



Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

Wissenschaftliche Berichte
FZKA 6767

Proceedings
12. Workshop Fuzzy Systeme
Dortmund, 13.-15. November 2002

R. Mikut, M. Reischl (Hrsg.)
Institut für Angewandte Informatik

November 2002

Forschungszentrum Karlsruhe

in der Helmholtz-Gemeinschaft

Wissenschaftliche Berichte

FZKA 6767

Proceedings 12. Workshop Fuzzy Systeme

Dortmund, 13.-15. November 2002

Ralf Mikut, Markus Reischl (Hrsg.)

Institut für Angewandte Informatik

Forschungszentrum Karlsruhe GmbH, Karlsruhe

2002

Für diesen Bericht behalten wir uns alle Rechte vor

Forschungszentrum Karlsruhe GmbH
Postfach 3640, 76021 Karlsruhe

Mitglied der Hermann von Helmholtz-Gemeinschaft
Deutscher Forschungszentren (HGF)

ISSN 0947-8620

VORWORT

Dieser Tagungsband enthält die Beiträge des 12. Workshops „Fuzzy Systeme“ des Fachausschusses 5.22 „Fuzzy Control“ der VDI/VDE-Gesellschaft für Mess- und Automatisierungstechnik (GMA) und der Fachgruppe „Fuzzy-Systeme und Soft-Computing“ der Gesellschaft für Informatik (GI), der vom 13.-15. November 2002 im Haus Bommerholz, Dortmund, stattfindet.

Der jährliche Workshop unseres Fachausschusses bietet ein Forum zur Diskussion neuer methodischer Ansätze und industrieller Anwendungen auf dem Gebiet der Fuzzy-Logik und in angrenzenden Gebieten wie Künstlichen Neuronalen Netzen und Evolutionären Algorithmen. Besondere Schwerpunkte sind automatisierungstechnische Anwendungen, z.B. in der Verfahrenstechnik, Energietechnik, Kfz-Technik, Robotik und Medizintechnik, aber auch Lösungen in anderen Problemgebieten (z.B. Data Mining für technische und nichttechnische Anwendungen) sind von Interesse.

Die Ergebnisse werden von den ca. 50 Mitgliedern und Gästen aus Hochschulen, Forschungseinrichtungen und der Industrie präsentiert und in Klausuratmosphäre intensiv diskutiert. Dabei ist es gute Tradition, auch neue Ansätze und Ideen bereits in einem frühen Entwicklungsstadium vorzustellen, in dem sie noch nicht vollständig ausgereift sind.

Nähere Informationen zum Fachausschuss erhalten Sie unter

<http://www.iai.fzk.de/medtech/biosignal/gma/index.html>.

Die Herausgeber bedanken sich an dieser Stelle bei allen Autoren und Rednern sowie bei Prof. Dr. Klawonn (Fachhochschule Braunschweig/Wolfenbüttel), Herrn Dr. Runkler (Siemens AG), Herrn Dr. Jäkel (Forschungszentrum Karlsruhe) und Herrn Dr. Kroll (ABB Heidelberg), die maßgeblich an der Vorbereitung des Workshops beteiligt waren.

Ralf Mikut und Markus Reischl

INHALTSVERZEICHNIS

Klawonn, F., <i>FH Braunschweig/Wolfenbüttel:</i> Visualisierung von Fuzzy-Clusteranalyse-Ergebnissen	1
Kiendl, H.; Haendel, L., <i>Universität Dortmund:</i> Datenbasierte Regelgenerierung mit modell-basierten Clusterverfahren	13
Seising, R., <i>Universität Wien:</i> Eine kleine Geschichte der Fuzzy-Systeme in der Medizintechnik (Übersichtsvortrag)	27
Loose, T.; Jäkel, J.; Mikut, R., <i>Forschungszentrum Karlsruhe:</i> Datenbasierte Generierung natürlichsprachlicher Erklärungstexte am Beispiel der Instrumentellen Ganganalyse	43
von Schmidt, B., <i>Deutsches Zentrum für Luft- und Raumfahrt:</i> Geometrische Veranschaulichung von Fuzzy-Klassifikationssystemen und Multilayer Perceptronen und Parallelen zwischen beiden Modellen	58
Kästner, W.; Traichel A.; Förster T.; Hampel R., <i>Hochschule Zittau/ Görlitz (FH):</i> Zur Problematik der Gewichtsverteilung in Künstlichen Neuronalen Netzen	70
Frey, Ch.; Kuntze, H.-B.; Munser, R., <i>FhG IITB Karlsruhe:</i> Anwendung von Neuro-Fuzzy-Methoden zur multisensoriellen Schadensdiagnose in Abwasserkanälen	81
Kempf, R.; Staab, H., <i>Technische Universität Darmstadt:</i> Funktionswurzeln und Lernalgorithmen für abgetastete Systeme	90
Klose, A., <i>Universität Magdeburg:</i> An Evolutionary Algorithm for Semi-Supervised Fuzzy Classification	100
Felgner, F., <i>Universität Kaiserslautern:</i> Entwicklung einer Fuzzy-Control-Bibliothek in Modelica	107
Haendel, L., <i>Universität Dortmund:</i> Benchmark: Qualitätskontrolle bei KFZ-Aggregaten	119
Mikut, R., <i>Forschungszentrum Karlsruhe:</i> Fuzzy-Modellbildung für den Benchmarkdatensatz Kfz-Aggregate	127

Pfeiffer, B.M.; Reischl, M., <i>Siemens AG, Forschungszentrum Karlsruhe:</i> Entwicklungsstand und WWW-Präsenz des Tutorials "Erfolgreiche Anwendungen von Fuzzy Control" des GMA FA 5.22	135
Silva, C. A.; Runkler, T. A.; Sousa, J. M., <i>Siemens AG, Universität Lissabon:</i> Evolved ant colonies optimization in the scheduling of logistic proc- esses	145
Kuhn, C., <i>Technische Universität Ilmenau:</i> Ein Zwei-Welten-Modell für die Prozessanalyse und Klassifikation	159
Beck, S.; Lehmann, A.; Martin, J.; Mikut, R., <i>Forschungszentrum Karlsruhe:</i> Modellbildung und Fuzzy-Gelenkpositionsregelung für eine 5-Finger- Roboterhand mit flexiblen Fluidaktoren	177
Karimanzira, D., <i>Technische Universität Ilmenau:</i> Ein neuronales Netzwerkssystem zur Kompensation stückweiser steti- ger Nichtlinearitäten in Regelungssystemen	192
Patzwahl, S.; Kramer, K. D.; Nacke, T., <i>Hochschule Harz:</i> Computational Intelligence auf Mikrocontrollersystemen am Beispiel eines Fuzzy-Systems zur Optimierung von Biogasanlagen	200

Visualisierung von Fuzzy-Clusteranalyse-Ergebnissen

Frank Klawonn

Fachbereich Informatik
Fachhochschule Braunschweig/Wolfenbüttel
Salzdahlumer Str. 46-48
38302 Wolfenbüttel
Tel.: (05331) 9396111
Fax: (05331) 9396002
E-Mail: f.klawonn@fh-wolfenbuettel.de

1 Einleitung

Die Clusteranalyse gehört zu dem Bereich der explorativen Datenanalyseverfahren, bei denen es darum geht, Strukturen, Muster oder Gruppen in Daten zu finden. In diesem Sinne nimmt die Clusteranalyse eine unüberwachte Klassifikation vor. Fuzzy-Clustering berücksichtigt die Tatsache, dass die Klassen oder Cluster innerhalb der Daten im allgemeinen nicht vollständig separiert sind, so dass anstelle einer festen Zuordnung eines Datum zu genau einer Klasse jeweils ein Zugehörigkeitsgrad zwischen 0 und 1 berechnet wird. Die gebräuchlichsten Fuzzy-Clusteranalyse-Techniken berechnen die Clustereinteilung, indem sie versuchen, eine Zielfunktion zu minimieren. Die wesentlichen Variablen dieser Zielfunktion sind die Zugehörigkeitsgrade und die Parameter, die die Lage und Form der Cluster bestimmen. Die Verfahren berechnen immer zumindest irgendeine Clustereinteilung, die aber einem unerwünschten lokalen Minimum der Zielfunktion entsprechen kann. Selbst wenn das globale Minimum gefunden wird, kann das Clusteranalyse-Ergebnis unzureichend sein, wenn die Clusterformen oder die Clusteranzahl falsch gewählt wurden. Da die Daten im allgemeinen hochdimensional sind, scheidet eine direkte Betrachtung der Daten und des Analyseergebnisses aus. Visualisierungsmethoden wie die Multidimensionale Skalierung können häufig nur ein sehr eingeschränktes Bild wiedergeben. Trotzdem ist eine Beurteilung eines Clusteranalyse-Ergebnisses unerlässlich. Clustergütemaße versuchen, dieses Problem zu lösen. Allerdings wird bei diesen Gütemaßen eine große Datenanzahl, die auf mehrere Cluster verteilt wird, häufig auf eine einzige Maßzahl reduziert.

In diesem Beitrag wird aufgezeigt, wie aufbauend auf den Prinzipien der Clustergütemaße eine grafische Veranschaulichung vorgenommen werden kann. Diese grafische Repräsentation enthält wesentlich mehr Informationen als ein Gütemaß, das eine Informationskompression auf eine einzelne reelle Zahl vornimmt. Auf diese Weise können Inkonsistenzen in der Clustereinteilung oder sogar in einzelnen Clustern leichter festgestellt werden.

Abschnitt 2 führt kurz die für das zielfunktionsbasierte Fuzzy-Clustering benötigten Grundlagen ein. Die Grundkonzepte für Clustergütemaße werden im Abschnitt 3 vorgestellt. Die alternativen grafischen Repräsentationen für die Clustergüte sind Thema des Abschnitts 4. In den abschließenden Bemerkungen wird aufgezeigt, dass

die Visualisierungsmethoden auch für harte (nicht-fuzzy) Clusteringverfahren geeignet sind.

2 Zielfunktionsbasiertes Fuzzy-Clustering

Fuzzy-Clustering zielt auf das Auffinden von Strukturen in Daten ab. Ein Datensatz wird in eine endliche Anzahl von Clustern aufgeteilt, die die Strukturen in den Daten repräsentieren. Beim Fuzzy-Clustering wird – im Gegensatz zum herkömmlichen partitionierenden (harten) Clustering – ein einzelner Datenvektor nicht eindeutig einem Cluster zugeordnet, sondern es wird ein Zugehörigkeitsgrad des Datenvektors zu jedem Cluster bestimmt, um verrauschte und nicht eindeutig zuzuordnende Daten handhaben zu können. Die meisten Fuzzy-Clustering-Verfahren basieren auf der Minimierung einer Zielfunktion unter Nebenbedingungen. Der am häufigsten verwendete Ansatz ist das probabilistische Clustering mit der zu minimierenden Zielfunktion

$$f = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij} \quad (1)$$

unter den Nebenbedingungen

$$\sum_{i=1}^c u_{ij} = 1 \quad \text{für alle } j = 1, \dots, n. \quad (2)$$

Dabei wird die Anzahl der Cluster zunächst fest gewählt. Für den zu clusternden Datensatz $\{x_1, \dots, x_n\} \subset \mathbb{R}^p$ müssen die Zugehörigkeitsgrade u_{ij} des jeweiligen Datenvektors x_j zum i ten Cluster bestimmt werden. d_{ij} ist ein Distanzmaß, das den Abstand oder die Unähnlichkeit des Datenvektors x_j zum Cluster i spezifiziert, zum Beispiel der (quadratierte) euklidische Abstand von x_j zum i ten Clusterzentrum. Der Parameter $m > 1$ – Fuzzifier genannt – steuert, wie stark Cluster bezogen auf die Zugehörigkeitsgrade überlappen dürfen. Die Nebenbedingungen (2) begründen den Namen probabilistisches Clustering, da die Zugehörigkeitsgrade u_{ij} in diesem Fall auch als die Wahrscheinlichkeit, dass x_j zum Cluster i gehört, interpretierbar sind.

Die in der Zielfunktion zu optimierenden Parameter sind die Zugehörigkeitsgrade u_{ij} und die Clusterparameter, die hier nicht explizit angegeben sind. Sie sind in den Distanzen d_{ij} verborgen. Der gebräuchliche Weg, dieses nicht-lineare Optimierungsproblem – die Minimierung der Zielfunktion (1) – zu lösen, besteht aus einem alternierenden Schema, das jeweils abwechselnd einen der Parametersätze Zugehörigkeitsgrade und Clusterparameter festhält und den anderen Parametersatz bezüglich des festgehaltenen optimiert. In dieser Arbeit geht es nicht um die verschiedenartigen Clusterformen (Kugeln, Ellipsoide, lineare Mannigfaltigkeiten, Quadriken, ...), die durch eine geeignete Wahl der Distanzfunktion darstellbar sind. (Einen Überblick über derartige Verfahren findet man z.B. in [2, 5].) Die hier vorgestellten Visualisierungstechniken lassen sich auf fast alle üblichen Clusterformen anwenden. Allerdings gibt es für Shell-Clustering-Techniken, die die Erkennung von Objekträndern z.B. in Form von Geraden, Kreis- oder Ellipsenrändern haben, besser angepasste Güte-maße. Da Shell-Clustering aber meistens für Anwendungen in der Bildverarbeitung

entwickelt wurde, handelt es sich in diesem Fall meistens um zweidimensionale Daten, bei denen die Visualisierung ohnehin unproblematisch ist.

Die hier vorgestellten Visualisierungstechniken lassen sich ebenfalls auf das Noise Clustering [3] anwenden, bei dem der probabilistische Grundgedanke beibehalten wird. Neben den zu bestimmenden Clustern wird jedoch noch ein zusätzliches Cluster für Rauschdaten eingeführt. Alle Datenvektoren besitzen einen festen (großen) Abstand zum Rauschcluster. Auf diese Weise behalten Datenvektoren, die an der Grenze zwischen zwei Clustern liegen, weiterhin einen hohen Zugehörigkeitsgrad zu beiden Clustern. Datenvektoren, die Rauschen darstellen und von allen Clustern weiter entfernt sind, werden dem Rauschcluster mit einem hohen Zugehörigkeitsgrad zugeordnet und beeinflussen die anderen Cluster weniger.

Possibilistisches Clustering [6], bei dem die probabilistische Nebenbedingung unter Einführung eines Strafterms für kleine Zugehörigkeitsgrade vollständig aufgegeben wird, um die triviale Lösung $u_{ij} = 0$ zu vermeiden, ist nur sehr begrenzt für die hier vorgestellten Visualisierungstechniken geeignet. Possibilistisches Clustering ist ohnehin formal nicht sauber begründbar, da das Ziel der Clusteranalyse explizit in der Vermeidung des globalen Minimums der Zielfunktion und in der Suche nach einem lokalen Minimum besteht [7].

3 Clustergütemaße

Clustergüte bezeichnet das Problem, ob eine gegebene (Fuzzy-)Clustereinteilung überhaupt den Daten angemessen ist. Das im vorigen Abschnitt erwähnte alternierende Optimierungsschema versucht immer, eine bestmögliche Anpassung der parametrisierten Clusterformen – bei vorgegebener Anzahl – an die Daten zu erreichen. Das bedeutet allerdings keineswegs, dass selbst das globale Optimum der Zielfunktion eine sinnvolle Clustereinteilung der Daten repräsentiert. Die Anzahl der Cluster kann falsch gewählt worden sein oder die gesuchten Clusterformen können in den Daten unter Umständen überhaupt nicht vorhanden sein, sofern überhaupt Strukturen den Daten zugrundeliegen. Das Problem der Clustergüte ist ähnlich wie bei der linearen Regression, wo eine bestmögliche Anpassung einer linearen Funktion an die Daten keineswegs bedeutet, dass ein linearer Zusammenhang in den Daten vorhanden ist.

Clustergütemaße werden sowohl zur Beurteilung eines Clusteranalyseergebnisses ganz allgemein genutzt als auch zur Bestimmung der Anzahl der Cluster. Zur Anzahlbestimmung wird die Clusteranalyse mit verschiedenen Anzahlen von Clustern durchgeführt und das bzgl. des ausgewählten Gütemaßes beste Ergebnis verwendet.

Es sollen hier kurz einige Clustergütemaße vorgestellt werden, um die zugrundeliegenden Strategien zu erklären. Ein vollständiger Überblick würde hier zu weit führen, so dass wieder auf die Literatur [2, 5] verwiesen werden soll. Außerdem sollen hier nur globale Gütemaße betrachtet werden, die das gesamte Clusteranalyseergebnis bewerten und nicht nur einzelne Cluster.

Der Partitionskoeffizient [1] ist definiert durch

$$\frac{\sum_{i=1}^c \sum_{j=1}^n u_{ij}^2}{n}.$$

Je größer dieser Wert ist, desto besser sollte das Clusteranalyseergebnis sein. Der größte Wert 1 wird genau dann angenommen, wenn die Fuzzy-Clustereinteilung scharf ist, d.h. $u_{ij} \in \{0, 1\}$. Der kleinste Wert $1/c$ wird erreicht, wenn alle Datenvektoren allen Clustern mit dem gleichen Zugehörigkeitsgrad $1/c$ zugeordnet werden. Das bedeutet, dass ein Fuzzy-Clusteranalyseergebnis als umso besser angesehen wird, je schärfer die Einteilung ist.

Die Partitionsentropie [1]

$$\frac{\sum_{i=1}^c \sum_{j=1}^n u_{ij} \ln(u_{ij})}{n}$$

wurde der Shannon-Entropie nachempfunden. Je kleiner der Wert der Partitionsentropie, desto besser wird das Clusteranalyseergebnis angesehen.

In [4] werden Gütemaße eingeführt, die nicht nur die Zugehörigkeitsgrade, sondern auch die Volumina der Cluster berücksichtigen. Es sei

$$A_i = \frac{\sum_{j=1}^n u_{ij}^m (x_j - v_i)(x_j - v_i)^\top}{\sum_{j=1}^n u_{ij}^m}$$

die (Fuzzy-)Kovarianzmatrix des i ten Clusters, wobei

$$v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}$$

das Clusterzentrum des i ten Clusters ist und die x_j die Datenvektoren sind.

Das Gütemaß Fuzzy-Hypervolumen wird folgendermaßen definiert:

$$FHV = \sum_{i=1}^c \sqrt{\det(A_i)}.$$

Ein kleiner Wert von FHV weist auf kleine, kompakte und daher gute Cluster hin.

Die mittlere Partitionsdichte ist durch

$$\frac{1}{c} \sum_{i=1}^c \frac{S_i}{\sqrt{\det(A_i)}}$$

gegeben, wobei $S_i = \sum_{j \in Y_i} u_{ij}$ und

$$Y_j = \{j \in \{1, \dots, n\} \mid (x_j - v_i)^\top A^{-1} (x_j - v_i) < 1\}$$

die Menge der Datenvektoren in der Nähe des Clusterzentrums v_i ist. S_i entspricht der Anzahl der Datenvektoren, die dem Cluster i zugeordnet werden. Daher ist $\frac{S_i}{\sqrt{\det(A_i)}}$ proportional zur mittleren Dichte der Daten in Cluster i . Ein hoher Wert des Gütemaßes deutet auf eine gute Clustereinteilung.

Abschließend soll noch die Partitionsdichte

$$\frac{\sum_{i=1}^c S_i}{FHV}$$

erwähnt werden, bei der ein großer Wert wiederum auf eine gute Clustereinteilung hinweist.

Es soll noch einmal betont werden, dass Gütemaße wie der Partitionskoeffizient oder die Partitionsentropie die Bewertung der Clustereinteilung allein aufgrund der Zugehörigkeitsgrade vornehmen, während das Fuzzy-Hypervolumen und die (mittlere) Partitionsdichte zusätzlich die Abstände der Daten zu den Clustern berücksichtigen. Beide Prinzipien werden bei den folgenden Visualisierungsmethoden benutzt.

