalb bietet sich die Anwendung der Weighted-Sum-Methode an [15]. Diese Methode gewichtet die einzelnen Optimierungskriterien in geeigneter Weise, wodurch das Mehrzieloptimierungsproblem (27) in ein einzelnes Optimierungsproblem der Form

$$J(\xi) = [\tau_1 \dots \tau_n] \cdot \begin{bmatrix} J_1(\xi) \\ \vdots \\ J_n(\xi) \end{bmatrix} \rightarrow \min \quad (28)$$

überführt wird.

Entscheidend für die erfolgreiche Anwendung der Weighted-Sum-Methode ist die Definition eines geeigneten Vektors  $\tau$ , der die einzelnen Optimierungskriterien ausgewogen ge-



wichtet. Würde man die jeweiligen quadratischen Fehler gleich gewichten, so würden die Ausgangsgrößen mit dem größten Wertebereich den Optimierungsprozess dominieren.

Um dies auszuschließen, werden die quadratischen Fehler gemäß dem Verhältnis der Varianzen der jeweiligen gemessenen Zustandsableitungen  $\dot{x}_i$  gewichtet und damit normiert. Bezeichnet man mit  $\sigma_{\dot{x}_i}$  die Varianz der Messdaten der  $i$ -ten Zustandsableitung, so können die Einträge des Gewichtungsvektors durch

$$\tau_i = \frac{1}{n-1} \cdot \frac{\sum_{j=1, j \neq i}^n \sigma_{\dot{x}_j}}{\sum_{j=1}^n \sigma_{\dot{x}_j}} \quad (29)$$

definiert werden.

### 3.2.2 Bestimmung der Gradienten

Der nächste Schritt ist die Bestimmung des in (25) benötigten Gradienten der Gütefunktion (28) zu

$$\mathbf{g} = \frac{\partial J(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \frac{\partial (\boldsymbol{\tau}^T \cdot \mathbf{J}(\boldsymbol{\xi}))}{\partial \boldsymbol{\xi}} = \boldsymbol{\tau}^T \cdot \frac{\partial \mathbf{J}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}, \quad (30)$$

wobei

$$\frac{\partial \mathbf{J}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \begin{pmatrix} \frac{\partial J_1(\boldsymbol{\xi})}{\partial \xi_1} & \dots & \frac{\partial J_1(\boldsymbol{\xi})}{\partial \xi_l} \\ \vdots & \ddots & \vdots \\ \frac{\partial J_n(\boldsymbol{\xi})}{\partial \xi_1} & \dots & \frac{\partial J_n(\boldsymbol{\xi})}{\partial \xi_l} \end{pmatrix} \quad (31)$$

die Jacobi-Matrix bezüglich des Parametervektors ist.

Für die jeweiligen Einträge von (31), also die Ableitung einer Gütefunktion  $J_i$  gilt allgemein

$$\frac{\partial J_i(\boldsymbol{\xi})}{\partial \xi_v} = - \sum_{k=1}^N (\hat{x}_i^{(k)} - x_i^{(k)}) \cdot \frac{\partial \dot{x}_i^{(k)}(\boldsymbol{\xi})}{\partial \xi_v} \quad (32)$$

mit

$$\frac{\partial \dot{x}_i^{(k)}(\boldsymbol{\xi})}{\partial \xi_v} = \sum_{\mathbf{j}, \mathbf{q}} s_{w_i(\mathbf{j}, \mathbf{q})}^{x_i} \cdot \frac{\partial \Xi(\mathbf{j}, \mathbf{q})(\hat{\mathbf{x}}^{(k)}, \hat{\mathbf{u}}^{(k)})}{\partial \xi_v}. \quad (33)$$

Die Ableitung von  $\Xi(\hat{\mathbf{x}}^{(k)}, \hat{\mathbf{u}}^{(k)})$  nach  $\xi_v$ , d. h. der Kernposition  $s_{r_v}^{x_v}$  bzw.  $s_{q_p}^{x_p}$ , berechnet sich beispielsweise zu

$$\frac{\partial \Xi(\hat{\mathbf{x}}^{(k)}, \hat{\mathbf{u}}^{(k)})}{\partial s_{r_v}^{x_v}} = \frac{\partial \mu_{j_v}^{x_v}(\hat{x}_v^{(k)})}{\partial s_{r_v}^{x_v}} \prod_{i=1, i \neq v}^n \mu_{j_i}^{x_i}(\hat{x}_i^{(k)}) \prod_{p=1}^m \mu_{q_p}^{u_p}(\hat{u}_p^{(k)}), \quad (34)$$

da in jeder Regel nur eine Zugehörigkeitsfunktion pro Dimension aktiviert wird, sodass der Parameter  $s_{r_v}^{x_v}$  nur einmal auftritt.