4 Visualisierung

Zunächst sollen nur die Zugehörigkeitsgrade zur visuellen Beurteilung eines Clusteranalyseergebnisses herangezogen werden. Das zugrundeliegende Prinzip besteht darin, dass eine Fuzzy-Clustereinteilung umso besser beurteilt wird, desto schärfer sie ist. Natürlich kann eine schärfere Einteilung immer durch eine kleinere Wahl des Fuzzifiers m erreicht werden, ohne dass das Ergebnis tatsächlich besser wird. Daher sollte der Fuzzifier immer mitberücksichtigt werden.

Zur Veranschaulichung der grafischen Gütemaße dient der in Abbildung 1 dargestellte Datensatz, der offenbar drei gut separierte Cluster enthält. Dieser Datensatz wurde mit dem bekannten Fuzzy-c-Means-Algorithmus geclustert, sowohl mit drei als auch mit vier Clustern. Das Ergebnis bei drei Clustern entspricht der intuitiv korrekten Einteilung der Daten. Abbildung 1 zeigt die Clusterzentren im Falle von vier Clustern.

Eine sehr einfache Visualisierungsmethode besteht in der Betrachtung der Verteilung der Zugehörigkeitsgrade. Im Idealfall einer scharfen Clustereinteilung wird jedes Datum eindeutig einem Cluster mit Zugehörigkeitsgrad 1 zugeordnet und allen anderen mit Zugehörigkeitsgrad 0. Man würde in diesem Fall eine relative Häufigkeit von $n/(cn) = 1/c$ beim Wert 1 und entsprechend $(c-1)n/(cn) = (c-1)/c$ beim Wert 0 erwarten. In diesem Fall ergäbe sich ein Histogramm, das einen Wert von $(c-1)/c$ auf der linken, $1/c$ auf der rechten Seite und dazwischen 0 zeigt.

Ein derartiges Histogramm wäre i.A. asymmetrisch und das Verhältnis von linker zu rechter Seite wäre abhängig von der Anzahl der Cluster. Insbesondere bei einer größeren Clusteranzahl würde der Wert auf der linken Seite alles andere stark dominieren. Daher führen wir eine Skalierung durch, bei der im Idealfall einer scharfen Einteilung links und rechts der Wert 1 stehen würde. Das bedeutet, dass bei der Zählung der (relativen) Häufigkeiten die Werte bei 0 mit dem Faktor $c/(c-1)$ und Werte bei 1 mit dem Faktor c multipliziert werden. Zur Darstellung des Histogramms wird die Gewichtung der Werte zwischen 0 und 1 entsprechend den beiden Extremwerten linear interpoliert. Ein einzelner Balken im Histogramm zeigt daher nicht die relativen Häufigkeiten von Zugehörigkeitsgraden in einem Bereich zwischen a und b

$$\frac{1}{n} \text{card}\{(i, j) \in \{1, \dots, c\} \times \{1, \dots, n\} \mid a \leq u_{ij} < b\},$$

sondern die skalierte Häufigkeit

$$\frac{1}{n} \sum_{(i,j): a \leq u_{ij} < b} \left(\frac{c(c-2)}{c-1} u_{ij} + \frac{c}{c-1} \right).$$

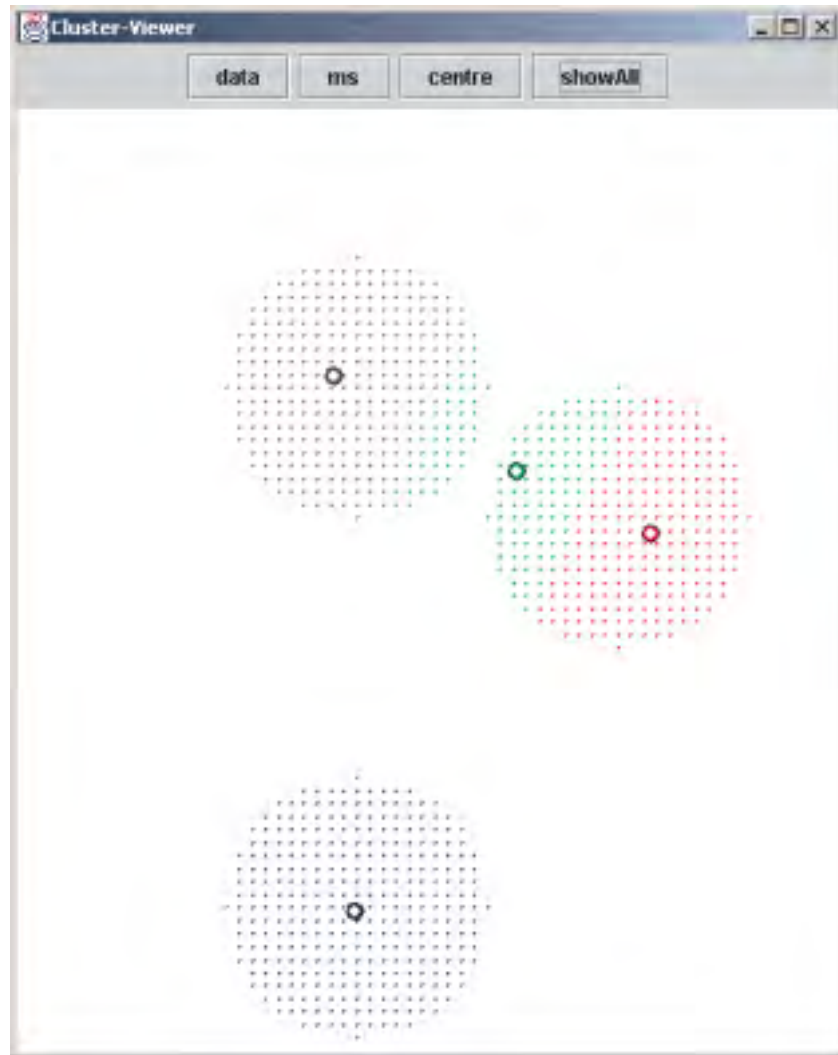


Abbildung 1: Clusteranalyse-Ergebnis bei vier Clustern

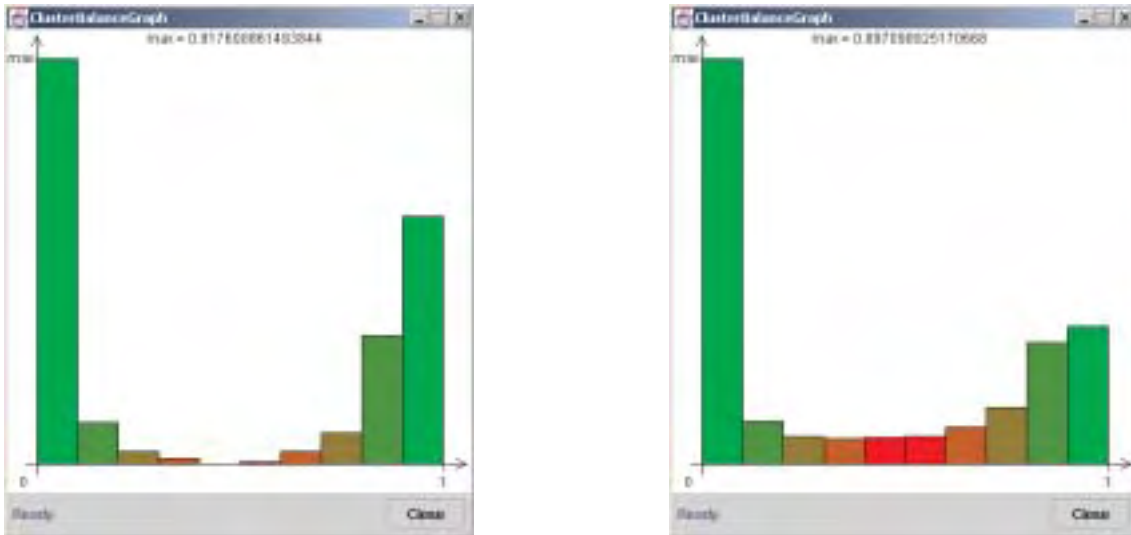


Abbildung 2: Skalierte Verteilungen der Zugehörigkeitsgrade für drei (links) und vier (rechts) Cluster

Abbildung 2 zeigt die skalierten Histogramme für den künstlichen Datensatz, wenn der Clusteralgorithmus von drei bzw. vier Clustern ausgeht. Der Höhe der Histogramme ist normalisiert, der Wert auf der linken Seite beträgt für das linke Histogramm 0.92, für das rechte Histogramm 0.90. Wesentlich signifikanter ist jedoch, dass bei der falschen Clusteranzahl wesentlich mehr Zugehörigkeitsgrade im mittleren und weniger im rechten Bereich auftreten, was auf ein eher schlechtes Clusteranalyseergebnis hindeutet.

Diese Histogramme beinhalten zwar schon wichtige Informationen. Es sollte aber zumindest noch ein weiteres rein auf Zugehörigkeitsgraden basierendes Diagramm betrachtet werden. Für jeden Datenvektor x_j wird das Cluster mit dem größten Zugehörigkeitsgrad – z.B. Cluster i_1 – und das Cluster mit dem zweitgrößten Zugehörigkeitsgrad – etwa i_2 – bestimmt. Dann wird für jeden Datenvektor ein Punkt mit den Koordinaten (u_{i_1j}, u_{i_2j}) gezeichnet, was zu einem Diagramm wie in Abbildung 3 führt.

Offensichtlich müssen alle Punkte in dem Dreieck liegen, das durch die Punkte $(0,0)$, $(0.5,0.5)$ und $(1,0)$ aufgespannt wird, da die erste Koordinate immer größer als die zweite sein muss und wegen der probabilistischen Nebenbedingung (2) zusätzlich $u_{i_1j} + u_{i_2j} \leq 1$ gelten muss.

Im Idealfall einer (nahezu) scharfen Clustereinteilung würden alle Punkte in der Nähe des Punktes $(1,0)$ liegen. Punkte in der Nähe von $(0.5,0.5)$ deuten auf mehrdeutige Datenvektoren hin, die etwa in der Mitte zwischen zwei Clustern liegen. Punkte bei $(0,0)$ stammen üblicherweise von Rauschdaten, die einen geringen Zugehörigkeitsgrad zu allen Clustern aufweisen.

Das obere linke Diagramm in Abbildung 3 zeigt die Situation zu Beginn der Clusteranalyse, wo die Clusterzentren zufällig initialisiert werden. Selbstverständlich ergibt sich für die Zufallsinitialisierung ein sehr schlechtes Bild und nur sehr wenige Punkte liegen in der Nähe des Idealpunktes $(1,0)$.

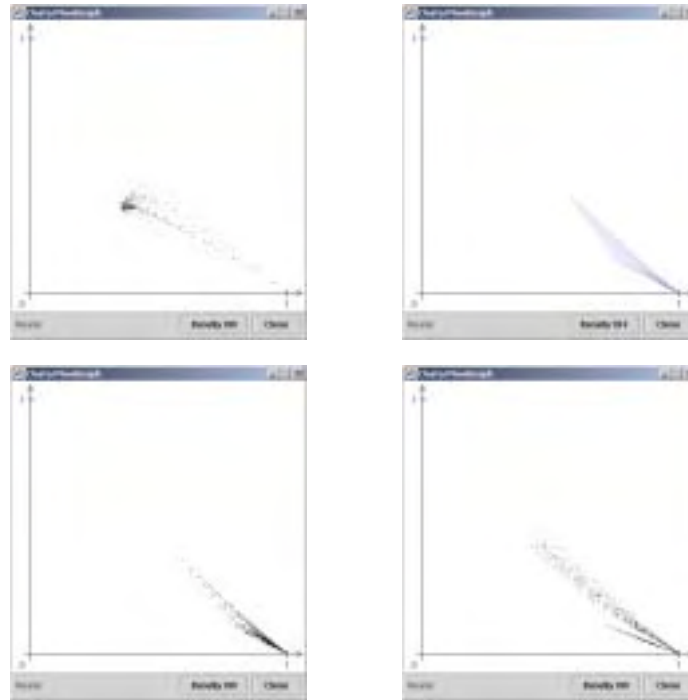


Abbildung 3: Maximale Zugehörigkeitsgrade für die zufällige Anfangsinitialisierung der Clusterzentren (oben links), für drei (unten links), vier (unten rechts) Cluster nach Konvergenz des Algorithmus und ein Intensitätsplot für einen großen Datensatz (oben rechts)

Der untere Teil von Abbildung 3 zeigt die Diagramme nach vollständiger Durchführung der Clusteranalyse mit drei bzw. vier Clustern. Die Clustereinteilung mit der korrekten Anzahl von drei Clustern weist deutlich mehr Werte in der Nähe des Idealpunktes (1,0) auf.

Im Fall sehr großer Datensätze sollte anstelle der einzelnen Punkte ein Intensitätsplot vorgezogen werden, bei dem die Farbe oder Intensität anzeigt, wie groß die Dichte der Punkte im entsprechenden Bereich ist. Das obere rechte Diagramm zeigt einen derartigen Intensitätsplot für einen ähnlichen Datensatz wie in Abbildung 1 mit einer deutlich erhöhten Datendichte mit insgesamt 192231 anstatt 951 Datenvektoren.

Bisher basierten die Visualisierungen allein auf den Zugehörigkeitsgraden. Im vorhergehenden Abschnitt wurde bereits erwähnt, dass mehr Information für die Beurteilung der Clustergüte herangezogen werden kann. Gütemaße wie das Fuzzy-Hypervolumen oder die (mittlere) Partitionsdichte berücksichtigen auch die Abstände der Datenvektoren zu den Clustern. Sehr nützlich ist die Betrachtung der Zugehörigkeitsgrade pro Cluster aufgetragen über den Distanzen. Für jedes Cluster i wird ein eigenes Diagramm erstellt, in dem für jeden Datenvektor x_j ein Punkt an der Stelle (d_{ij}, u_{ij}) gezeichnet wird. Dies führt zu Diagrammen, wie sie in Abbildung 4 zu sehen sind.

Wie sollte ein ideales Diagramm aussehen? Bei kleinen Distanzen sollten sich hohe, bei größeren Distanzen niedrige Zugehörigkeitsgrade ergeben. Es soll kurz erläutert werden, welche Effekte auftreten, wenn die Clusterzentren ungünstig gewählt werden, obwohl gute Clusterstrukturen in den Daten vorhanden sind. Typische Pro-

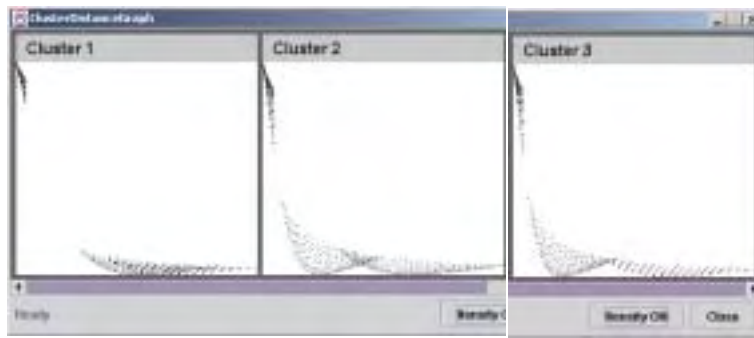


Abbildung 4: Zugehörigkeitsgrade aufgetragen über den Distanzen bei drei Clustern



Abbildung 5: Zugehörigkeitsgrade aufgetragen über den Distanzen bei vier Clustern

bleme, die in diesem Fall auftreten, sind die folgenden:

- Ein Cluster muss zwei oder mehr Datencluster abdecken. Dieser Fall tritt vor allem dann auf, wenn die Anzahl der Cluster zu klein gewählt worden ist. In diesem Fall würden sich nur wenige Daten im oberen linken Teil des Diagramms mit sehr geringem Abstand zu dem Cluster ergeben.
- Zwei oder mehr Cluster teilen sich dasselbe Datencluster. Dies tritt üblicherweise auf, wenn die Anzahl der Cluster zu groß gewählt wurde. In diesem Fall treten selbst für kleine Distanzen auch kleine Zugehörigkeitsgrade im linken unteren Teil des Diagramms auf.

Abbildung 4 zeigt die entsprechenden Diagramme für den künstlichen Datensatz, wenn drei Cluster verwendet werden. Alle drei Diagramme sehen sehr gut aus. Man erkennt sogar, dass das Cluster 1, das das untere Datencluster in Abbildung 1 abdeckt, am besten von allen anderen Clustern separiert ist.

Bei (der falschen Wahl von) vier Clustern ergeben sich die in Abbildung 5 dargestellten Diagramme. Cluster 1 deckt wiederum das untere Datencluster in Abbildung 1 ab und das Diagramm stimmt nahezu mit dem von Cluster 1 bei drei Clustern überein. Das Diagramm für Cluster 4 (mit dem Clusterzentrum oben links in Abbildung 5) sieht ebenfalls akzeptabel aus. Wie in Abbildung 1 zu sehen ist, deckt dieses Cluster auch einigermaßen korrekt das linke obere Datencluster ab. Allerdings deutet der relativ kontinuierliche Abfall der Zugehörigkeitsgrade an, dass das Cluster nicht sehr gut von den anderen Clustern separiert ist. Cluster 3 mit dem Clusterzentrum



Abbildung 6: Intensitätsplot für die Zugehörigkeitsgrade über den Distanzen für einen großen Datensatz

oben in der Mitte in Abbildung 1 ist das schlechteste Cluster. Selbst für kleine Distanzen treten trotzdem kleine Zugehörigkeitsgrade auf, da es sich seine Daten mit dem rechten Cluster 2 teilen muss. Außerdem gibt es eine Lücke im Bereich mittelgroßer Zugehörigkeitsgrade, was anzeigt, dass das Cluster anscheinend von zwei verschiedenen Datenclustern gespeist wird. Cluster 2 weist niedrige Zugehörigkeitsgrade auch für kleine Distanzen auf, da es sich seine Daten mit Cluster 3 teilen muss.

Wie bei den Grafen mit den maximalen Zugehörigkeitsgraden sollte im Fall großer Datensätze ein Intensitätsplot wie in Abbildung 6 bevorzugt werden.

Als weiteres Beispiel zur Illustration dient der in Abbildung 7 dargestellte künstliche Datensatz, bei dem Rauschdaten hinzugefügt wurden.

Abbildung 8 zeigt die (skalierte) Verteilung der Zugehörigkeitsgrade und das Diagramm für die maximalen Zugehörigkeitsgrade für den verrauschten Datensatz, wenn er mit drei Clustern geclustert wird. Diese Diagramme sollten denen in den Abbildungen 2 bzw. 3 gegenübergestellt werden. Die Gemeinsamkeiten und Unterschiede sind offensichtlich und brauchen nicht weiter erläutert zu werden. In Abbildung 9 sind die Zugehörigkeitsgrade über den Distanzen dargestellt (vgl. Abbildung 4 und 5).

5 Schlussbemerkungen

In diesem Beitrag wurden verschiedene Visualisierungstechniken zur Beurteilung eines Clusteranalyseergebnisses vorgestellt. Klassische Gütemaße können dazu verwendet werden, eine Clusteranalyse vollständig unüberwacht durchzuführen. Allerdings verlässt man sich dabei auf Verfahren, die eine extreme Informationskompression vornehmen. Im Gegensatz dazu beinhalten die grafischen Gütemaße wesentlich mehr Informationen und erlauben sogar die Beurteilung und Identifikation guter und schlechter Cluster.

Obwohl alle grafischen Gütemaße die Zugehörigkeitsgrade verwenden, lassen sie sich trotzdem auf scharfe (nicht-fuzzy) Clusteringverfahren anwenden. Dazu müssen nur

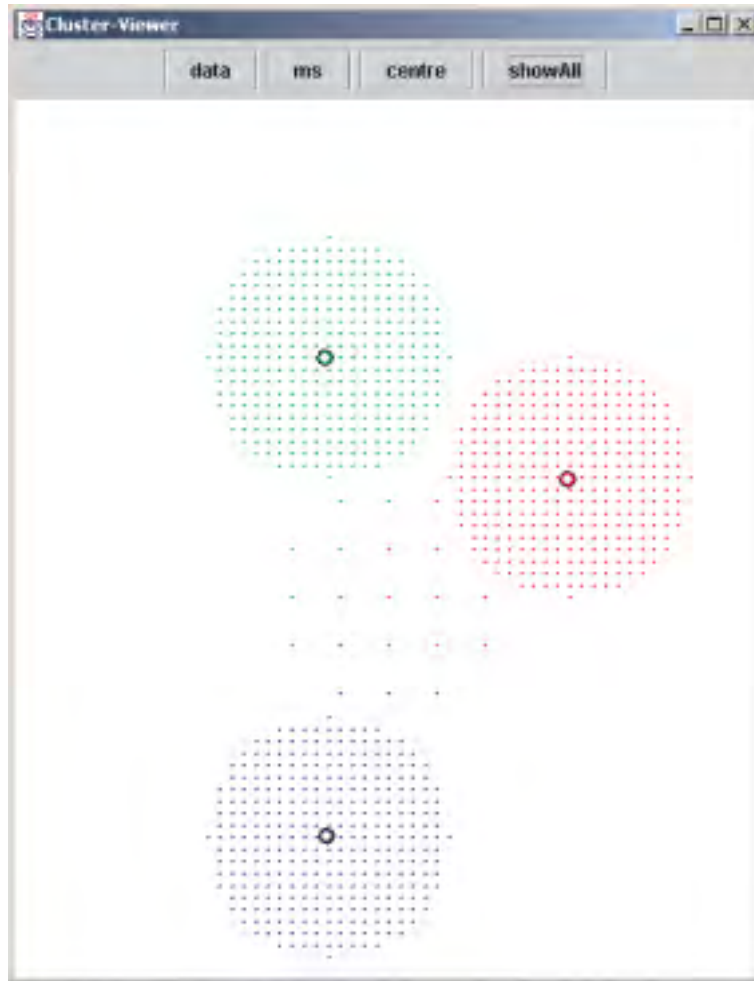


Abbildung 7: Clusteranalyseergebnis mit drei Clustern bei verrauschten Daten

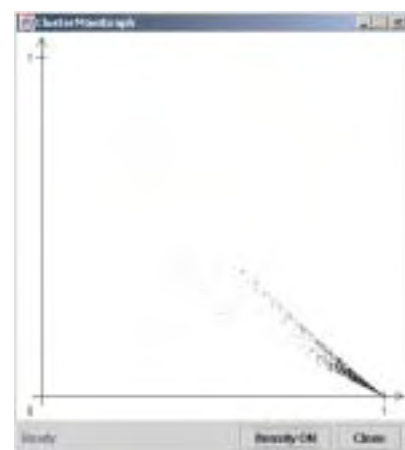
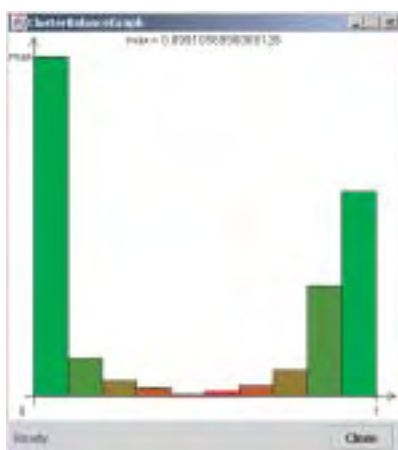


Abbildung 8: Skaliertes Histogramm und maximale Zugehörigkeitsgrade für den verrauschten Datensatz

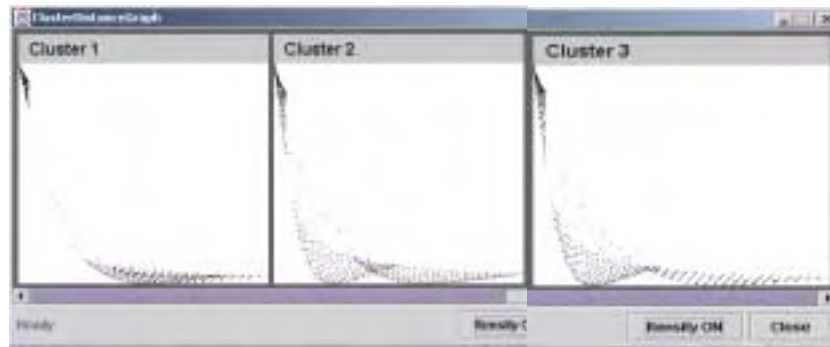


Abbildung 9: Zugehörigkeitsgrade aufgetragen über den Distanzen bei verrauschten Daten

nach der Durchführung der scharfen Clusteranalyse einmalig die Zugehörigkeitsgrade nach den üblichen Formeln berechnet werden, bevor die Visualisierungsmethoden angewendet werden können.

Danksagung: Großen Dank schulde ich Vera Chekhtman und Edgar Janz, die die Visualisierungskonzepte implementiert haben.

Literatur

- [1] Bezdek, J. C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York. 1981.
- [2] J.C. Bezdek, J. C.; Keller, J.; Krishnapuram, R.; Pal, N. R.: *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Kluwer, Boston. 1999.
- [3] Davé, R. N.: Characterization and Detection of Noise in Clustering. *Pattern Recognition Letters* 12 (1991), S. 657–664.
- [4] Gath, I; Geva, A. B.: Unsupervised Optimal Fuzzy Clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence* 11 (1989), S. 773–781.
- [5] Höppner, F.; Klawonn, F.; Kruse, R.; Runkler, T.: *Fuzzy Cluster Analysis*. Wiley, Chichester. 1999.
- [6] Krishnapuram, R.; Keller, J.: A Possibilistic Approach to Clustering. *IEEE Trans. on Fuzzy Systems* 1 (1993), S. 98–110.
- [7] Timm, H.; Borgelt, C.; Kruse, R.: A Modification to Improve Possibilistic Cluster Analysis. *IEEE Intern. Conf. on Fuzzy Systems Honolulu* (2002).

Datenbasierte Regelgenerierung mit modell-basierten Clusterverfahren

H. Kiendl, L. Haendel

Lehrstuhl für Elektrische Steuerung und Regelung

Universität Dortmund, 44221 Dortmund

Tel.: 0231 755 2760

kiendl@esr.e-technik.uni-dortmund.de *oder* kiendl@hyperfuzzy.de

haendel@esr.e-technik.uni-dortmund.de *oder* lars@haendels.de

Zusammenfassung

Betrachtet wird die Aufgabe der datenbasierten Modellierung von Prozessen für die Datenpunkte erhoben wurden. Jeder Datenpunkt besteht aus den Werten einer oder mehrerer Eingangsgrößen und einer dazugehörenden Ausgangsgröße. Das Modellierungsziel besteht darin, ausgehend von den Datenpunkten ein Modul zu *lernen*, das die Ausgangsgrößenwerte für zukünftige Datenpunkte, für die nur die Eingangsgrößenwerte bekannt sind, vorhersagt.

Der hier vorgestellte SMBC Algorithmus erweitert unüberwacht arbeitende modellbasierte Clusterverfahren in Richtung eines überwachten Generierens von Clustern und darauf basierenden WENN-DANN-Regeln. Cluster werden individuell nach Relevanz Gesichtspunkten bewertet. Dies erhöht ihre Interpretierbarkeit. Durch Mechanismen zum Löschen, Hinzufügen und Vereinigen von Clustern wird automatisch eine passende Clusteranzahl ermittelt. Die prinzipielle Leistungsfähigkeit des entwickelten Verfahrens wird an mehreren Benchmarkproblemen demonstriert.

1 Einleitung

Betrachtet wird die Aufgabe der datenbasierten Modellierung von künstlichen oder realen Prozessen für die Datenpunkte erhoben wurden. Jeder Datenpunkt $P(\mathbf{x}, y)$ besteht aus einem m dimensionalen Eingangsvektor \mathbf{x} und einem dazugehörenden Ausgangsgrößenwert y . Zu unterscheiden sind Regressions- und Klassifikationsaufgaben. Bei Regressionsaufgaben ist die Ausgangsgröße reellwertig, bei Klassifikationsaufgaben nimmt sie nur ganzzahlige Werte an, die die verschiedenen Klassen repräsentieren.

Das Modellierungsziel besteht darin, aus der zur Verfügung stehenden Menge von Datenpunkten ein Modul zu *lernen*, das die Ausgangsgrößenwerte für zukünftige Datenpunkte, für die nur die Eingangsgrößenwerte bekannt sind, vorhersagt. Fuzzy-Systeme sind zur Lösung dieser Aufgabe prinzipiell geeignet; sie modellieren das Prozessverhalten mit Hilfe einer aus WENN-DANN-Regeln bestehenden Regelbasis. Für die datenbasierte Generierung von Mamdani-Fuzzy-Systemen ist das Fuzzy-ROSA-Verfahren [KK94, Kro99, Sla01] entwickelt worden. Als wesentliche Einschränkung sind die mit dem Fuzzy-ROSA-Verfahren generierten Regelprämissen allerdings nur in der Lage, achsparallele Gebiete des Eingangsraumes zu adressieren. Um diese Einschränkung zu überwinden, werden hier geeignete Clusterverfahren entwickelt.

Fuzzy-Clusterverfahren partitionieren eine gegebene Menge von Datenpunkten in mehrere Gruppen, die als Cluster bezeichnet werden. Datenpunkte können dabei gleichzeitig zu mehreren Clustern gehören. Die Datenpunkte sollen dabei innerhalb eines Clu-

sters möglichst ähnlich und zwischen verschiedenen Clustern möglichst unähnlich zu einander sein. Modell-basierte Clusterverfahren wie der *Fuzzy-C-Means* (FCM) oder der *Gustafson-Kessel* (GK) Algorithmus [GK79, HKK97, JMF00] arbeiten mit einer a priori festzulegenden Anzahl von Clustern und kennzeichnen diese durch ein Zentrum und – beim GK Algorithmus – durch eine Kovarianzmatrix, welche den Schwerpunkt bzw. die Ausdehnung der dem Cluster zugeteilten Datenpunkte beschreiben. Als Maß für die Ähnlichkeit von Datenpunkten wird der Abstand derselben verwendet. Zu gegebenen Daten werden die Clusterparameter iterativ mit einem *Expectation-Maximization* (EM) Algorithmus [DLR77] bestimmt: Die Clusterparameter werden geeignet initialisiert. Im sogenannten *E-Schritt* wird die Zuteilung jedes Datenpunktes zu jedem Cluster abstandsabhängig bestimmt. Datenpunkte werden dabei umso mehr einem Cluster zugeteilt, umso näher sie ihm sind. Im darauf folgenden sogenannten *M-Schritt* werden die Clusterparameter aufgrund der Zuteilungen neu ermittelt. Die beiden Schritte werden bis zur Erfüllung eines Abbruchkriteriums wiederholt.

2 Grundidee

FCM und GK Algorithmus arbeiten *unüberwacht*, d.h. ohne besondere Berücksichtigung einer Ausgangsgröße. Im Kontext der datenbasierten Regelgenerierung ist es jedoch gewünscht, Cluster zu erhalten, die sich durch möglichst homogene bzw. gleiche Ausgangsgrößenwerte auszeichnen und es erscheint sinnvoll, dieses bereits während des Clusterprozesses zu berücksichtigen. Ferner ist die Anzahl der zu verwendenden Cluster selten im vorhinein bekannt und sollte sich deshalb während des Clusterprozesses selbständig einstellen. Daher werden FCM bzw. GK Algorithmus durch die von uns entwickelten Strategieelemente wie folgt erweitert und das daraus resultierende Verfahren wird fortan als SMBC¹ Algorithmus bezeichnet.

- Erweiterung der Repräsentation der Cluster um einen Ausgangsgrößenwert
- Modifikation der Zuteilung von Datenpunkten, die bezüglich des Ausgangsgrößenwertes *unpassend* zu einem Cluster sind
- automatische Ermittlung einer passenden Clusteranzahl
 - durch Einfügen neuer Cluster in Gebiete des Eingangsraums, in denen inhomogene Cluster, d.h. Cluster denen Datenpunkte mit stark unterschiedlichen Ausgangsgrößenwerten zugeteilt wurden, liegen
 - Übertragung und Einsatz der Relevanzkonzepte des Fuzzy-ROSA-Verfahrens zur Bewertung von Clustern. Löschen von Clustern mit besonders schlechter Bewertung und Vereinigen von Clustern, wenn dies anhand der resultierenden Relevanzbewertung sinnvoll erscheint.

3 Basisalgorithmus

Im folgenden wird die auf dem FCM Algorithmus basierende Basis-Variante des SMBC Algorithmus beschrieben.

¹engl.: supervised modell-based-clustering

3.1 Clusterprototypen und Partitionsmatrix

Ein Cluster wird beschrieben durch ein im Eingangsraum definiertes sogenanntes Clusterzentrum \mathbf{v} und durch einen zugeordneten Ausgangsgrößenwert c . Die Zuteilung jedes Datenpunktes der zu clusternden Datenstichprobe zu jedem Cluster sei zusammengefaßt in der sogenannten *Partitionsmatrix* \mathbf{U} . Das Element $u_{i,t}$ dieser Matrix kennzeichnet die Zuteilung des Datenpunktes P_i zum Cluster C_t . Die Partitionsmatrix hat genauso viele Zeilen wie Datenpunkte in der zu clusternden Stichprobe enthalten sind und genauso viele Spalten wie Cluster vorhanden sind. Für jeden Datenpunkt hat die über alle Cluster gebildete Summe der Zuteilungen den Wert 1. Ordnet man einem Datenpunkt P_i die Masse 1 zu, so läßt sich $u_{i,t}$ auch als der Anteil interpretieren, der dem Cluster t von der Masse des Datenpunktes P_i zugeteilt wird.

Zu einer Stichprobe werden bei gegebener Partitionsmatrix die Clusterzentren und Ausgangsgrößenwerte jedes Clusters als eine Art Schwerpunkt der zugeteilten Datenpunktmassen gemäß

$$v_{t,j} = \frac{\sum_{i=1}^N u_{i,t}^\alpha x_{i,j}}{\sum_{i=1}^N u_{i,t}^\alpha} \quad (1)$$

$$c_t = \frac{\sum_{i=1}^N u_{i,t}^\alpha y_i}{\sum_{i=1}^N u_{i,t}^\alpha} \quad (2)$$

berechnet. Dabei bezeichnet der Index j die jeweilige Eingangsgröße und der Index t das jeweilige Cluster. N ist die Anzahl der Datenpunkte und α ist ein wählbarer *Fuzziness-Parameter*, für den wir den üblicherweise verwendeten Wert $\alpha = 2$ wählen.

Der Abstand eines Datenpunktes zu einem Cluster wird gemäß

$$d^2(\mathbf{x}, \mathbf{v}) = \sum_{j=1}^m (x_j - v_j)^2 \quad (3)$$

im Eingangsraum als euklidischer Abstand zwischen dem Eingangsvektor \mathbf{x} des Datenpunktes und dem Zentrum \mathbf{v} des Clusters ermittelt.

Zu einer Stichprobe werden bei gegebenen Clustern C_t die Elemente $u_{i,t}$ der Partitionsmatrix nach der Vorschrift

$$u_{i,t} = \frac{1}{\sum_{h=1}^K \left(\frac{d_{i,t}}{d_{i,h}} \right)^{\frac{2}{\alpha-1}}} \quad (4)$$

ermittelt. Dabei ist $d_{i,t}$ der nach Gl. (3) ermittelte Abstand des Datenpunktes $P_i = (\mathbf{x}_i, y_i)$ zum Clusterzentrum \mathbf{v}_t und K ist die Anzahl der Cluster.

Die Zuteilung eines Datenpunktes zu einem Cluster ist hiernach um so größer, je geringer der Abstand des Datenpunktes zum betrachteten Cluster – relativ zu den Abständen zu den anderen Clustern – ist.

3.2 Modifikationsmatrix

Die Partitionsmatrix wird unüberwacht, d.h. ohne Berücksichtigung der Ausgangsgrößenwerte der jeweiligen Datenpunkte und der Cluster ermittelt. Anhand des Ausgangsgrößenwertes eines Clusters ist jedoch feststellbar, wie gut ein Datenpunkt hinsichtlich seines Ausgangsgrößenwertes zu einem Cluster *paßt*. Im folgenden wird die Idee verfolgt, diese Information bereits während des Clusterprozesses auszunutzen. Hierzu wird die Zuteilung eines Datenpunktes mittels eines Modifikationsfaktors ω abgeschwächt, wenn sich die Ausgangsgrößenwerte des Datenpunktes und des Clusters stark voneinander unterscheiden. Zur Ermittlung der Modifikationsfaktoren werden im folgenden für den Fall der Regression bzw. der Klassifikation unterschiedliche Vorgehensweisen beschrieben. Analog zur Partitionsmatrix werden die Modifikationsfaktoren in der Matrix Ω zusammengefaßt. Das Element $\omega_{i,t}$ gibt an, in welchem Maße die Zuteilung des Datenpunktes P_i zum Cluster C_t abzuschwächen ist.

Anstelle der ursprünglichen Partitionsmatrix wird dann in den Gln. (1) und (2) zur Bestimmung der Clusterparameter die modifizierte Partitionsmatrix \mathbf{U}_{mod} verwendet, die sich ergibt, indem die ursprüngliche Partitionsmatrix \mathbf{U} elementweise mit der Modifikationsmatrix Ω multipliziert wird. Gegebenenfalls ist es sinnvoll, die modifizierte Partitionsmatrix so zu re-normieren, daß die Summe der Elemente jeder Zeile wieder den Wert 1 ergibt.

Regression Bei hinreichend gleichmäßiger Verteilung der Ausgangsgrößenwerte der Datenpunkte wird die Ähnlichkeit zweier Ausgangsgrößenwerte anhand eines Gaußkerns nach

$$\omega_{i,t} = \exp - \left(\frac{c_t - y_i}{2\sigma\beta} \right)^2 \quad (5)$$

bewertet. Dabei bezeichnet σ die auf der zu clusternden Stichprobe berechnete Standardabweichung der Ausgangsgröße und β ist ein einzustellender Strategieparameter des Verfahrens. Er bestimmt die Breite des Ausgangsgrößenintervalls, die von einem Cluster abgedeckt wird.

Bei stark unterschiedlicher Verteilung der Ausgangsgrößenwerte der Datenpunkte kann es vorteilhaft sein, die Ausgangsgröße in einigen Wertebereichen genauer als in anderen aufzulösen. Dann empfiehlt es sich, den Parameter β – basierend auf der Verteilung der Ausgangsgrößenwerte – für den jeweiligen Ausgangsgrößenwert des Clusters gemäß

$$\omega_{i,t} = \begin{cases} \exp - \left(\frac{c_t - y_i}{\beta_{l,t}} \right)^2 & y < c \\ \exp - \left(\frac{c_t - y_i}{\beta_{r,t}} \right)^2 & y \geq c \end{cases} \quad (6)$$

individuell anzupassen. Die angepaßten Werte für $\beta_{l,t}$ und $\beta_{r,t}$ werden durch

$$\beta_{l,t} = c_t - y_l \quad \text{und} \quad \beta_{r,t} = y_r - c_t \quad (7)$$

festgelegt. Dabei werden y_l und y_r so bestimmt, daß innerhalb der Intervalle $[y_l, c]$ bzw. $[c, y_r]$ jeweils ungefähr n Datenpunkte enthalten sind. Der Wert des Parameters

n ist über den einzustellenden Strategieparameters γ an die Anzahl der zu clusternden Datenpunkte N und an die aktuelle Clusteranzahl K gemäß

$$n = \frac{\gamma}{K}N \quad (8)$$

angekoppelt.