Schließlich gilt für die Ableitung der dreiecks- und rampenförmigen Zugehörigkeitsfunktionen

$$\frac{\partial \mu_{j_v}^{x_v}(\hat{x}_v^{(k)})}{\partial s_{r_v}^{x_v}} = \begin{cases} \frac{(\hat{x}_v^{(k)} - s_{j_i}^{x_i})}{(s_{j_i}^{x_i} - s_{j_{i-1}}^{x_i})^2} \cdot \delta(s_{j_{i-1}}^{x_i}, s_{j_i}^{x_i}, \hat{x}_v^{(k)}) & \text{wenn } r_v = j_i - 1, \\ -\frac{(\hat{x}_v^{(k)} - s_{j_{i-1}}^{x_i})}{(s_{j_i}^{x_i} - s_{j_{i-1}}^{x_i})^2} \cdot \delta(s_{j_{i-1}}^{x_i}, s_{j_i}^{x_i}, \hat{x}_v^{(k)}) & \\ -\frac{(\hat{x}_v^{(k)} - s_{j_{i+1}}^{x_i})}{(s_{j_{i+1}}^{x_i} - s_{j_i}^{x_i})^2} \cdot \delta(s_{j_i}^{x_i}, s_{j_{i+1}}^{x_i}, \hat{x}_v^{(k)}) & \text{wenn } r_v = j_i, \\ \frac{(\hat{x}_v^{(k)} - s_{j_i}^{x_i})}{(s_{j_{i+1}}^{x_i} - s_{j_i}^{x_i})^2} \cdot \delta(s_{j_i}^{x_i}, s_{j_{i+1}}^{x_i}, \hat{x}_v^{(k)}) & \text{wenn } r_v = j_i + 1, \\ 0 & \text{sonst,} \end{cases} \quad (35)$$

wobei

$$\delta(a, b, \hat{x}_v^{(k)}) = \begin{cases} 1 & \text{wenn } a < \hat{x}_v^{(k)} \leq b, \\ 0 & \text{sonst.} \end{cases} \quad (36)$$

Die Ableitungen nach den Kernpositionen  $s_q^u$  ergeben entsprechende Ergebnisse.

Genau genommen ist die Ableitung der Zugehörigkeitsfunktion in der Kernposition nicht differenzierbar. Allerdings kann man die Ableitung in der Kernposition ohne praktische Einschränkung entweder zu null oder auf den links- oder rechtsseitigen Grenzwert setzen.

### 3.3 Gleichzeitige Optimierung aller Zugehörigkeitsfunktionen

Nachdem in den vorigen Abschnitten die Lernverfahren für die Kernpositionen und die Kernpositionsableitungen vorgestellt wurden, werden diese nun in einen Gesamtalgorithmus [8] integriert, sodass eine gleichzeitige Optimierung aller Parameter erfolgen kann:

1. Initialisierungsschritt: Die Anzahl der Zugehörigkeitsfunktionen und die Lage der Kernpositionen in jeder Dimension wird festgelegt. Außerdem werden die Kernpositionsableitungen mittels Least Squares oder QP initialisiert.
2. Lernen der Kernpositionen: Aktualisieren der Kernpositionen  $\boldsymbol{\xi}$  durch Anwendung des Gradientenverfahrens (Abschnitt 3.2)
3. Lernen der Kernpositionsableitungen: Aktualisieren der Kernpositionsableitungen  $s^{x_i}$  durch Anwendung des Least-Squares-Verfahrens (Abschnitt 3.1.1) oder der QP (Abschnitt 3.1.2).
4. Überprüfung des Abbruchkriteriums: Unterschreitet die Parameteränderung eine vorgegebene Grenze oder steigt der Validierungsfehler an, wird der Algorithmus beendet. Ansonsten gehe zu Schritt 2.

Der Algorithmus bietet den Vorteil, unabhängig vom Grad des Vorwissens anwendbar zu sein. Ist beispielsweise eine

komplette Regelbasis gegeben, so werden die Kernpositionsableitungen mit Beschränkungen optimiert, ist keine Regelbasis, sondern nur die Zahl der Kernpositionen in jeder Dimension festgelegt, wird entsprechend ohne Beschränkungen optimiert.