Klassifikation Im Falle der Klassifikation sind die vorkommenden Ausgangsgrößenwerte der Datenpunkte ganzzahlig und es existiert keinerlei Ordnung der Zahlenwerte wie es beispielsweise bei einer ordinalen Größe der Fall wäre. Daher werden die Modifikationsfaktoren durch

$$\omega_{i,t} = \begin{cases} 1 & y_i = c_t \\ 0 & \text{sonst} \end{cases} \quad (9)$$

festgelegt. Dies hat zur Folge, daß sich der Ausgangsgrößenwert eines Clusters nach dessen Initialisierung nicht mehr ändern kann.

3.3 Clusterprozeß

Zu gegebenen Datenpunkten werden die Clusterparameter wie bereits in Abschnitt 1 beschrieben mit einem iterativen EM Algorithmus ermittelt: Zunächst werden K_{Start} Startcluster geeignet initialisiert, z.B. indem zufällig K_{Start} Datenpunkte der Stichprobe ausgewählt und die Eingangsraumpositionen und Ausgangsgrößenwerte dieser Datenpunkte als Clusterzentrum und Clusterausgangsgrößenwert verwendet werden. Anschließend wird im E-Schritt zu den aktuellen Clusterparametern die modifizierte Partitionsmatrix anhand von Gl. (4) und einer der Gln. (5), (6) oder (9) berechnet. Darauf folgend werden im M-Schritt die Clusterparameter anhand von Gl. (1) und (2) neu ermittelt. Um die Anzahl der Cluster automatisch den Eigenarten der jeweiligen Datenstichprobe anzupassen, werden im Anschluß an den M-Schritt die in den Abschnitten 3.6 und 3.7 beschriebenen Mechanismen zum Löschen, Hinzufügen und Vereinigen von Clustern eingesetzt. Damit sind alle Teilschritte eines Lernschrittes beendet und es werden bis zum Erreichen eines geeignet definierten Abbruchkriteriums dann diese Schritte ab dem E-Schritt wiederholt.

3.4 Prognosemechanismus

Jedes Cluster C_t ist bekanntlich direkt in eine Fuzzy-Regel der Form

$$\text{WENN } \textit{Eingangsvektor innerhalb des Clusters} \text{ DANN } y = c_t \quad (10)$$

überführbar. Der Aktivierungsgrad μ_t der Prämisse, d.h. das Maß nach dem bestimmt wird, in welchem Grade ein Eingangsvektor \mathbf{x} innerhalb eines Clusters C_t liegt, wird durch

$$\mu_t = \exp(-d_t^2) \quad (11)$$

definiert. Dabei bezeichnet d_t den nach Gl. (3) berechneten Abstand von Eingangsvektor und Clusterzentrum.

Wenn zur Verknüpfung mehrerer linguistischer Terme in der Prämisse als UND-Operator die Multiplikation verwendet wird, kann die Prämisse auch auf die folgende, üblichere Form

$$\text{WENN } x_1 \text{ ist } A_1 \text{ UND } \dots \text{ UND } x_n \text{ ist } A_n \quad \text{mit } A_j = \exp(-d_j^2) \quad (12)$$

gebracht werden.

Wir verwenden bei Regressionsaufgaben zur Akkumulation die gewöhnliche Summe und zur Defuzzifizierung die *Schwerpunkt*-Methode (COG). Bei Klassifikationsaufgaben verwenden wir dagegen zur Akkumulation und zur Defuzzifizierung jeweils die *Maximum*-Methode.

3.5 Relevanzbewertung

Im Fuzzy-ROSA-Verfahren sind zur Bewertung der Relevanz einer Regel mehrere statistisch motivierte Kriterien entwickelt worden. Der Grundgedanke besteht darin, jede einzelne Regel danach zu bewerten, ob sie einen signifikanten Zusammenhang des Ein-Ausgangsverhaltens beschreibt. Die Übertragung des Relevanzkonzeptes auf eine Regel, die wie beschrieben einem Cluster zugeordnet wird, erfolgt hier derart, daß die Homogenität der Ausgangsgrößenwerte der Datenpunkte innerhalb eines Clusters betrachtet wird. Gemessen wird der Homogenitätsgrad als Trefferquote

$$\varrho_t = \frac{\sum_{i=1}^N \omega_{i,t} \mu_{i,t}}{\sum_{i=1}^N \mu_{i,t}}, \quad (13)$$

die sich als das Verhältnis der Summe der mit den Modifikationsfaktoren gewichteten Zugehörigkeiten der Datenpunkte zum Cluster C_t zur Summe der ungewichteten Zugehörigkeiten ergibt.

Man beachte, daß hier in Gl. (13) nicht die Zuteilung sondern die Zugehörigkeit der Datenpunkte zu den Clustern verwendet wird, damit jedes Cluster unabhängig von den anderen bewertet wird. Dies ist für den im Abschnitt 3.7 beschriebenen Vereinigungsmechanismus und für die Interpretierbarkeit der generierten Cluster von Bedeutung.

Die bei der Relevanzbewertung der Cluster erhaltenen Werte ϱ_t können als Glaubensgrad bei der Verarbeitung des resultierenden Regelsatzes verwendet werden.

3.6 Löschen und Hinzufügen von Clustern

Aufgrund ihrer Äquivalenz zu Regeln sollten Cluster gelöscht werden, wenn ihre Relevanzbewertung zu schlecht ist oder wenn die Summe der Zuteilungen zu gering ist. Für diese beiden Löschbedingungen sind passende Schwellwerte ϱ_{min} und u_{min}^α zu wählen². Dort wo Datenpunkte liegen, die in – bezüglich der Ausgangsgrößenwerte – inhomogenen Clustern zusammengefaßt sind, werden gezielt neue Cluster eingefügt: Das Konzept

²Bei Berechnung der Clusterparameter werden in Gl. (1) und (2) die mit α potenzierten Zuteilungen verwendet. Dementsprechend bezieht sich der Schwellwert u_{min}^α auf die Summe $\sum_{i=1}^N u_{i,t}^\alpha$.

der Modifikationsfaktoren induziert, daß dem einzelnen Datenpunkt sogenannte *Verlustmasse*³ zugeordnet wird. Dies ist die Differenz zwischen der ursprünglichen Masse 1 eines Datenpunktes und der Summe der Teilmassen, die sich aufgrund der modifizierten Zuteilung ergeben. Am Ende eines Lernschritts wird ein Datenpunkt, für den sich eine von Null verschiedene Verlustmasse ergeben hat, zufällig ausgewählt. Die Auswahlwahrscheinlichkeit wird proportional zu den Verlustmassen der Datenpunkte angesetzt. Der ausgewählte Datenpunkt dient zur Initialisierung eines neuen Clusters in der aktuellen Clusterpopulation. Wie auch bei der Initialisierung wird die Eingangsraumposition des Datenpunktes als neues Clusterzentrum und die Ausgangsgröße des Datenpunktes als Clusterausgangswert verwendet.

3.7 Vereinigen von Clustern

Eine geringere Anzahl von Clustern bedeutet ein weniger komplexes Modell und wirkt sich positiv auf die Interpretierbarkeit und oft auch auf die Generalisierungsfähigkeit und damit auf die Prognosegüte aus. Deshalb werden eng benachbarte Cluster, deren zugeweilte Datenpunkte ähnliche Ausgangsgrößenwerte aufweisen und denen damit ähnliche Clusterausgangswerte zugewiesen worden sind, als Kandidaten für eine Vereinigung angesehen.

Zwei Cluster werden hier miteinander vereinigt, wenn die Relevanzbewertung des vereinigten Clusters nicht schlechter ist, als die beste Relevanzbewertung der bisherigen einzelnen Cluster. Die Vereinigung zweier Cluster geschieht, indem die entsprechenden Spalten der modifizierten Partitionsmatrix addiert und anschließend die Clusterparameter und die Relevanzbewertung neu berechnet werden.

Die Cluster, für die eine Vereinigung versucht wird, werden wie folgt ausgewählt: Für jedes Cluster C_t wird aus der Menge der Cluster gleichen oder⁴ ähnlichen Ausgangsgrößenwertes das am nächsten liegende sogenannte *Partnercluster* ermittelt. Die Cluster werden mit zunehmenden Abstand zu ihrem Partnercluster in einer Rangliste sortiert⁵. Nun wird die Rangliste schrittweise durchgegangen und versucht, das jeweilige Cluster mit seinem Partnercluster zu vereinigen, wenn keines der beiden betrachteten Cluster bisher vereinigt wurde und somit nicht mehr in der Clusterpopulation existiert.

4 Details und Erweiterungen

4.1 Geometrieparameter

Beim FCM Algorithmus werden die Cluster lediglich durch ein Clusterzentrum gekennzeichnet; die Geometrie der dem Cluster zugeweilten Datenpunkte wird nicht berücksichtigt. Damit impliziert das verwendete Clustermodell sphärische, kugelförmige Cluster gleicher Größe. Im Gegensatz dazu werden die Clusterprototypen beim GK Algorithmus durch Bestimmung der Kovarianzmatrix anhand der zugeweilten Datenpunkte

³Diese wird unabhängig von einer eventuellen Re-Normalisierung der Partitionsmatrix nachgehalten.

⁴Im Falle der Klassifikation werden nur die Cluster mit gleichem Ausgangsgrößenwert betrachtet. Im Falle der Regression werden zu C_t dagegen alle Cluster betrachtet, deren Ausgangsgrößenwert – hier y_i genannt – in Verbindung mit dem Ausgangsgrößenwert c_t des Clusters C_t einen Modifikationsfaktor ergibt, der oberhalb eines festzulegenden Schwellwertes ω_{min} liegt.

⁵Sollte ein Cluster C_t kein Partnercluster besitzen wird dieses Cluster ignoriert.

durch zusätzliche Geometrieparameter charakterisiert. Dabei werden die Elemente der Kovarianzmatrix⁶ bei gegebener Stichprobe durch die Partitionsmatrix⁷ \mathbf{U} zu

$$\mathbf{Q}_t = \sum_{i=1}^N u_{i,t}^\alpha (\mathbf{x}_i - \mathbf{v}_t)^T (\mathbf{x}_i - \mathbf{v}_t) \quad (14)$$

bestimmt. Ferner wird für die Berechnung der Partitionsmatrix anstelle des euklidischen Abstandes zwischen Clusterzentrum und Datenpunkt für jedes Cluster eine Art individuelle Mahalanobis-Distanz

$$d_{i,t}^2 = \sqrt[m]{\det(\mathbf{Q}_t)} (\mathbf{x}_i - \mathbf{v}_t)^T \mathbf{Q}_t^{-1} (\mathbf{x}_i - \mathbf{v}_t) \quad (15)$$

verwendet. Der Term $\sqrt[m]{\det(\mathbf{Q}_t)}$ bewirkt eine Normalisierung der Clustergröße. Damit impliziert das beim GK Algorithmus verwendete Clustermodell ellipsoide Cluster beliebiger Orientierung aber gleicher Größe. Für die Berechnung der Zugehörigkeit eines Datenpunktes zu einem Cluster wird jedoch auf diese Normierung verzichtet⁸. Für die Startinitialisierung der Geometrieparameter wird die Einheitsmatrix verwendet. Beim Hinzufügen eines neuen Clusters werden die Geometrieparameter vom nächstliegenden Cluster übernommen.

Die Anzahl der zu berechnenden Parameter der Kovarianzmatrix steigt quadratisch mit der Dimensionalität des Problems. Damit steigt gleichermaßen die Anzahl der Datenpunkte, die notwendig sind, um die Parameter mit einer bestimmten statistischen Robustheit zu bestimmen. Ebenso ist es bei Verwendung der vollen Kovarianzmatrix nicht mehr möglich, die den Clustern äquivalenten Fuzzy-Regeln auf die Form (12) zu bringen, da aufgrund der Drehung des Koordinatensystems keine komponentenweisen Abstände mehr ermittelt werden können. Oft werden daher nur die Diagonalelemente der Kovarianzmatrix ermittelt und sämtliche Elemente der Nebendiagonalen zu Null gesetzt, womit die Clusterprototypen jedoch wieder nur achsparallele Strukturen implizieren. Die Diagonalelemente sind bis auf einen zu vernachlässigenden konstanten Faktor identisch mit den komponentenweise berechneten Varianzen der den Cluster zugeteilten Datenpunkte bezüglich des Clusterzentrums.

Ein allgemeinerer von Banfield & Raftery entwickelter Weg wird in [BR93, FR98] beschrieben. Durch Re-Parametrisierung der Kovarianzmatrix mittels Singulärwertzerlegung ist es möglich, sehr flexibel eine große Vielzahl von verschiedenen Clustermodellen mit der kennzeichnenden Matrix

$$\mathbf{Q}_t = \mathbf{D}_t \mathbf{\Lambda}_t \mathbf{D}_t^T \quad (16)$$

aufzustellen. Dabei ist \mathbf{D}_t eine orthogonale Matrix mit den Eigenvektoren der Matrix \mathbf{Q}_t . Die Matrix $\mathbf{\Lambda}_t$ ist eine Diagonalmatrix mit den Eigenwerten der Matrix \mathbf{Q}_t als Diagonalelementen; sie kann weiter in $\mathbf{\Lambda}_t = \lambda_t \mathbf{A}_t$ zerlegt werden. Die Matrix \mathbf{D}_t bestimmt damit die Orientierung, die Matrix \mathbf{A}_t die Form und der Parameter λ_t die

⁶Gustafson & Kessel verwendeten ursprünglich sogenannte Fuzzy-Kovarianzmatrizen, die sich nach Gl. (14) ergeben wenn zusätzlich sämtliche Elemente durch $\sum_{i=1}^N u_{i,t}^\alpha$ dividiert werden. Siehe hierzu [HKK97] Seite 43.

⁷Wir verwenden hier die modifizierte Partitionsmatrix \mathbf{U}_{mod} .

⁸Die Zugehörigkeiten werden zur Relevanzbewertung der Cluster und zur Auswertung des äquivalenten Regelsatzes verwendet.

Größe des Clusterellipsoids. Jeder Term wird entweder individuell für jedes Cluster einzeln ermittelt, als Mittel über alle Cluster berechnet oder aber a priori festgelegt.

4.2 Skalierbarkeit

Von großer Bedeutung für die Abschätzung der praktischen Anwendbarkeit eines Lernverfahrens für ein gegebenes Problem ist die sogenannte *Ordnung*. Sie gibt an, wie sich die Laufzeit bzw. der Speicherplatzbedarf von einem Algorithmus mit zunehmender Größe des zu bearbeitenden Problems erhöht. Die Größe des Problems wird hier durch die Anzahl der Datenpunkte N und die Anzahl der Eingangsgrößen m bestimmt. Wenn mit einer festen maximalen Anzahl an Lernschritten und einer Obergrenze für die Anzahl der Cluster gearbeitet wird, hat der SMBC Algorithmus eine jeweils mit der Anzahl der Datenpunkte linear und mit der Anzahl der Eingangsgrößen kubisch zunehmende Ordnung $O(Nm^3)$. Dies ergibt sich wie folgt: Für jeden Lernschritt müssen einmalig die Partitions- und Modifikationsmatrix berechnet und daran anschließend die Clusterparameter neu ermittelt werden. Diese Operationen skalieren linear mit der Anzahl der Datenpunkte und der Cluster. Für die Berechnung der Abstände der Datenpunkte zu den Clustern ist die Inversion einer $m \times m$ Matrix erforderlich, die bei Verwendung eines üblichen Gauß'schen Eliminationsverfahrens eine Ordnung von $O(m^3)$ aufweist⁹. Allerdings ist zu beachten, daß die obigen Voraussetzungen mit steigender Stichprobengröße ggf. nicht mehr zu rechtfertigen sind und es sinnvoll sein kann, eine größere Anzahl an Lernschritten und Clustern vorzusehen.

4.3 RBF-Netze

Funktionsweise RBF-Netze [Orr96] modellieren den Zusammenhang zwischen der Ausgangsgröße und den Eingangsgrößen durch eine Linearkombination

$$y = f(\mathbf{x}) = \sum_{t=1}^K c_t^* h_t(\mathbf{x}). \quad (17)$$

von K Teilfunktionen h_t . Als Teilmodell werden häufig durch einen Mittelpunkt \mathbf{v} und skalare oder komponentenweise Radien r bzw. \mathbf{r} definierte Kernelfunktionen gewählt. Ein einfaches Beispiel für den eindimensionalen Fall ist

$$h(x) = \exp\left(-\frac{(x - v)^2}{r^2}\right). \quad (18)$$

Die Erweiterung für den mehrdimensionalen Fall ist trivial. Neben Exponentialfunktionen werden in der Literatur verschiedene andere Funktionen untersucht. Beispielsweise schlägt Orr [Orr96] den allgemeinen Ansatz

$$h(\mathbf{x}) = F(d^2) \quad \text{mit} \quad d^2 = (\mathbf{x} - \mathbf{v})^T \mathbf{R}^{-1} (\mathbf{x} - \mathbf{v}) \quad (19)$$

⁹Je nach verwendeten Clustermodell ist die Inversion einer Matrix nicht erforderlich bzw. bei einer Diagonalmatrix mit einer Ordnung von $O(m)$ durchführbar, womit sich die Ordnung des Verfahrens auf $O(Nm)$ reduziert.

für die Modelle $h(\mathbf{x})$ vor, wobei \mathbf{R} eine geeignet definierte und invertierbare Matrix und F irgendeine eindimensionale Kernelfunktion ist, die vorzugsweise kontinuierlich, beschränkt und integrierbar zu 1 zu wählen ist.

Das Lernen eines RBF-Netzes wird oft auf die Bestimmung günstiger Gewichtungsfaktoren für einen festen Kanon von möglichen Kernelfunktionen beschränkt. Für eine bestimmte Auswahl aus der Menge der möglichen Kernelfunktionen ist dann die Bestimmung optimaler Gewichtungsfaktoren durch Lösung eines linearen Gleichungssystems mittels der Pseudoinversen¹⁰ durchführbar. Die Initialisierung, d.h. die Vorgabe möglicher Kernelfunktionen erfolgt u.a. durch Clusterverfahren. Alternativ werden alle Lerndatenpunkte als mögliche Kernelmittelpunkte verwendet. Die Ausdehnungsparameter ergeben sich dann – zumeist sehr einfach – aus statistischen Kennwerten der Lerndaten.

Einordnung und Vergleich Das aus Abschnitt 3.4 im Falle der Regression resultierende Modul liefert zu einem gegebenen Eingangsvektor \mathbf{x} den prognostizierten Ausgangsgrößenwert y in der Form

$$y(\mathbf{x}) = \frac{\sum_{t=1}^K c_t \mu_t(\mathbf{x})}{\sum_{t=1}^K \mu_t(\mathbf{x})}. \quad (20)$$

Darin sind c_t die den Clustern C_t zugeordneten Ausgangsgrößenwerte und $\mu_t(\mathbf{x})$ die Zugehörigkeitsgrade des Vektors \mathbf{x} zu den einzelnen Clustern C_t . Kennzeichnend für unseren Ansatz ist, daß sich die Werte c_t aus dem Clusterprozeß ergeben. Dieser ist – wie beschrieben – im Interesse einer guten Interpretierbarkeit der resultierenden Regeln so angelegt, daß sich möglichst homogene Cluster ergeben. Dies sind Cluster, die möglichst nur Datenpunkte mit gleichen oder ähnlichen Ausgangsgrößenwerten abdecken. Dementsprechend lassen sich die Funktionen $\mu_t(\mathbf{x})$ und die Werte c_t als lokale Charakterisierungen der Menge der Datenpunkte interpretieren.

Mit

$$u_t(\mathbf{x}) = \frac{\mu_t(\mathbf{x})}{\sum_{h=1}^K \mu_h(\mathbf{x})} \quad (21)$$

läßt sich Gl. (20) auch in der Form

$$y(\mathbf{x}) = \sum_{t=1}^K c_t u_t(\mathbf{x}) \quad (22)$$

schreiben. Damit kann man die aus dem Clusterprozeß hervorgegangenen Funktionen $u_t(\mathbf{x})$ zur Initialisierung der Kernelfunktionen für ein RBF-Netz verwenden und anstelle der bisherigen Werte c_t mit Hilfe der Pseudoinversen neue Werte c_t^* bestimmen.

In ersten Versuchen konnte die Prognosegüte damit jedoch nur geringfügig gesteigert werden. Im Gegensatz dazu führte die Initialisierung der Menge möglicher Kernelfunk-

¹⁰Die Pseudoinverse $\mathbf{A}^+ = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$ löst die Gl. $\mathbf{A} \mathbf{x} = \mathbf{b}$ im Sinne der Minimierung der quadratischen Fehler zu $\mathbf{x} = \mathbf{A}^+ \mathbf{b}$.

tionen mit allen Lerndatenpunkten zu wesentlich besseren Ergebnissen; allerdings unter letztendlicher Verwendung einer viel größeren Anzahl an Kernelfunktionen.

Die Ursache hierfür ist darin zu sehen, daß die Auswahl der Kernelfunktionen beim üblichen RBF-Ansatz mit Blick auf das Gesamtverhalten erfolgt. Es wird dabei also das Zusammenspiel der Kernelfunktionen von Anfang an berücksichtigt. Dies führt zu einer hohen Prognosegüte allerdings auf Kosten der Interpretierbarkeit: Häufig ergeben sich stark überlappende Kernelfunktionen, d.h. für einen angelegten Eingangsvektor sind meistens die Funktionswerte mehrerer Kernelfunktionen von Null verschieden. Zudem sind die sich ergebenden Gewichtungsfaktoren – die positiv und negativ sein können – nicht mehr lokal interpretierbar.

Der hier aufgezeigte Zielkonflikt zwischen Modellierungsgüte und Interpretierbarkeit ist im übrigen generell bei der datenbasierten Fuzzy-Modellierung anzutreffen. Man kann – wie beim Fuzzy-ROSA-Verfahren – auf Regeln abzielen, die jede für sich einen lokal sinnvollen Aspekt der betrachteten Daten darstellen und somit interpretierbar sind. Alternativ kann man auch auf die Interpretierbarkeit verzichten und die Regeln nebst Zugehörigkeitsfunktionen allein danach auswählen bzw. bestimmen, daß das resultierende Modul eine möglichst hohe Prognosegüte liefert.

Zur Abrundung der Einordnung und zum Vergleich sei hier angemerkt, daß sich die für RBF-Netze kennzeichnende Vorschrift (17) auch wie folgt interpretieren läßt: Jedem Tupel $(c_t^*, h_t(\mathbf{x}))$ entspricht eine Regel WENN \mathbf{x} ist $h_t(\mathbf{x})$ DANN $y = c_t^*$, wobei der Erfülltheitsgrad der Prämisse sich direkt zu $\mu_t(\mathbf{x}) = h_t(\mathbf{x})$ ergibt und die Defuzzifizierung durch die Drehmomentmethode (TOR) [Kie97] erfolgt. Diese Sicht macht deutlich, daß die datenbasierte Generierung von interpretierbaren Regeln für ein Fuzzy-Modul, in dem die Drehmomentmethode zur Defuzzifizierung verwendet wird, ein noch nicht befriedigend gelöstes Problem aufwirft: Die Relevanz einer Regel kann nicht durch eine isolierte Betrachtung dieser Regeln, sondern nur durch Berücksichtigung des Zusammenspiels aller Regeln ermittelt werden.

5 Experimente

Operatoren/Parameter		erläuternder Kontext
Aggregation	Produkt	Fuzzy-Modul
Glaubensgrad	Produkt	Fuzzy-Modul
K_{Start}	1	Anzahl Startcluster
α	2	Fuzziness-Parameter
β	0.2	Modifikationsfaktoren nach Gl. (5)
γ	6	Modifikationsfaktoren nach Gl. (6)
u_{min}^α	5	minimale Zuteilung; Löschen von Clustern
ϱ_{min}	0	minimale Relevanz; Löschen von Clustern
ω_{min}	0.9	Ähnlichkeitsschwelle für Vereinigung

Tabelle 1: Gewählte Einstellungen für die Fuzzy-Operatoren und freien Strategieparameter für die nachfolgenden Experimente

5.1 Illustrationsbeispiel *quadratische Form*

Als zu modellierender Prozeß wird zunächst die quadratische Form

$$y = \mathbf{x}^T \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix} \mathbf{x} \quad (23)$$

betrachtet, für die 1000 Datenpunkte durch zufällig gewählte Eingangsraumpositionen erzeugt wurden. Für die Berechnung der Modifikationsfaktoren wird die histogrammbasierte Variante nach Gl. (6) gewählt und abweichend von Tabelle 5 der Parameter u_{min}^α auf den Wert 2 eingestellt. Abbildung 1 zeigt die Verteilung der Datenpunkte und die sich nach 40 Lernschritten ergebende Clusterkonfiguration im Eingangsraum. Für jedes Cluster sind dabei das Zentrum und die Ellipse, die alle Punkte mit gleicher Zugehörigkeit – und zwar $\mu = \frac{1}{e}$ – zum Cluster enthält, eingezeichnet. Offensichtlich sind die Cluster wunschgemäß entlang der Höhenlinien der quadratischen Form ausgerichtet.

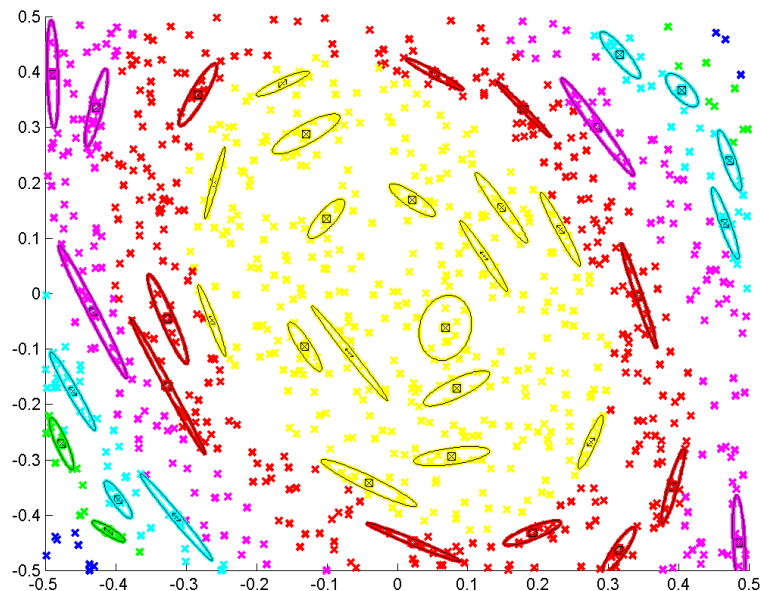


Abbildung 1: Datenpunkte und mit dem SMBC Algorithmus generierte Cluster im Eingangsraum

5.2 Benchmarkprobleme

Der SMBC Algorithmus wird anhand eines Regressions- und zweier Klassifikationsbenchmarks getestet und die Ergebnisse werden mit denen verglichen, die sich mit dem Fuzzy-ROSA-Verfahren bei Anwendung der in [Sla01] definierten Standardvorgehensweise bei 2-facher Kreuz-Validierung ergeben. Da der SMBC Algorithmus nicht deterministisch arbeitet, werden für jede Kreuz-Validierung jeweils 10 Testläufe durchgeführt und die resultierenden Mittelwerte angegeben. Als Prognosegüte ist – je nach Aufgabentyp – jeweils der mittlere absolute Fehler MAE bzw. der mittlere Klassifizierungsfehler MCE auf Testdaten als Ergebnis angegeben. FR_{Sys} bezeichnet die Standardvorgehensweise des Fuzzy-ROSA-Verfahrens, $FR_{Sys+OCR}$ bedeutet, daß der mit

der Standardvorgehensweise generierte Regelsatz anschließend bezüglich der auf den Lerndaten erzielten Güte optimiert wurde. $SMBC_{Cov}$ und $SMBC_{Diag}$ bezeichnen das jeweils für den SMBC Algorithmus verwendete Clustermodell. Beim ersteren wurde die volle, beim letzteren nur die diagonalisierte Kovarianzmatrix verwendet. Die Regelanzahl N_R bzw. die Clusteranzahl K ist angegeben.

Die Modifikationsfaktoren werden für Regressionsaufgaben nach Gl. (5) und für Klassifikationsaufgaben nach Gl. (9) ermittelt und die modifizierte Partitionsmatrix wird re-normalisiert. Die Eingangsgrößen werden mittelwert-bereinigt und auf die jeweilige Standardabweichung normiert¹¹. Insgesamt werden jeweils 40 Lernschritte durchgeführt und als endgültige Clusterkonfiguration diejenige mit der besten Prognosegüte auf den Lerndaten ausgewählt.

Benchmark	FR_{Sys}		$FR_{Sys+OCR}$		$SMBC_{Cov}$		$SMBC_{Diag}$	
	MAE/MCE	N_R	MAE/MCE	N_R	MAE/MCE	K	MAE/MCE	K
Iris ¹²	6.0%	7	6.0%	7	4.8%	4.3	4.7%	4.3
Wine	11.2%	105	10.7%	15	1.8%	3.8	2.4%	3.6
Mackey-Glass	0.07	59	0.05	20	0.026	21.3	0.032	30.4

Tabelle 2: Ergebnisse für die Benchmarks *Iris*, *Wine* und *Mackey-Glass* für den SMBC und das Fuzzy-ROSA-Verfahren im Vergleich

Es zeigt sich, daß mit dem SMBC Algorithmus bessere Ergebnisse erzielbar sind als mit der Standardvorgehensweise beim Fuzzy-ROSA-Verfahren. Allerdings ist anzumerken, daß für das Fuzzy-ROSA-Verfahren weitergehende Ansätze zur automatischen Optimierung der eingangsseitigen Zugehörigkeitsfunktionen und Auswahl von Eingangsgrößen existieren, die teilweise zu besseren Prognosegüten führen [SNK01].

6 Fazit und Ausblick

Der SMBC Algorithmus erweitert unüberwacht arbeitende modell-basierte Clusterverfahren für den Fall des überwachten Regellernens. Cluster werden dabei einzeln für sich nach Relevanzgesichtspunkten bewertet, was auch die Interpretierbarkeit des generierten Regelsatzes erhöht. Durch Mechanismen zum Löschen, Hinzufügen und Vereinigen von Clustern wird automatisch eine passende Clusteranzahl ermittelt.

Zur Zeit findet eine Rückkopplung der Relevanz und der Größe der Cluster in den Clusterprozeß nicht statt. Der Gath-Geva (GG) [GG89] Algorithmus verwendet im Gegensatz zum GK Algorithmus Prototypen, die ellipsoide Cluster unterschiedlicher Größe implizieren. Die hier entwickelten Strategieelemente könnten statt mit dem FCM oder GK Algorithmus auch problemlos mit dem GG Algorithmus zusammen verwendet werden. Ob dies zu einer Verbesserung der erzielten Prognosegüten führt ist zu untersuchen.

In hochdimensionalen Räumen kann die verwendete Abstandsfunktion aufgrund der COD-Problematik¹³ zu unbefriedigenden Ergebnissen führen. In solchen Fällen ist es sinnvoll, entweder vorher global oder während des Clusterprozesses lokal für jedes Cluster eine Eingangsgrößenselektion¹⁴ durchzuführen.

¹¹Hierbei werden auch auf Testdaten die auf den Lerndaten berechneten Werte verwendet.

¹³engl.: curse of dimensionality

¹⁴engl.: feature selection

Eine weitere und nicht näher untersuchte Einsatzmöglichkeit für den SMBC Algorithmus ergibt sich aus der Nutzung der in [Kie98] beschriebenen Querverbindung zwischen Clusterverfahren und Kohonenkarten. Die hier vorgestellten Methoden zur Überwachung des Clusterprozesses durch Rückkopplung der Ausgangsgrößenwerte und zum Hinzufügen, Löschen und Vereinigen von Clustern könnten dazu führen, daß sich die Neuronen bei der – dann in der Größe dynamischen – Kohonenkarte bezüglich der Homogenität der Ausgangsgrößenwerte besser positionieren.

Anmerkung Weitere Erläuterungen und der Pseudo-Code des SMBC Algorithmus sind zu finden in [Hae02].

Danksagung Diese Arbeit wurde von der Deutschen Forschungsgemeinschaft im Rahmen des Graduiertenkollegs *Modellierung und modellbasierte Entwicklung komplexer technischer Systeme* gefördert.

Literatur

- [BR93] BANFIELD, J.D. ; RAFTERY, A.E.: Model-based Gaussian and Non-Gaussian Clustering. In: *Biometrics* 49 (1993), Nr. 3, S. 803–821
- [DLR77] DEMPSTER, A.P. ; LAIRD, N.M. ; RUBIN, D.B.: Maximum likelihood from incomplete data via the EM algorithm. In: *Journal of the Royal Statistical Society B* 39 (1977), S. 1–38
- [FR98] FRALEY, Chris ; RAFTERY, Adrian E.: How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. In: *The Computer Journal* 41 (1998), Nr. 8, S. 578–588
- [GG89] GATH, I. ; GEVA, A.B.: Unsupervised optimal fuzzy clustering. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1989), July, S. 773–781
- [GK79] GUSTAFSON, D. ; KESSEL, W.: Fuzzy clustering with a fuzzy covariance matrix. In: *Proc. IEEE CDC* (1979), S. 761–766
- [Hae02] HAENDEL, L.: *Clusterverfahren zur datenbasierten Regelgenerierung*, Universität Dortmund, Fakultät für Elektrotechnik, Diss., 2002. – in Vorbereitung
- [HKK97] HÖPPNER, F. ; KLAWONN, F. ; KRUSE, R.: *Fuzzy-Clusteranalyse: Verfahren für die Bilderkennung, Klassifizierung und Datenanalyse*. Braunschweig; Wiesbaden : Vieweg, 1997
- [JMF00] JAIN, A.K. ; MURTY, M.N. ; FLYNN, P.J.: Data Clustering: A Review. In: *ACM Computing Surveys* 31 (2000)
- [Kie97] KIENDL, H.: *Fuzzy-Control methodenorientiert*. München : Oldenbourg Verlag, 1997
- [Kie98] KIENDL, H.: Self-organising Adaptive Moment-based Clustering. In: *Proceedings of the Seventh IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '98), Anchorage, USA, 1998* Bd. 2. Piscataway, NJ : IEEE Press, 1998, S. 1470–1475
- [KK94] KRABS, M. ; KIENDL, H.: Automatische Generierung von Fuzzy-Regeln mit dem ROSA-Verfahren. In: *VDI-Berichte* 1113 (1994), S. 29–39
- [Kro99] KRONE, A.: *Datenbasierte Generierung von relevanten Fuzzy-Regeln zur Modellierung von Prozesszusammenhängen und Bedienstrategien*. Düsseldorf : VDI Verlag, 1999 (Fortschritt-Berichte VDI, Reihe 10, Nr. 615)
- [Orr96] ORR, M. *Introduction to Radial Basis Function networks*. 1996
- [Sla01] SLAWINSKI, T.: *Analyse und effiziente Generierung von relevanten Fuzzy-Regeln in hochdimensionalen Suchräumen*, Universität Dortmund, Diss., 2001. – VDI Verlag Düsseldorf, Fortschrittberichte VDI, Reihe 10, Nr. 686
- [SNK01] SCHAUTEN, D. ; NICOLAUS, B. ; KIENDL, H.: Evolutionäres Konzept zur Selektion relevanter Merkmalsätze für die datenbasierte Fuzzy-Modellierung. In: *Proc. 11. Workshop Fuzzy-Control des GMA-FA 5.22* (2001)

Eine kleine Geschichte der Fuzzy-Systeme in der Medizin

Rudolf Seising

Abteilung für Medizinische Experten- und Wissensbasierte Systeme
Institut für Medizinische Computerwissenschaften
Universität Wien, Medizinische Fakultät
Spitalgasse 23, A-1090 Wien, Österreich
Tel. 0043 (1) 40 400 6664
Fax. 0043 (1) 40 400 6667
E-Mail: rudolf.seising@akh-wien.ac.at

1 Vorbemerkung

Bei der nicht mehr überschaubar großen Zahl der oft auch gar nicht als solche erkennbaren Fuzzy-Systeme in unserer wissenschaftlich-technischen Welt kann jede „kleine“ Geschichte dieser Systeme – auch eingeschränkt auf jene „in der Medizin“ – nur einen Ausschnitt präsentieren. Fuzzy-Anwendungssysteme gibt es seit den 70er Jahren in den verschiedenen Wissenschafts-, Wirtschafts- und Industriebereichen. Davon profitierte auch die Medizin, denn Fuzzy-Methoden werden selbstverständlich auch in den technischen Systemen der heutigen „Apparatemedizin“ realisiert.

Diese „Kleine Geschichte der Fuzzy-Systeme“ verfolgt allerdings nicht die Entwicklung der Fuzzy-Technologie in der Medizin. Sie beschränkt sich vielmehr auf den wissenschaftshistorischen Entwicklungspfad, der vom ursprünglichen Konzept der „Fuzzy-Systeme“ zu dessen Anwendung in der medizinischen Diagnostik führt, einem der wichtigsten und grundlegendsten Bereiche der Medizin!

2 Einführung

Von Beginn an hatte Lotfi Zadeh daran gedacht, die Theorie der Fuzzy-Systeme in den Lebenswissenschaften anzuwenden, auch wenn er 1965 in seiner ersten Zeitschriftenpublikation über „Fuzzy Sets“ als mögliche Anwendungsfelder lediglich „pattern recognition, communication of information, and abstraction“ nannte [1]. Vier Jahre später schlug er vor, seine neue Mathematik auch für die Biologie zu nutzen, da die klassische Mathematik dafür möglicherweise nicht ausreicht, „... the possibility that classical mathematics – with its insistence on rigor and precision – may never be able to provide totally satisfying answers to the basic questions relating to the behavior of animate systems“ [2]. Lebende Organismen seien um ein Vielfaches komplexer als unbelebte, von Menschen konstruierte oder beobachtete Systeme, argumentierte er hier und schloß damit allerdings nahtlos an Sätze an, die er schon geschrieben hatte, als seine Theorie der Fuzzy Sets noch gar nicht ersonnen war.