### 4 Modellierung eines Bioreaktors

Die Leistungsfähigkeit zeitkontinuierlicher rekurrenter Fuzzy-Systeme zur Modellierung technischer Systeme soll am Beispiel eines Bioreaktors gezeigt werden. Die Modellierung derartiger Bioreaktoren erfolgt durch Aufstellen der Massenbilanzen [4]. Betrachtet man nur die normierten Zellmassen  $x_1$  und Substratmassen  $x_2$ , so lassen sich die folgenden Bilanzen

$$\dot{x}_1 = \mu(x_2)x_1 - ux_1, \tag{37}$$

$$\dot{x}_2 = \sigma(x_2)x_1 - ux_2 \tag{38}$$

herleiten. Hierbei ist  $u$  die Verdünnungsrate, die als Stellgröße dient,  $\mu$  die Wachstumsrate und  $\sigma$  die Substratverbrauchsrate.

Im vorliegenden Benchmark-System (37), (38) werden die Wachstums- und Verbrauchsrate mit dem Substrat-Inhibition-Modell [4]

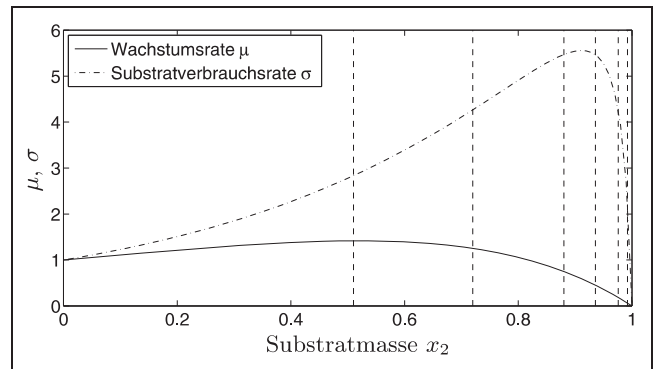
$$\mu(x_2) = (1 - x_2)e^{x_2/\gamma}, \tag{39}$$

$$\sigma(x_2) = \frac{1 + \beta}{1 + \beta - x_2} \mu(x_2) \tag{40}$$

und den Modellparametern  $\beta = 0,02$ ,  $\gamma = 0,48$  nachgebildet.

Zur Modellierung des Bioreaktors wird zunächst die Zahl der Zugehörigkeitsfunktionen in jeder Dimension sowie die Lage der Kernpositionen festgelegt. Die Erfahrung im Umgang mit dem Reaktor zeigt, dass insbesondere die Substratmasse  $x_2$  stark nichtlinear in die Zustandsraumbeschreibung (37), (38) eingeht. Deswegen wird die Zahl der Zugehörigkeitsfunktionen in  $x_2$  auf sechs festgelegt. Für die Zellmasse  $x_1$  und die Verdünnung  $u$  sind jeweils zwei Zugehörigkeitsfunktionen ausreichend, sodass sich 24 Regeln in jeder Zustandsgröße ergeben. Die beiden äußeren Kernpositionen werden auf den Minimal- und Maximalwert der Trainingsdaten in der jeweiligen Dimension festgelegt, um die Zahl der zu optimierenden Parameter zu verringern und die Extrapolation von Trainingsdaten zu vermeiden. Die Regelbasis wird zunächst nicht vorgegeben. Entsprechend wird auch die Zahl der Ausgangszugehörigkeitsfunktionen (Kernpositionsableitungen) nicht beschränkt.

Zur Generierung von Trainingsdaten wird der Prozess mit einem amplitudenmodulierten Pseudo-Rausch-Binär-Signal im Bereich  $u \in [0,3 \ 1,1]$  angeregt. Bei der Wahl dieses Anregungsbereichs wurde darauf geachtet, dass insbesondere der Bereich hoher Substratkonzentrationen ausreichend angeregt wird, da vor allem hier stark nichtlineares Verhalten zu beobachten ist (vgl. Bild 4).

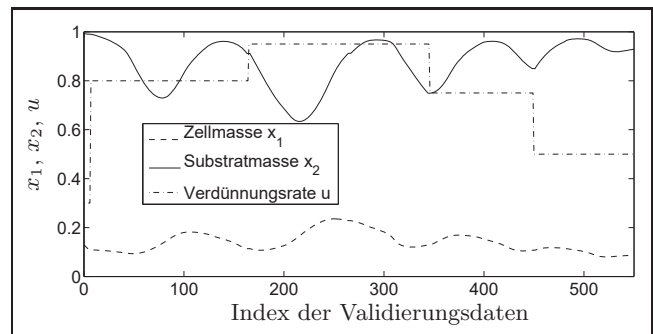


**Bild 4:** Typischer Verlauf von Wachstums- und Substratverbrauchsrate über der Substratmasse. Die senkrechten gestrichelten Linien markieren die optimierten Kernpositionen  $s_2^2$ .

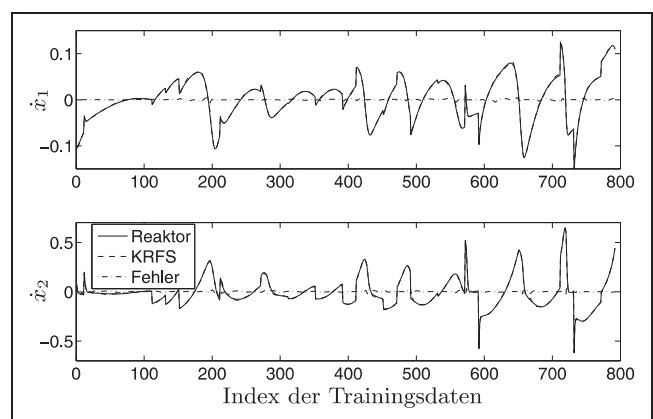
Zusätzlich wurde ein Validierungsdatensatz generiert (Bild 5), mit dem die Generalisierungsfähigkeit des KRFS überprüft werden soll.

Nach der Initialisierung werden nun die Kernpositionen und Kernpositionsableitungen mithilfe des Trainingsdatensatzes gelernt.

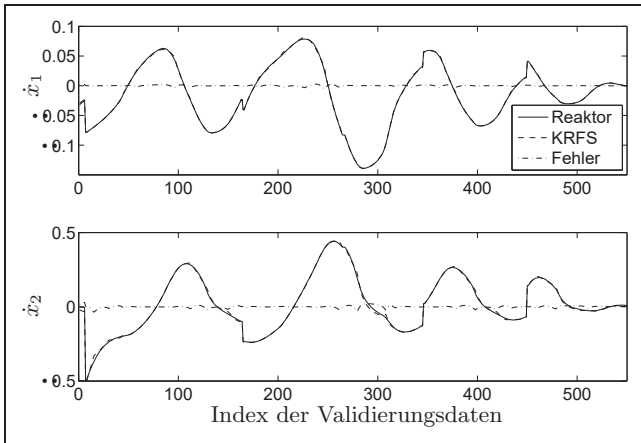
Die Verläufe der Ableitungen der Zellmasse und der Substratmasse sind in Bild 6 gezeigt. Man erkennt, dass das KRFS eine sehr gute Übereinstimmung mit den Trainingsdaten aufweist. Der mittlere quadratische Fehler (RMSE)



**Bild 5:** Verlauf der Validierungsdaten.



**Bild 6:** Vergleich der Zustandsableitungen  $\dot{x}_1$  und  $\dot{x}_2$  für den Trainingsdatensatz. Das rekurrente Fuzzy-System kann das dynamische Verhalten des Bioreaktors sehr gut nachbilden.



**Bild 7:** Vergleich der Zustandsableitungen  $\dot{x}_1$  und  $\dot{x}_2$  für den Validierungsdatensatz.

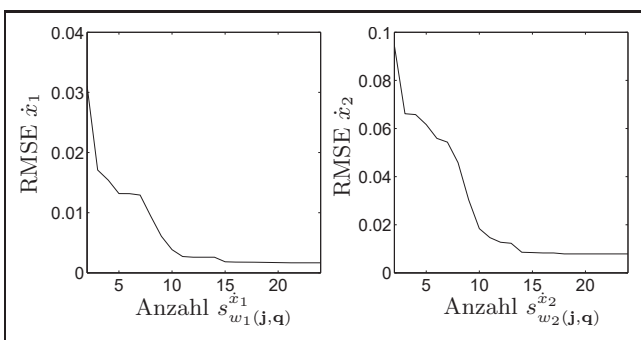
für eine Ein-Schrittprädiktion beträgt 0,14% für  $\dot{x}_1$  und 0,66% für  $\dot{x}_2$ .

Die Generalisierungsfähigkeit des KRFS lässt sich anhand der Verläufe des Validierungsdatensatzes in Bild 7 zeigen. Auch hier ist eine sehr gute Übereinstimmung mit einem RMSE von 0,17% für  $\dot{x}_1$  und 0,92% für  $\dot{x}_2$  zu beobachten.