Dass er eine „neue Mathematik“ für notwendig hält, um lebende Systeme zu beschreiben und wissenschaftlich zu untersuchen, läßt sich nämlich schon in einem seiner drei Jahre älteren Texte nachlesen: „we need a radically different kind of mathematics, the

mathematics of *fuzzy*¹ or cloudy quantities which are not describable in terms of probability distributions.“ Diese „neue Mathematik“ sollte sich von ihrer „konventionellen“ Schwesterdisziplin, „the mathematics of precisely-defined points, functions, sets, probability measures, etc.“ geeignet unterscheiden. Wie dies genau aussehen würde, hatte Zadeh damals noch nicht vor Augen gehabt, umso deutlicher aber wozu sie brauchbar wäre: „for coping with the analysis of biological systems, and that to deal effectively with such systems, which are generally orders of magnitude more complex than man-made systems:“[3]

Obwohl Zadeh sein Konzept der Fuzzy Set Theorie erst 1965 veröffentlichte, wurde schon seine drei Jahre frühere Forderung nach einer für die Lebenswissenschaften besser geeigneten „Fuzzy“-Mathematik von einem Lebenswissenschaftler gelesen, zitiert und bestätigt: Der Radiologe Lee Browning Lusted, Professor an der University of Rochester School of Medicine, Rochester, N. Y. und späterer Herausgeber des International Journal of Medical decision making, schloß aus seinen damaligen Untersuchungen zur medizinischen Diagnostik, dass seine Wissenschaft eine neue mathematische Methode zur Sammlung und Codierung medizinischer Informationen brauche. In einem Beitrag für ein mehrbändiges Werk über Computer in der biomedizinischen Forschung ließ er seine Einleitung mit Zadehs Worten von 1962 ausklingen:

„In a recent statement Professor L. A. Zadeh (1962) summed up the situation as follows: »In fact, there is a fairly wide gap between what might be regarded as „animate“ system theorists and „inanimate“ system theorists at the present time, and it is not at all certain that this gap will be narrowed, much less closed, in the near future.

There are some who feel that this gap reflects the fundamental inadequacy of the conventional mathematics – the mathematics of precisely-defined points, functions, sets, probability measures, etc. - for coping with the analysis of biological systems, and that to deal effectively with such systems, which are generally orders of magnitude more complex than man-made systems, we need a radically different kind of mathematics, the mathematics of fuzzy or cloudy quantities which are not describable in terms of probability distributions. Indeed, the need for such mathematics is becoming increasingly apparent even in the realm of inanimate systems, for in most practical cases the *a priori* data as well as the criteria by which the performance of a man-made system is judged are far from being precisely specified or having accurately-known probability distributions.«“ ([5], S. 321)

Auch Lusted war sich der Unzulänglichkeiten „konventioneller“ Mathematik bei der Anwendungen in der Medizin bewusst; er hatte gemeinsam mit dem Professor für Elektrotechnik und Mathematiker für Datenverarbeitungssysteme am National Bureau of Standards, Richard Steven Ledley, der 1960 Gründungspräsident der National Biomedical Research Foundation (NBRF) wurde, eine Vielzahl von Texten publiziert, allen voran den 1959 erschienene Artikel “Reasoning foundations of Medical Diagnoses” [6], um mathematischen Methoden in der Medizin und insbesondere bei der medizinischen Diagnostik ein Anwendungsfeld zu bereiten. Im genannten Artikel analysierten die beiden Autoren für diesen Zweck die Symbolische Logik, die Wahrscheinlichkeitstheorie und die Spieltheorie.

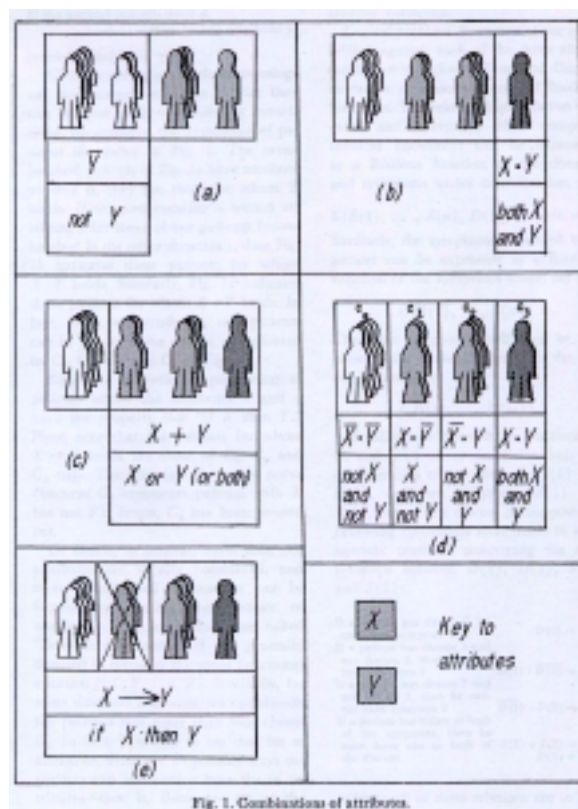
¹ Hervorhebung durch Kursivbuchstaben nicht im Original. (R. S.)

Ledley und Lusted argumentierten, dass der Arzt seine Diagnose mittels logischer Schlussfolgerungen aus Informationen stellt, die er aus zwei Quellen beziehen kann:

- 1) *Medizinisches Wissen*, das sind Informationen über Beziehungen (Relationen), die zwischen Symptomen und Erkrankungen existieren, und
- 2) *Symptome* eines Patienten, die auf weitere für diesen Patienten interessante Informationen schließen lassen.

Die logische Analyse, der Ledley und Lusted den ärztlichen Diagnoseprozeß dann unterzogen, sei hier knapp skizziert (vgl. Figur 1):

- „Attribute“ eines Patienten, etwa das Anzeichen „Fieber“ oder die Krankheit „Lungenentzündung“, stellten sie durch Kleinbuchstaben x, y, \dots dar, während sie die Behauptungen (Aussagen) über das jeweilige Attribut durch entsprechende Großbuchstaben X, Y, \dots kennzeichneten.
- Steht Y für die Aussage „Der Patient hat das Attribut y .“, so ist deren Negation die Aussage $\neg Y$: „Der Patient hat nicht das Attribut y .“
- $X \cdot Y$ repräsentiert die kombinierte Aussage „Der Patient hat die Attribute x und y .“, während $X + Y$ die folgende Kombination bedeutet: „Der Patient hat Attribut x oder Attribut y oder beide.“
- „Wenn der Patient das Attribut x hat, dann hat er Attribut y .“ wird folgendermaßen symbolisiert $X \Rightarrow Y$.



Figur 1: Kombinationen von Attributen. Entnommen aus ([6], S. 10).

Da allgemein mehr als zwei Attribute und auch kompliziertere Ausdrücke zu betrachten sind, führten Ledley und Lusted „Boolesche Funktionen“ $f(X, Y, \dots, Z)$ ein, um entsprechende Kombinationen formulieren zu können. Für mehr als zwei Attribute klassifizierten sie die Patienten in entsprechend viele Mengen C_i : Für m Attribute gibt es 2^m

Möglichkeiten, einem Patienten zuzukommen oder nicht, somit gibt es 2^m Mengen C_i : $C_0, C_1, \dots, C_{2^m-1}$.

Beschränken wir uns hier auf lediglich zwei Attribute, nämlich Symptome (S) und Erkrankungen (D), dann gilt:

- $S(i)$, d. h.: „Der Patient hat Symptom i .“ $i = 1, \dots, n.$
- $D(j)$ d. h.: „Der Patient hat Erkrankung j .“ $j = 1, \dots, m.$

Entsprechende Beispiele aus einem Diagnoselehrbuch lauten:

- Wenn der Patient die Erkrankung 2 hat,
dann hat er das Symptom 1. $D(2) \Rightarrow S(1)$
- Wenn der Patient Erkrankung 1 und nicht
Erkrankung 2 hat, dann hat er Symptom 2. $D(1) \cdot \neg D(2) \Rightarrow S(2)$
- Wenn er Krankheit 1 und nicht Krankheit 2 hat,
dann hat er nicht Symptom 2. $\neg D(1) \cdot D(2) \Rightarrow \neg S(2)$
- Wenn er entweder eines von beiden oder beide
Symptome hat, dann hat er eine von beiden
oder beide Erkrankungen. $S(1) + S(2) \Rightarrow D(1) + D(2)$

Solche Lehrbuchregeln lassen sich streng genommen nur anwenden, wenn ihre Prämissen sicher erfüllt sind. Eine Ärztin oder ein Arzt müsste dazu ganz genau wissen, ob ein Symptom bzw. eine Erkrankung bei einem Patienten bzw. einer Patientin vorliegt, oder nicht. In vielen Fällen ist diese Gewissheit aber auch nach Anamnese, ärztlicher Untersuchung und geeigneten Labortests nicht gegeben und dann werden Lehrbuchregeln wie die obigen für die praktische Umsetzung etwa folgendermaßen modifiziert: „If a patient has disease 2, then there is only a certain chance that he will have symptom 2 – that is, say, approximately 75 out of 100 patients will have symptom 2.“ ([6], S. 13) So jedenfalls argumentierten Ledley und Lusted im Abschnitt, der auf die oben skizzierte logische Analyse folgte; sie zogen dann auch sogleich die Konsequenz: „Since »chance« or »probabilities« enter into »medical knowledge«, then chance, or probabilities, enter into the diagnosis itself.“ ([6], S. 13) Daß in $\frac{3}{4}$ der auftretenden Fälle ein bestimmtes Symptom eine bestimmte Erkrankung beweist, läßt die Diagnose nicht sehr exakt erscheinen, und für einen Patienten, dessen Gesundheit betroffen ist, ist sie wenig hilfreich. Möglicherweise betonte Lusted aus diesem Grunde sechs Jahre später die „inadequacy of conventional mathematical methods for dealing with biological problems“ ([5], S. 321).

3 Fuzzy-Systeme

Seit 1958 war Lotfi Zadeh Professor für Elektrotechnik an der University of California, Berkeley. Mehr als zehn Jahre hatte er zuvor an der New Yorker Columbia University auf den Gebieten der Schaltkreis-, Regelungs-, Antennen-, Filter-, Kommunikations- und Netzwerktheorie gearbeitet. Er war maßgeblich an der Gründung des neuen Forschungsfeldes „Systemtheorie“ beteiligt, in dem ein abstraktes System eine „black box“ ist, mit input (u), output (y) und einem klar definierten Zustand (x) zu jedem Zeitpunkt t . Ein Gleichungssystem setzt diese drei Größen zueinander in Beziehung:

$$x_{t+1} = f(x_t, u_t), \quad y_t = g(x_t, u_t) \quad t = 0, 1, 2, \dots$$

„A New View of System Theory“² hieß dann ein Vortrag, den Zadeh beim „Symposium on System Theory“ im *Polytechnic Institute of Brooklyn* im April 1965 hielt, als sein berühmter Artikel „Fuzzy Sets“ schon angenommen, aber noch nicht erschienen war. In diesem Vortrag und auch im Text der Proceedings, der dann den Titel „Fuzzy Sets and Systems“ erhielt, bot Zadeh in der Tat eine völlig neue Sichtweise der Systemtheorie, denn er definierte „Fuzzy Systeme“: „ S is a fuzzy system, if $u(t)$ or $y(t)$ or $x(t)$ or any combination of them ranges over fuzzy sets.“ ([7] S. 33)



Fig. 1.2.1 Diagrammatic representation of a system G with input u and output y .

Figur 2: System mit Input und Output. Entnommen aus ([4], S. 9).

Ein zur Zeit t mit „erheblich mehr als 5“ angegebener input sei ein Fuzzy Set und ein System, das auf solchen ungenau definierten inputs agieren kann, sei ein Fuzzy System. Auch wenn Zustände von S durch fuzzy Adjektive wie „leicht“, „schwer“, nicht sehr schwer, sehr leicht etc. beschrieben würden, wenn es sich dabei also um Fuzzy Sets handele, dann sei S ein Fuzzy System. Ein System S mit diskreter Zeit könne dann durch die gewöhnlichen Zustandsgleichungen charakterisiert werden.

$$x_{t+1} = f(x_t, u_t) \quad y_t = g(x_t, u_t)$$

„The difference between fuzzy and non-fuzzy systems, then, lies in the nature of the ranges of the variables u_t , y_t and x_t .“ ([7], S. 33)

Um Fuzzy Systeme von den formal ähnlich gebildeten stochastischen Systemen abzugrenzen, schrieb Zadeh, wohl um auf die zu erwartende Kritik seitens der Wahrscheinlichkeitstheoretiker und Statistiker schon vorweg einzugehen: „The difference between stochastic and fuzzy systems is that in the latter the source of imprecision is nonstatistical in nature and has to do with the lack of sharp boundaries of the classes entering into the descriptions of the input, output or state.“ ([7], S. 33 f)

Als Systeme, die sowohl hinsichtlich ihrer Beschreibung wie auch ihres zeitlichen Verhaltens einen unvermeidlichen Grad an Fuzziness aufweisen, sah Zadeh etwas später, wie schon oben erwähnt, die biologischen Systeme an. Fuzziness sei der für die Unbrauchbarkeit exakter Mathematik zu zahlende Preis, wenn Systeme zu analysieren sind, die aus einer großen Zahl miteinander wechselwirkender Elemente bestehen, oder bei denen sehr viele Variablen berücksichtigt werden müssen. Seine Theorie der „so-called fuzzy sets and systems“ sei allerdings geeignet, mit solcher quantitativen oder quasi-quantitativen Fuzziness umzugehen. ([2], S. 200)

Die Beschreibung biologischer Systeme war für Zadeh aber schon 1969 gar nicht das vielversprechendste mögliche Anwendungsfeld für seine neue mathematischen Theorie, denn als dieses sah er klar und deutlich den Bereich der medizinischen Diagnosen an ([2], S. 200). Hier lägen die vielleicht natürlichsten Anwendungen seiner Fuzzy-Konzepte, und seine damaligen Ausführungen wiesen in der Tat in die Richtung späterer Entwicklungen:

² Hervorhebung durch Kursivbuchstaben nicht im Original. (R. S.)

- „Specifically, from the point of view of fuzzy set theory, a human disease, e.g., diabetes, may be regarded as a fuzzy set in the following sense. Let $X = \{x\}$ denote the collection of human beings. Then diabetes is a fuzzy set, say D , in X , characterized by a membership function $\mu_D(x)$ which associates with each human being x his grade of membership in the fuzzy set of diabetes.” ([2], S. 205)
- “In some cases, it may be more convenient to characterize a fuzzy set representing a disease not by its membership function but by its relation to various symptoms which in themselves are fuzzy in nature. For example, in the case of diabetes a fuzzy symptom may be, say, a hardening of the arteries. If this fuzzy set in X is denoted by A , then we can speak of the fuzzy inclusion relation between D and A and assign a number in the interval $[0,1]$ to represent the “degree of containment” of A in D . In this way, we can provide a partial characterization of D by specifying the degrees of containment of various fuzzy symptoms A_1, \dots, A_k in D . When arranged in a tabular form, the degrees of containment constitute what might be called a *containment table*.” ([2], S. 205)

4 Fuzzy Sets und Fuzzy Systeme in der Medizin

Offenbar in Unkenntnis der auf den vorhergehenden Seiten angesprochenen Vorschläge Zadehs zu Anwendungen der Fuzzy Set Theorie in der Medizin, schrieb Merle Anne Albin zu Anfang der 70er Jahre an einer Dissertation über dieses Thema bei dem mit Zadeh befreundeten Mathematikprofessor Hans Bremermann an der University of California in Berkeley. Der Promotionsausschuß, dem auch Zadeh angehörte, nahm Albins Arbeit *Fuzzy Sets and Their Applications to Medical Diagnosis and Pattern Recognition* am 14. Juni 1975 an; sie wies keinen der oben zitierten Texte von Zadeh im Literaturverzeichnis auf, und beginnt mit der Feststellung, dass sowohl die mathematische Theorie der medizinischen Diagnose als auch die Theorie der Fuzzy Sets sehr neue Gebiete seien, die Ihre meisten Anstöße im vorausgegangenen Jahrzehnt zu verzeichnen hatten, und mit der Behauptung: „It is not too surprising, then, that the application of fuzzy sets to medical diagnosis has never before been seriously attempted.“ ([8], S. 1)

Auf die oben genannten Arbeiten Zadehs mit Bemerkungen zu möglichen Anwendungen von Fuzzy Sets in der Medizin gingen dann auch weder Harry Wechsler in seinem 1976 erschienenen Artikel *Applications of Fuzzy Logic to Medical Diagnosis* [9] noch Alonso Perez-Ojeda in seiner 1976 in Toronto erschienenen M.S. Thesis *Medical Knowledge Network. A Database for Computer Aided Diagnosis* ein [12], nicht P. Tautu und G. Wagner in ihrem im Januar 1978 erschienenen Übersichtsartikel *The Process of Medical diagnosis: Routes of Mathematical Investigations* [10], und nicht R. C. Elder und A. O. Esogbue in den 1979 und 1980 erschienenen beiden Teile ihrer auf Elders M.S. Thesis *Fuzzy Systems Theory and Medical Decision Making* [11] zurückgehende Arbeit über ein Fuzzy-Modell für ärztliche Entscheidungsprozesse [13, 14].

Alle diese Arbeiten behandelten die Möglichkeiten der Fuzzy Set Theorie in der Medizin, ohne an die expliziten Bemerkungen Zadehs zu diesem Thema anzuknüpfen; stattdessen wurde das Rad unabhängig voneinander neu erfunden:

- Medizinische Laborbefunde, die entweder als *normal* oder als *pathologisch* klassifiziert wurden ließen sich fuzzifizieren; ein kontinuierlicher Übergang etwa von *niedrigen* zu *normalen* zu *hohen* Blutungszeiten oder von *leicht erhöhtem* zum *stark erhöhten* abnormen Cholesterinspiegel entspricht mehr der medizinischen Denkweise. Die Zugehörigkeitsfunktionen legen Werte fest, mit denen die Testergebnisse zum jeweiligen Fuzzy Set gehören [8, 10, 13, 14, 15]

Ein weiterreichender Vorschlag, die Beziehungen zwischen Symptomen und Krankheiten mathematisch zu fassen, kam von Elie Sanchez, der 1974 an der Medizinischen Fakultät in Marseille eine humanbiologische Arbeit über „Equations de Relations Floues“ eingereicht [18] und später u. a. in der damals von Zadeh mit herausgegebenen Zeitschrift *Information and Control*, in der 1965 auch die „Fuzzy Sets“ publiziert wurden, eine Theorie der „Fuzzy-Relationen“ schuf.

4.2 Relationen, Fuzzy Relationen und deren Komposition

Eine „Fuzzy-Relation“ ist die „Fuzzifizierung“ des herkömmlichen mengentheoretischen Relationenbegriffs. Werden mit X and Y herkömmliche Mengen bzw. mit $X \times Y$ deren kartesisches Produkt bezeichnet, so sind:

- $L(X)$ die Menge aller Fuzzy Sets in X ,
- $L(Y)$ die Menge aller Fuzzy Sets in Y und
- $L(X \times Y)$ die Menge aller Fuzzy Sets in $X \times Y$.

Relationen zwischen X und Y sind Teilmengen ihres kartesischen Produkts $X \times Y$, während Fuzzy-Relationen zweier Mengen X und Y von Zadeh 1973 als Teilmengen in $L(X \times Y)$, der Menge aller Fuzzy Sets in $X \times Y$ definiert wurden [16].

Es seien nun X , Y und Z drei herkömmliche Mengen mit der Fuzzy-Relation Q zwischen X und Y und der Fuzzy-Relation R zwischen Y und Z : $Q \in L(X \times Y)$ und $R \in L(Y \times Z)$.

Ließen sich diese beiden Fuzzy-Relationen zu einer neuen Fuzzy-Relation $T \in L(X \times Z)$ zwischen X und Z kombinieren?

In der herkömmlichen Mengentheorie erhält man durch Komposition der beiden Relationen $q \in X \times Y$ und $r \in Y \times Z$ die Relation $t \in X \times Z$, bezeichnet durch $t = q * r$, aufgrund folgender Definition:

$$t = q * r = \{(x, z) \mid \exists y : (x, y) \in q \wedge (y, z) \in r\}$$

Fuzzy-Relationen $Q \in L(X \times Y)$ und $R \in L(Y \times Z)$ lassen sich ebenfalls zu einer neuen Fuzzy-Relation $T \in L(X \times Z)$ kombinieren, wenn die logischen Verknüpfungen durch die entsprechenden Fuzzy-Operatoren der Zugehörigkeitsfunktionen ersetzt werden.

- In der obigen Definition der Komposition herkömmlicher Relationen findet sich ein logisches UND (\wedge), das zur „Fuzzifizierung“ durch den auf die entsprechenden Zugehörigkeitsfunktionen angewendeten Minimum-Operator ersetzt wird.³
- In der obigen Definition der Komposition herkömmlicher Relationen findet sich der Ausdruck „ $\exists y$ “, („es existiert ein y “). Das existierende $y \in Y$ ist das erste, oder das zweite, oder das dritte ... (usw.), logisch geschrieben: $\bigvee_{y \in Y}$. Diese logische ODER-Verknüpfung wird bei der „Fuzzifizierung“ durch den auf die entsprechenden Zugehörigkeitsfunktionen angewendeten Maximum-Operator ersetzt.⁴

³ Selbstverständlich können auch die anderen vorgeschlagenen Fuzzy-Operatoren benutzt werden; man erhält dann entsprechend andere Fuzzy-Relationen.

⁴ Auch für das „Fuzzy-Oder“ gibt es neben dem Max-Operator weitere Verknüpfungsoperatoren, die dann zu anderen Fuzzy-Relationen führen.

Die Fuzzy-Relation $T = Q * R$ wird somit über folgende, von Zadeh als „Max-min-Komposition“ eingeführte Kombinationsregel⁵ für die Zugehörigkeitsfunktionen definiert [16]:

$$\mu_T(x, y) = \max_{y \in Y} \min\{\mu_Q(x, y); \mu_R(y, z)\}, y \in Y$$

Sanchez hatte die Theorie der Fuzzy-Relationen in den 70er Jahren ausgebaut und interessante Ergebnisse bei deren Anwendung „in transportation problems and in belief systems“ erwartet. ([19], S. 47) Er selbst schlug eine andere Richtung ein: „We plan to investigate *medical*⁶ aspects of fuzzy relations at some future time.“ ([19], S. 47)

Zwei zusammengehörige Artikel, die er für einen 1979 von Gupta, Ragade und Yager herausgegebenen Band über Fortschritte in der Fuzzy Set Theorie und ihre Anwendungen publizierte [21, 22], geben über seine Untersuchungen Aufschluß: Im ersten Text, „Compositions of Fuzzy Relations“ wurden die abstrakt-mathematischen Grundlagen für die Ausführungen im zweiten Text „Medical Diagnosis and Composite Fuzzy Relations“ gelegt. Darin wird betont, dass medizinische Diagnosen oftmals gestellt werden müssen, ohne dass eine präzise Analyse möglich ist. Von den meist nur ungenau beschreibbaren Symptome eines Patienten muß dann auf eine oder mehrere Erkrankungen geschlossen werden. Dabei können weder die Menge der in Frage kommenden Krankheiten noch der Schluß von den Symptomen auf die Erkrankung(en) präzise sein.

Nun wiederum wohl ohne die Arbeit von Perez-Ojeda zu kennen, führte Sanchez, der inzwischen der Abteilung für Biomathematik und Informatik der Medizinischen Fakultät in Marseille angehörte, die Beziehung zwischen einer Menge von Symptomen und einer Menge von Krankheiten als Fuzzy-Relation ein, die er „Medizinisches Wissen“ nannte: „In a given pathology, we denote by S a set of symptoms, D a set of diagnosis and P a set of patients. What we call „medical knowledge“ is a fuzzy relation, generally denoted by R , from S to D expressing associations between symptoms, or syndroms, and diagnosis, or groups of diagnosis.“ ([22], S. 438.)

In Zadehs „Max-Min-Komposition“ fand er einen „Inferenzmechanismus“, der von ungenauen Symptombeschreibungen (Fuzzy-Sets aus S) auf ungenaue Beschreibungen der Erkrankung des Patienten (Fuzzy Sets aus D) zu folgern erlaubt. Mit dieser Inferenzregel lassen sich medizinische Diagnosen D_j über die Krankheit eines Patienten aus Symptomen S_i mit Hilfe des medizinischen Wissens R fuzzy-logisch ableiten. Die Zugehörigkeitsfunktion wird dabei folgendermaßen berechnet:

$$\mu_T(p, d) = \max_{s \in S} \min\{\mu_Q(p, s); \mu_R(s, d)\}, \quad \text{wobei } s \in S, d \in D.$$

Bei Berücksichtigung einer Menge P aller betrachteten Patienten und einer Fuzzy-Relation Q zwischen P und der Symptomemenge S war nun mit Hilfe der Max-min-Kompositionsregel eine Fuzzy-Relation $T=Q*R$ mit der Zugehörigkeitsfunktion $\mu_T(p,d)$ erhältlich:

$$\mu_T(p, d) = \max_{s \in S} \min\{\mu_Q(p, s); \mu_R(s, d)\}, \quad s \in S, d \in D, p \in P.$$

Mit $\mu_R(s,d)$ ist die Zugehörigkeitsfunktion der Fuzzy-Relation R des „Medizinischen Wissens“ bezeichnet. Diese Fuzzy-Relation R lässt sich als Matrix schreiben, deren Einträge nach Befragung von Ärzten nach deren diagnostischen Wissen und ihrer Erfahrung vorgenommen werden kann. Dieses medizinische Expertenwissen muß dazu in *Grade der Verbindung* zwischen Symptomen und Diagnosen übersetzt werden.

⁵ Die Max-min-Kompositionsregel wird bei unendlichen Mengen durch die Sup-min-Kompositonsregel ersetzt. Hier reicht es aber anzunehmen, dass alle Mengen endlich sind.

⁶ Hervorhebung durch Kursivbuchstaben nicht im Original. (R. S.)

1980 berichteten Joly, Sanchez, Gouvernet und Valty über ihre Versuche mit Testläufen für 21 Patienten, die Max-min-Komposition zur Diagnose von Herzerkrankungen anzuwenden. Die von Kardiologen mit entsprechenden Verbindungsgraden ausgefüllte Matrix der Fuzzy-Relation R setzte fünf Symptome zu drei Diagnosen ins Verhältnis: *normal*, *linksventrikuläre Hypertrophie* und *valvuläre Kardiopathie*.

5. Computerunterstützte medizinischen Diagnostik in Wien

5.1 CADIAG (Computer-Assisted DIAGnosis)

1968 hatten der Medizinprofessor G. Grabner vom Institut für Medizinische Computerwissenschaften der Universität Wien und dem dortigen Allgemeinen Krankenhaus und W. Spindelberger von IBM Österreich erstmals über das von ihnen entwickelte, durch die Arbeiten von Ledley und Lusted motivierte Computerverfahren zur diagnostischen Hilfestellung publiziert [23], das bald darauf in der Differentialdiagnostik von Lebererkrankungen erstmals angewendet wurde [24].

Dieses Konsultationssystem sollte aktiver Partner des Arztes im diagnostischen Prozeß sein, medizinisches Expertenwissen aus der internen Medizin bereitstellen und aufgrund von ihm mitgeteilten Patientensymptommustern medizinische Diagnosen logisch folgern und begründen können. Die Kliniker erhofften sich auf diesem Wege eine Steigerung des Qualitätsniveaus ihrer Diagnostik, schnellere Diagnoseerstellung und die Vermeidung von Fehldiagnosen.

In diesem System konnten vielerlei möglichen Beziehungen zwischen Symptomen, Diagnosen, Krankheiten und Patienten durch Formeln des Kalküls der Prädikatenlogik 1. Ordnung dargestellt werden. Bewußt und betont wurde darauf verzichtet, Wahrscheinlichkeitstheorie und Statistik zu benützen, denn: „Das Ziel des Programms ist nicht die Erstellung einer einzigen wahrscheinlichen Diagnose, sondern es sollen alle jene Krankheiten erfaßt werden, die bei einer bestimmten Symptomkonstellation überhaupt möglich sind.“ ([24], S. 585)

Die Grenzen der zweiwertigen Logik gaben allerdings auch die Grenzen der Darstellung des „Medizinischen Wissens“ zur Verarbeitung im Computer vor. Da Aussagen in der klassischen Aussagenlogik nur entweder „wahr“ oder „falsch“ sein können, waren Symptome nur entweder „vorhanden“ oder „nicht vorhanden“, konnten Diagnosen nur entweder „zutreffen“ oder „nicht zutreffen“, und da Ärzte und Kliniken gar nicht immer alle möglichen Daten zu erheben imstande sind, wurde in vielen Fällen der Terminus „nicht untersucht“ notiert, und auch eine Diagnose musste oft „möglich“ bleiben, weil sie weder sicher gestellt noch sicher ausgeschlossen werden konnte. Somit war es plausibel, die Verwendung einer dreiwertigen Logik bei der computerunterstützten medizinischen Diagnose zu erwägen, und schon im zweiten Jahr wurde im Wiener System das bei dieser Gelegenheit ein neues Design und den Namen CADIAG erhielt, die ihm unterliegende zweiwertige Logik-Kalkül, durch einen dreiwertigen Logik-Kalkül ersetzt: Neben den beiden Werten „1“ und „0“ für Symptome bzw. Diagnosen gab es nun auch die „2“ für „möglich“ oder „nicht untersucht“.

Die zwischen Symptomen und Krankheiten möglichen Beziehungen wurden in sieben logischer Relationen unterteilt:

(1) OB obligat und beweisend

Besteht zwischen einem Symptom S und einer Krankheit K die Relation OB , so muß das Symptom S bei dieser Krankheit auftreten und es beweist diese Krankheit K . Umgekehrt kann bei nicht vorhandenem Symptom S die Krankheit K ausgeschlossen werden.

Aussagenlogischer Operator: *Äquivalenz* $S \Leftrightarrow K$.

D	0	1	2
S			
0	1	0	2
1	0	1	2
2	2	2	2

(2) FB fakultativ und beweisend

Besteht zwischen einem Symptom S und einer Krankheit K die Relation FB , so muß das Symptom S bei dieser Krankheit nicht auftreten, tritt es aber auf, dann beweist es die Krankheit K .

Aussagenlogischer Operator: *Implikation* $S \Rightarrow K$.

D	0	1	2
S			
0	1	1	1
1	0	1	2
2	2	1	2

(3) ON obligat und nicht beweisend

Das Symptom S muß bei Krankheit K auftreten, es beweist sie aber nicht. Umgekehrt wird K bei nicht vorhandenem Symptom S ausgeschlossen.

Aussagenlogische Interpretation: $S \Leftarrow K$ oder $\neg S \Rightarrow \neg K$.

D	0	1	2
S			
0	1	0	2
1	1	1	1
2	1	2	2

(4) FN fakultativ und nicht beweisend

Das Symptom S muß bei Krankheit K nicht auftreten; wenn es auftritt, beweist es die Krankheit K nicht. Die Stärke dieser Beziehung kann daher nicht sicher definiert werden; diese Relation wird daher bei Ungewißheiten relativ häufig gewählt.

Aussagenlogische Interpretation: $(S \Rightarrow K) \vee (S \Rightarrow \neg K)$.

D	0	1	2
S			
0	1	1	1
1	1	1	1
2	1	1	1

(5) A ausschließend

Ist das Symptom S vorhanden, so ist die Krankheit K ausgeschlossen.

Aussagenlogische Interpretation: $S \Rightarrow \neg K$ oder $\neg (S \wedge K)$.

D	0	1	2
S			
0	1	1	1
1	1	0	2
2	1	2	2

(6) H Hinweis

Das Symptom S weist auf eine Zweitdiagnose hin.

Aussagenlogische Interpretation: $S \wedge K_1 \Rightarrow K_1 \vee K_2 \vee K_3 \vee \dots \vee K_N$.

D	0	1	2
S			
0	1	0	2
1	0	1	2
2	2	2	2

(7) ' unbekannt oder unspezifisch

Es gibt entweder keinen bekannten oder nur einen unspezifischen Zusammenhang zwischen Symptom S und Krankheit K .

5.2 CADIAG-II

Etwa ein Jahrzehnt später kam es erneut zu einer gravierenden Systemveränderung, die eine neue Versionsnummer rechtfertigte.⁷ CADIAG-II wurde durch Erweiterungsarbeiten von Klaus-Peter Adlassnig geprägt, der im September 1976 an das Wiener Institut für Medizinische Computerwissenschaften kam und folgende Situation konstatierte: in der medizinischen Diagnostik „gibt es meist keine scharfen Grenzen zwischen den einzelnen Krankheiten, das Auftreten mehrerer Krankheiten bei einem Patienten verwischt das Symptombild und erschwert die diagnostische und therapeutische Entscheidung, die Einordnung erhobener Befunde in normal oder pathologisch ist in Grenzfällen oft willkürlich, die Beschreibung der Intensität eines Schmerzes kann nur verbal erfolgen und ist somit von der subjektiven Einschätzung des Patienten abhängig und in Krankheitsbeschreibungen können äußerst selten genaue Zuordnungen zwischen Symptomen und Krankheiten getroffen werden.“ ([26], S. 12)

Adlassnig sah einen grundsätzlichen Unterschied zwischen dem hohen Präzisionsgrad in den exakten Natur- und Ingenieurwissenschaften einerseits und der „Ungenauigkeit andererseits, die bei Beschreibungen aus der Soziologie, Psychologie, Medizin, Linguistik, Literatur, Kunst, Philosophie u. a. zu finden ist.“ ([26], S. 12) In der Theorie der Fuzzy Sets, deren Anwendungspotential für die computerunterstützte Diagnose er dem Beitrag von Moon, Jordanov, Perez und Turksen auf der MEDINFO '77 in Toronto kennenlernte [15], sah Adlassnig „ein Konzept, das die große Komplexität und Ungenauigkeit der Definitionen in diesen Gebieten berücksichtigt“ ([26], S. 12): “Fuzzy set theory with its capability of defining inexact medical entities as fuzzy sets, with its linguistic approach providing an excellent approximation to medical texts as well as its power of approximate reasoning, seems to be perfectly appropriate for designing and developing computer assisted diagnostic, prognostic and treatment recommendation systems.” ([27], S. 205) Somit schrieb Adlassnig durch Integration der Fuzzy-Logik in das CADIAG-System gewissermaßen den Erweiterungsprozeß von der harten Logik zur „soft“ Logik in der computerunterstützten medizinischen Diagnose fort [25].

Die Wissensbasis von CADIAG-II enthält neben definitorischem, statistischem und erfahrungsgeleitetem Wissen wiederum auch medizinisch-diagnostisches Wissen, das nun aber als fuzzy-logische Relationen repräsentiert wurde,

- zwischen Symptomen und Krankheiten (S_i, D_j),
- zwischen Symptomen (S_i, S_j),
- zwischen Krankheiten (D_i, D_j), und
- zwischen Symptomkombinationen und Krankheiten (SC_i, D_j).

Um diese Fuzzy-Relationen für Diagnosen nutzen zu können, sind Bewertungen über das *Auftreten* des jeweiligen Symptoms bei und seine *Beweiskraft* für die entsprechende Krankheit von besonderem Interesse. Der erste Aspekt wird durch die *Häufigkeit* gemessen, mit der das Symptom bei Patienten bisher aufgetreten ist, die unter der Krankheit litten, der zweite Aspekt kennzeichnet die „*logische Stärke*“, mit der aus dem Auftreten des Symptoms auf die Krankheit geschlossen werden kann.

Es wurden Ärzte als die medizinischen Experten befragt, um Fuzzy-Bewertungen dieser beiden Aspekte jeweiliger Fuzzy-Relationen zu erhalten. Die Antworten der Mediziner wurden in ein Schema vorab definierter linguistischer Begriffe eingeordnet, denen Fuzzy-Sets zugeordnet sind. Auf die Fragen

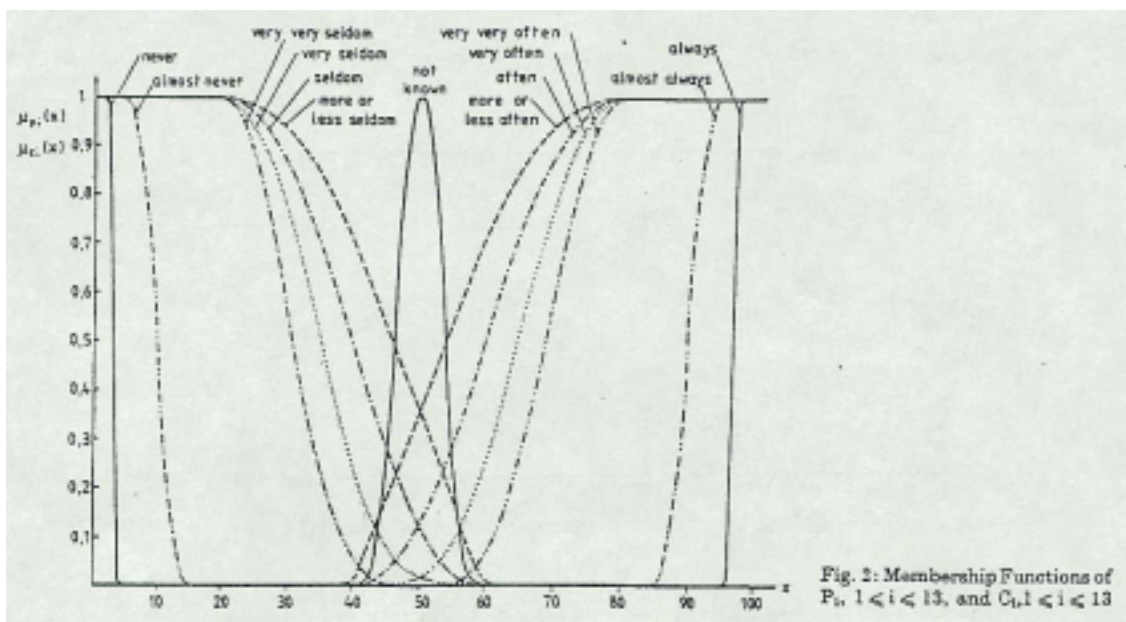
⁷ Das erste CADIAG-System hatte ursprünglich allerdings gar keine Versionsnummer.

- Wie oft tritt S_i bei D_j auf? (Auftreten A) und
- Wie streng beweist S_i D_j ? (Beweiskraft B)

waren folgende Antwortmöglichkeiten vorgesehen: *immer, fast immer, sehr sehr oft, sehr oft, oft, mehr oder weniger oft, unspezifisch, selten, mehr oder weniger selten, sehr selten, fast nie, nie, unbekannt.*

Damit waren die logischen Relationen aus CADIAG-I zu Randelementen des Fuzzy-Relationenspektrums von CADIAG-II geworden, denn das „immer“-Auftreten eines Symptoms S_i bei der Krankheit D_j entspricht der früheren Beziehung „obligat“ und die Beweiskraft „immer“ entspricht der früheren Beziehung „beweisend“; das „nie“-Auftreten und die Beweiskraft „nie“ entsprechen dagegen dem früheren „ausschließend“.