Zur Veranschaulichung der Funktionsweise rekurrenter Fuzzy-Systeme wurde Bild 4 um die Lage der sechs optimierten Kernpositionen für  $x_2$  ergänzt. Da die Nichtlinearität hauptsächlich durch die Substratverbrauchsrate  $\sigma$  erzeugt wird, passt sich das durch die Kernpositionen entstehende Gitter (vgl. Bild 2) dieser an. Entsprechend liegen die Kernpositionen vermehrt im Bereich hoher Substratmassen.

Wie bereits angesprochen, ist die Interpretierbarkeit des gelernten Modells aufgrund der großen Zahl von 24 Ausgangszugehörigkeitsfunktionen kaum gegeben. Allerdings kann man bereits aus der Verteilung der Kernpositionsableitungen weitergehende qualitative Informationen gewinnen, sodass es gelingt, die Zahl an Kernpositionen zu verringern.

Bild 8 zeigt, wie sich der RMSE über der Zahl an Kernpositionsableitungen verändert, wobei die Parameter jeweils mit Quadratischer Programmierung optimiert wurden. Wie er-



**Bild 8:** Entwicklung des RMSE über der Zahl an Kernpositionsableitungen für  $\dot{x}_1$  (links) und  $\dot{x}_2$  (rechts). Erwartungsgemäß nimmt der RMSE bei geringerer Zahl an Kernpositionsableitungen zu.

Ableitung der Zellmasse  $\dot{x}_1$

		$x_2$												
		vs	s	ms	ml	l	vl							
$x_1$	s	7	7	6	6	5	5	$u=1$	$x_2$					
	l	10	9	8	6	4	4		vs	s	ms	ml	l	vl

Ableitung der Substratmasse  $\dot{x}_2$

		$x_2$												
		vs	s	ms	ml	l	vl							
$x_1$	s	5	5	5	5	5	4	$u=1$	$x_2$					
	l	6	9	10	10	8	6		vs	s	ms	ml	l	vl

**Bild 9:** Regelbasis, mit der die Dynamik des Bioreaktors linguistisch beschrieben werden kann. Hierbei ist vs = „very small“, s = „small“, ms = „medium small“, ml = „medium large“, l = „large“, vl = „very large“. Die Zahlen von 1–10 kennzeichnen aufsteigend linguistische Werte für die Ableitung der Zustände.

wartet verschlechtert sich der RMSE bei abnehmender Zahl an Kernpositionsableitungen, sodass ein Kompromiss zwischen Interpretierbarkeit und Modellgüte gefunden werden muss. Es zeigte sich, dass auch mit jeweils 10 Kernpositionsableitungen in  $\dot{x}_1$  ( $RMSE_{train} = 0,43\%$ ,  $RMSE_{valid} = 0,38\%$ ) und  $\dot{x}_2$  ( $RMSE_{train} = 1,84\%$ ,  $RMSE_{valid} = 1,83\%$ ) noch eine zufriedenstellende Modellgüte erreicht werden kann. Bild 9 zeigt die resultierende Regelbasis, mit der die dynamischen Eigenschaften des Bioreaktors linguistisch beschrieben werden können.

Aus der Regelbasis lässt sich das Verhalten der bestimmten Nichtlinearitäten, ausgedrückt durch Wachstums- und Substratverbrauchsrate, ablesen. Betrachtet man in der Regelbasis den Verlauf von  $\dot{x}_1$  über der Substratmasse  $x_2$ , so erkennt man in Übereinstimmung mit dem Verlauf der Wachstumsrate (Bild 4) eine kontinuierliche Abnahme, die mit  $x_1$  skaliert wird.

Das inhibierende Verhalten der Substratverbrauchsrate wird insbesondere aus der Regelbasis von  $\dot{x}_2$  deutlich. Betrachtet man hierzu die Regeln für  $x_1 = „l“$ , so erkennt man einen Anstieg der linguistischen Werte in  $\dot{x}_2$  bis  $x_2 = „ml“$  und einen deutlichen Abfall für größere Substratmassen. Dies entspricht exakt dem aufgrund der Lage der Kernpositionen in Bild 4 zu erwartenden Verhalten. Ist dagegen  $x_1 = „s“$ , tritt kein Sprung in den linguistischen Werten auf. Da  $x_1$  multiplikativ wirkt, hat die Nichtlinearität angesichts der hohen Verdünnungsrate nur geringen Einfluss und tritt in der Regelbasis nicht hervor.

## 5 Zusammenfassung

In diesem Beitrag wurde die Anwendung zeitkontinuierlicher rekurrenter Fuzzy-Systeme zur Modellierung technischer Systeme am Beispiel eines Bioreaktors gezeigt. Hierzu wurden Verfahren zur Optimierung der Zugehörigkeitsfunktionen vorgestellt, die unterschiedliche Formen von Vorwissen berücksichtigen. Die Ergebnisse zeigen, dass auf Basis von qualitativem Wissen und unter Nutzung von Messdaten genaue Modelle stark nichtlinearer Systeme einfach ermittelt werden können. Dabei können sowohl die Modellstruktur als auch die Modellparameter ermittelt werden.



## Literatur

- [1] J. Adamy. Breakout prediction for continuous casting by fuzzy mealy automata. In *Proc. of EUFIT'95*, S. 754–759, Aachen, 1995.
- [2] J. Adamy and A. Flemming. Equilibria of continuous-time recurrent fuzzy systems. *Fuzzy Sets and Systems*, 157(22):2913–2933, 2006.
- [3] J. Adamy and R. Kempf. Regularity and chaos in recurrent fuzzy systems. *Fuzzy Sets and Systems*, 140(2):259–284, 2003.
- [4] P. Agrawal, C. Lee, H. Lim, and D. Ramkrishna. Theoretical investigations of dynamic behaviour of isothermal stirred tank biological reactors. *Chemical Engineering Science*, 37:453–462, 1982.
- [5] A. Flemming and J. Adamy. Modeling solide oxide fuel cells using continuous-time recurrent fuzzy systems. *Engineering Applications of Artificial Intelligence*, 21:1289–1300, 2008.
- [6] V. Gorrini and H. Bersini. Recurrent fuzzy systems. In *Proc. 3rd IEEE Conf. on Fuzzy Systems*, volume 1, S. 193–198, Orlando, USA, 1994.
- [7] R. Isermann. *Identifikation dynamischer Systeme 1*. Springer, 1992.
- [8] J. Jang. Anfis: Adaptive-network-based fuzzy inference system. *IEEE Trans. Syst., Man, Cybern.*, 23:665–685, 1993.
- [9] R. Kempf and J. Adamy. Equilibria of recurrent fuzzy systems. *Fuzzy Sets and Systems*, 140(2):231–257, 2003.
- [10] R. Kempf and J. Adamy. Sequential pattern recognition employing recurrent fuzzy systems. *Fuzzy Sets and Systems*, 146(3):451–472, 2004.
- [11] G. Liu, J. Yang, and J. Whitborne. *Multiobjective Optimization and Control*. Research Studies Press, 2002.
- [12] D. Luenberger. *Linear and Nonlinear Programming*. Springer, 2003.
- [13] O. Nelles. *Nonlinear System Identification*. Springer, 2002.
- [14] T. Takagi and M. Sugeno. Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. Syst., Man, Cybern.*, 15(1):116–132, 1985.
- [15] L. Zadeh. Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control*, 8, 1963.

Manuskripteingang: 27. August 2008.



**Prof. Dr.-Ing. Jürgen Adamy** ist Leiter des Fachgebietes Regelungstheorie und Robotik im Fachbereich Elektrotechnik und Informationstechnik der Technischen Universität Darmstadt. Hauptarbeitsgebiete: Regelungsverfahren, Computational Intelligence, autonome mobile Roboter.

Adresse: Technische Universität Darmstadt, Fachbereich Elektrotechnik und Informationstechnik, Fachgebiet Regelungstheorie und Robotik, Landgraf-Georg-Str. 4, 64283 Darmstadt, Fax: +49-(0)6151-16-2507, E-Mail: adamy@rtr.tu-darmstadt.de



**Dipl. Ing. Andreas Schwung** ist wissenschaftlicher Mitarbeiter am Fachgebiet Regelungstheorie und Robotik im Fachbereich Elektrotechnik und Informationstechnik der Technischen Universität Darmstadt. Hauptarbeitsgebiet: Modellbildung mit rekurrenten Fuzzy-Systemen.

Adresse: Technische Universität Darmstadt, Fachbereich Elektrotechnik und Informationstechnik, FG Regelungstheorie und Robotik, 64283 Darmstadt, Fax: +49-(0)6151-16-2507, E-Mail: aschwung@rtr.tu-darmstadt.de