Als Zugehörigkeitsfunktionen benutzte Adlassnig in CADIAG-II die schon von Zadeh in [29] angegebenen Standardfunktionen, die für die oben genannten 13 Fuzzy-Sets folgendes Aussehen haben (vgl. Figur 2):



Figur 2: Zugehörigkeitsfunktionen der als Antwortmöglichkeiten auf die Fragen nach Auftreten (*presence*, p) bzw. Beweiskraft (*conclusiveness*, c) definierten Fuzzy Sets; entnommen aus ([28], S. 145)

Durch die Einführung einer weiteren grundlegenden Fuzzy-Relation zwischen der Menge P aller Patienten und der Symptomemenge S und mit der Möglichkeit, unter Anwendung der Max-Min-Kompositionsregel die Fuzzy-Relationen zwischen S und D , D und P , P und S und/oder deren Komplemente zu verknüpfen, entsteht für CADIAG-II ein großes Repertoire, Diagnosen zu ermitteln. Dabei schlägt das System vor, Diagnosen als bewiesen bzw. ausgeschlossen zu indizieren, oder weiterhin im Hypothesenstadium zu belassen, um erst später zu entscheiden, wie sie zu bewerten sind.

CADIAG-II wird im Wiener Allgemeinen Krankenhaus als „elektronisches Nachschlagewerk für mögliche Diagnosen“ und als „komplexes on-line Konsultationssystem für spezielle Fälle zur detaillierten und vollständigen differentialdiagnostischen Abklärung der Erkrankung des Patienten“ eingesetzt. ([30], S. 374.) Die Wissensbasis wurde 1993 durch „Krankheitsprofile und Symptomkombinationen für insgesamt 262 Krankheiten, darunter 185 rheumatologische Erkrankungen (69 Gelenkerkrankungen, 12

Erkrankungen der Wirbelsäule, 38 Krankheiten des Binde- und Stützgewebes, 45 Knochenkrankungen und 21 regionale Schmerzsyndrome) und 77 Krankheiten aus dem Bereich der Gastroenterologie (24 Gallenblasen- und Gallenwegserkrankungen, 10 Pankreaserkrankungen, 37 Dickdarmerkrankungen) und der Hepatologie (6 Hepatitiden)“ ([31], S. 54.) charakterisiert. In über 600 klinischen Fällen wurden seine Vorschläge mit den klinischen Diagnosen verglichen und seine Genauigkeit lag dabei zwischen 80 und 95 %. Die Fehldiagnosen sind meist darauf zurückzuführen, dass bereits begonnene Therapien die Symptome schon bekämpften, die Krankheit noch anfänglich bzw. schon stabil war, oder die Ergebnisse der Anamnese kaum aussagekräftig schienen. CADIAG-II befindet sich weiterhin im Experimentier- und Ausbaustadium.

6. Nachbemerkung

Lebendige Systeme, Organismen sind komplex und variabel und daher mit Theorien, Modellen und Methoden der exakten Wissenschaften nicht fassbar. Um ihr Verhalten dennoch beschreiben oder sogar beeinflussen zu können, sind Biologie und Medizin auf ein wissenschaftliches Instrumentarium angewiesen, mit dem Unschärfen, Unsicherheiten und Unvollständigkeiten bei der Systembeschreibung und deren Verhaltensvorhersage toleriert werden können.

Andererseits kann die Medizin nicht auf eine Erkenntniswissenschaft reduziert werden. Nachdem der Mediziner eine Krankheit aufgrund von Symptomen und Labortestergebnissen bei einem Patienten diagnostiziert hat, muß er als Arzt handeln. In der klinischen Wissenschaft geht es auch um die Therapie des Patienten und um seine Heilung. Entscheidungen müssen dazu oft unter großem Zeitdruck getroffen werden. Daß dabei Fehler gemacht werden können, ist nur allzu menschlich. Therapeutische Fehlentscheidungen sind auf Fehldiagnosen zurückzuführen und computerunterstützte Diagnosesysteme können diese Gefahr zu vermeiden helfen, indem sie eine Hauptdiagnose vorschlagen, an Zweit- und Nebendiagnosen erinnern, schwierige Differentialdiagnosen unterstützen und seltene Diagnosen nicht von vornherein ausschließen.

Diese „Kleine Geschichte der Fuzzy Systeme in der Medizin“ schilderte Aspekte der Entwicklung eines Anwendungspotentials von Fuzzy Sets, Fuzzy-Logik und Fuzzy-Relationen in der Medizin, die einen anderen Weg genommen hat, als die sehr viel bekanntere Fuzzy-Technologie, die möglicherweise nicht weniger erfolgreich sein wird.

Dank

Ich danke Lotfi Zadeh und Klaus-Peter Adlassnig für ihre Hilfen und ihre Bereitschaft, mir ihre mündlichen Erinnerungen sowie ihre (leider nicht mehr vollständig vorhandenen) schriftlichen Unterlagen über die Geschichte der Fuzzy Set Theorie und der Fuzzy-Systeme bzw. über die Entstehungsgeschichte von CADIAG- und CADIAG-II zur wissenschaftshistorischen Verarbeitung zu überlassen.

Literatur

- [1] Zadeh, L. A.: Fuzzy Sets. *Information and Control* 8, 1965, S. 338-353.
- [2] Zadeh, L. A.: Biological Applications of the Theory of Fuzzy Sets and Systems. *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*. Little, Brown and Company: Boston 1969, S. 199-206.
- [3] Zadeh, L. A.: From Circuit Theory to System Theory, *Proc. IRE*, Vol. 50, S. 856 ff.
- [4] Zadeh, L. A.: The Concepts of System, Aggregate, and State in System Theory. In: Zadeh, L. A., Polak, E. (Editors): *System Theory*. New York: McGraw-Hill 1969, S. 3-42.
- [5] Lusted, L. B.: Computer Techniques in Medical Diagnosis. In: Stacy, R. W., Waxmann, B. D. (Editors): *Computers in Biomedical Research* Volume I, New York and London: Academic Press 1965, S. 319-338.
- [6] Ledley, R. S. and Lusted, L. B.: Reasoning Foundations of Medical Diagnosis. *Science*, 3. July 1959, Volume 130, Nr. 3366, S. 9-21.
- [7] Zadeh, L. A.: Fuzzy Sets and Systems. *Proc. Symposium on System Theory, Polytechnic Institute of Brooklyn*, April 1965, S. 29-37.
- [8] Merle Anne Albin: *Fuzzy Sets and Their Application to Medical diagnosis and Pattern Recognition*. Ph D. thesis, University of California, Berkeley, 1975.
- [9] H. Wechsler: Applications of Fuzzy Logic to Medical Diagnosis. In: *Proceedings of the 1975 International Symposium on Multiple-Valued Logic*, IEEE 75CH0959-7C, 1975.
- [10] Tautu, P. and Wagner, G.: The Process of Medical Diagnosis: Routes of Mathematical Investigations. *Methods of Information in Medicine*, Vol. 17, No. 1, 1978, S. 1-10.
- [11] Elder, R. C.: Fuzzy Systems theory and medical decision making. M. S. Thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 1976.
- [12] Perez-Ojeda, A.: *Medical Knowledge Network. A Database for Computer Aided Diagnosis*. Master Thesis, Department of Industrial Engineering, University of Toronto 1976.
- [13] Esogbue, A. O. and Elder, R. C.: Fuzzy Sets and the Modelling of Physician Decision Processes, Part I: The Initial Interview-Information Gathering Session. *Fuzzy Sets and Systems* 2 (1979), 279-291.
- [14] Esogbue, A. O. and Elder, R. C.: Fuzzy Sets and the Modelling of Physician Decision Processes, Part II: Fuzzy Diagnosis Decision Models. *Fuzzy Sets and Systems* 3 (1980), 1-9.
- [15] Moon, R. E., Jordanov, S., Perez, A. and Turksen I. N.: *Medical Diagnostic System with Human-Like Reasoning Capability*. In: D. B. Shires, H. Wolf (eds.): MEDINFO 77. North-Holland Publishing Company Amsterdam – New York – Oxford, 1977, S. 115-119.
- [16] Zadeh, L. A.: Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-3, No. 1, January 1973, S. 28-44.
- [17] Smets, Ph., Vainsel, H., Bernard, R. and Kornreich F.: Bayesian Probability in Fuzzy Diagnosis. In : D. B. Shires, H. Wolf (eds.): MEDINFO 77. North-Holland Publishing Company Amsterdam – New York – Oxford, 1977, S. 213-218.
- [18] Sanchez, Elie: *Equations de Relations Floues*”. Thèse Biologie Humaine, Faculté de Médecine de Marseille. 1974.

- [19] Sanchez, Elie : Resolution of Composite Fuzzy Relation Equations. *Information and Control*, Vol. 30, S. 38-48, 1976.
- [20] Gupta, M. M., Ragade, R. K. an Yager R.R. (eds.): *Advances in Fuzzy Set Theory and Applications*. Amsterdam, New York, Oxford: North-Holland 1979.
- [21] Sanchez, E.: Compositions of Fuzzy Relations. Gupta, M. M., Ragade, R.K. and Yager R.R. (eds.): *Advances in Fuzzy Set Theory and Applications*. Amsterdam, New York, Oxford: North-Holland 1979, S.421-433
- [22] Sanchez, Elie: Medical Diagnosis and Composite Fuzzy Relations. Gupta, M. M., Ragade, R. K. an Yager R. R. (eds.): *Advances in Fuzzy Set Theory and Applications*. Amsterdam, New York, Oxford: North-Holland 1979, S. 437-444.
- [23] Spindelberger, W. und Grabner, G.: *Ein Computerverfahren zur diagnostischen Hilfestellung*. In: K. Fellinger (Hrsg.): *Computer in der Medizin – Probleme, Erfahrungen, Projekte*. Wien: Verlag Brüder Hollinek 1968, S. 189-221.
- [24] Gangl, A., Grabner, G. und Bauer, P.: Erste Erfahrungen mit einem Computerprogramm zur Differentialdiagnose der Lebererkrankungen. *Wien. Z. Inn. Medizin & Ihre Grenzgeb.* 50, 1969, S. 553-586.
- [25] Adlassnig, K.- P.: A fuzzy logical model of computer-assisted medical diagnosis. In: *Methods Inf, Med.* 19, 1980, S. 141.
- [26] Adlassnig, K.- P.: Ein einfaches Modell zur medizinischen Diagnostik mit fuzzy Teilmengen. In: *EDV in Medizin und Biologie* 13 (1) 1982, S. 12-16.
- [27] Adlassnig, K.-P.: A Survey on Medical Diagnosis and Fuzzy Subsets. Gupta, M. M. and Sanchez, E. (eds.): *Approximate Reasoning in Decision Analysis*. North-Holland Publ. Comp. 1982, S: 203-217.
- [28] Adlassnig, K.-P.: A Fuzzy Logical Model of Computer-Assisted Medical Diagnosis. *Methods of Information in Medicine* 19 (1980), 141-148.
- [29] Zadeh, L. A.: A Fuzzy-algorithmic Approach to the Definition of Complex or Imprecise Concepts. Bossel, H., Klaczko, N. und Müller, N.(eds.): *System Theory in the Social Sciences*. Birkhäuser: Basel 1976.
- [30] Adlassnig, K.-P., Horak, W., Kolarz, G., Dorda, W., Fröschl, K. A., Grabner, H. und Grabner, G.: Expertensysteme in der Medizin. *Österreichische Krankenhaus-Zeitung*, 32, 1991, S. 361-384.
- [31] Adlassnig, K.-P.: Medizinische Expertensysteme in der klinischen und Laboratoriumsdiagnostik. In: Peter Hucklenbroich, Richard Toellner (Hrsg.): *Künstliche Intelligenz in der Medizin. Klinisch-methodologische Aspekte medizinischer Expertensysteme*. Stuttgart, Jena, New York: Gustav Fischer 1993, S. 37-67.

Datenbasierte Generierung natürlichsprachlicher Erklärungstexte am Beispiel der Instrumentellen Ganganalyse

Tobias Loose, Jens Jäkel, Ralf Mikut

Forschungszentrum Karlsruhe GmbH, Institut für Angewandte Informatik,
D-76021 Karlsruhe, Postfach 3640,
Telefon: (07247) 82-5757, Fax: (07247) 82-5786, E-Mail: tobias.loose@iai.fzk.de

Zusammenfassung

Bei der Beurteilung von Krankheitsbildern (Diagnose, Therapiefortschritte) bietet die zusätzliche Verwendung entscheidungsunterstützender Systeme Vorteile gegenüber einer alleinigen subjektiven Einschätzung durch einen Mediziner. Der Entwurf derartiger Systeme kann im Wesentlichen datengestützt und weitgehend automatisch erfolgen. Probleme entstehen bei der Merkmalsextraktion aus Rohdaten (z. B. Zeitreihen) und der Interpretation der Ergebnisse.

Dieser Beitrag stellt ein datenbasiertes Entwurfsverfahren für entscheidungsunterstützende Systeme vor.¹ Am Beispiel der Ganganalyse werden Strategieelemente zur Lösung der oben genannten Probleme erläutert. Dabei werden aus gemessenen Zeitreihen Merkmale erzeugt, die anschließend für eine gezielte Fragestellung bewertet werden, wie z. B. zur Erklärung der Hauptprobleme bei Patienten und deren Entwicklung während der Therapie. Bei einer darauf folgenden Generierung von Fuzzy-Regelbasen werden Präferenzen bezüglich der bevorzugten Merkmale berücksichtigt. Die Regeln werden durch Zusatzinformationen erweitert und als natürlichsprachliche Erklärungstexte dargestellt. Somit kann insgesamt die Merkmalsextraktion systematisiert und die Interpretierbarkeit der Ergebnisse verbessert werden.

1 Einleitung

Entscheidungsunterstützende Systeme gewinnen für Diagnoseaufgaben und Therapieentscheidungen bei komplexem pathophysiologischen Verhalten im klinischen Einsatz zunehmend an Bedeutung [1, 2]. Die Motivation für den Einsatz solcher Systeme ergibt sich beispielsweise aus limitierten Aufnahmekapazitäten und subjektiv begründeten Entscheidungen der Untersucher sowie der Notwendigkeit einer langjährigen Erfahrung auf dem Gebiet („Lernkurve“).

Methodische Schwierigkeiten beim datenbasierten Entwurf resultieren aus den teilweise großen Rohdatenmengen, die beispielsweise aus mehreren simultan aufgezeichneten Zeitreihen entstehen. Aus diesen Rohdaten sind zunächst relevante Merkmale zu extrahieren, mit denen ein Klassifikatorentwurf erfolgt. Beide Schritte setzen eine geeignete klinische Problemformulierung (Schnittstelle Kliniker → entscheidungsunterstützendes System) voraus. Die so gefundenen Ergebnisse sind aber nicht von vornherein für einen Kliniker nachvollziehbar, validierbar und verständlich (Schnittstelle entscheidungsunterstützendes System → Kliniker), was aber für die klinische Akzeptanz von entscheidender Bedeutung ist.

¹Dieses Projekt wird seit 2001 durch die Deutsche Forschungsgemeinschaft (DFG) gefördert (GE 1139-1, BR 1303-6).

Wegen der enormen Komplexität und ungeklärter Fragen besteht bei der Instrumentellen Ganganalyse ein großer Bedarf an einer computerbasierten Entscheidungsunterstützung. Die Datenbasis umfasst simultan gemessene Zeitreihen, wie Gelenkwinkelverläufe, Bodenreaktionskräfte und Muskelaktivitäten während des Gehens (Rohdaten). Ungeklärte Fragen sind hierbei u. a. die Erkennung von Hauptproblemen und Kompensationsmechanismen gehbehinderter Patienten sowie die Validierung und Prognose von Therapien. Existierende computerbasierte Methoden in der Instrumentellen Ganganalyse [3] sind meist auf eng festgelegte Fragestellungen und bestimmte Patientengruppen zugeschnitten [4–7]. Weitere Einschränkungen derartiger Methoden sind die subjektiv und problemspezifisch gewählten Merkmale [8, 9] und die im Hinblick auf klinische Vorgehensweisen schwer interpretierbaren Ergebnisse (z. B. Merkmalstransformation durch Hauptkomponentenanalyse [10, 11]).

In einem Gemeinschaftsprojekt zwischen der Orthopädischen Universitätsklinik Heidelberg und dem Forschungszentrum Karlsruhe wird ein systematisches datenbasiertes Vorgehen entwickelt [12], das diese Einschränkungen überwinden soll. Der methodische Rahmen geht von einer automatisierten Nachbildung und Erweiterung der klinischen Analyse aus (Abschnitt 2). Hierbei wird eine Vielzahl interpretierbarer Merkmale verwendet und beibehalten, also nicht weiter transformiert. Durch die Formulierung klinischer Problemstellungen können durch verschiedene (statistische oder unscharfe) Verfahren die wichtigsten Merkmale für das gegebene Problem gesucht und als Ergebnis präsentiert werden. Wegen der Forderung nach Transparenz ist die datenbasierte Generierung eines Fuzzy-Systems aufbauend auf diesen interpretierbaren Merkmalen für die Beantwortung dieser klinischen Fragen prädestiniert [13, 14].

Ein so entwickeltes System enthält zwar relevante Regeln, die Diskussion der Ergebnisse ergibt aber oftmals typische Defizite bezüglich der Transparenz:

- Viele Problemstellungen sind durch verschiedene, in der Klassifikationsgüte nahezu gleichwertige Regelbasen beschreibbar. Die in diesen Regelbasen verwendeten Merkmale unterscheiden sich aber in ihrer Verständlichkeit unter klinischen Gesichtspunkten. Der datenbasierte Entwurf nimmt jedoch keine Rücksicht auf solche Präferenzen.
- Die formalisierte Schreibweise von Fuzzy-Regeln (z. B. mit vollautomatisch erzeugten und bezeichneten Termen) ist nur nach zusätzlichen Erklärungen vermittelbar und verständlich. Zudem stört die starke Einschränkung des Sprachumfangs von Fuzzy-Regeln, weil Bezüge zu den einzelnen Merkmalen und zwischen verschiedenen Ausgangsklassen (z. B. als Vergleich zwischen pathologischem und physiologischem Gangbild) verloren gehen.
- Aussagen über die Regelrelevanzen und -verlässlichkeiten fehlen, was Irritationen in Bezug auf die Wichtigkeit einzelner Regeln in Regelbasen auslöst.

Ähnliche Probleme sind beim vollautomatischen Entwurf entscheidungsunterstützender Systeme auch bei anderen klinischen und nichtklinischen Anwendungen zu erwarten.

Zur Verbesserung dieser Situation wird in diesem Beitrag eine Vorgehensweise vorgeschlagen, bei der

- für die Ganganalyse interpretierbare Merkmale systematisch generiert und problemspezifisch bewertet und verwendet werden (Abschnitt 2),
- die Regel- und Merkmalsbewertung mit zusätzlichen A-priori-Merkmalrelevanzen modifiziert und in die Darstellung der Ergebnisse einbezogen wird (Abschnitt 3) und
- auf dieser Basis natürlichsprachliche Erklärungstexte mit einem erweiterten Sprachumfang entsprechend der Generalized Constrained Language (GCL) [15, 16] generiert werden (Abschnitt 4, [17]).

2 Instrumentelle Ganganalyse und Merkmalsextraktion

Die Instrumentelle Ganganalyse [18] ist ein etabliertes Verfahren zur Quantifizierung von Bewegungen, insbesondere zur Untersuchung neurogener Bewegungsstörungen (ICP: infantile Zerebralparese, SCI: Spinal Cord Injury/Querschnittlähmung) und zur Anpassung von Prothesen. Durch die kommerzielle Verfügbarkeit leistungsfähiger Messsysteme wird die Ganganalyse inzwischen standardmäßig zu klinischen Dokumentationszwecken eingesetzt. Dazu werden Marker an definierten Stellen am Körper befestigt. Ausgehend von der Erfassung der räumlichen Markertrajektorien werden modellbasiert die Winkelverläufe für verschiedene Gelenke (Fuß, Knie, Hüfte, Becken usw.) in drei Ebenen (sagittal: von der Seite, transversal: von oben, frontal: von vorn) berechnet und auf 100% des Gangzyklus (Schritt) normiert in Charts dargestellt.

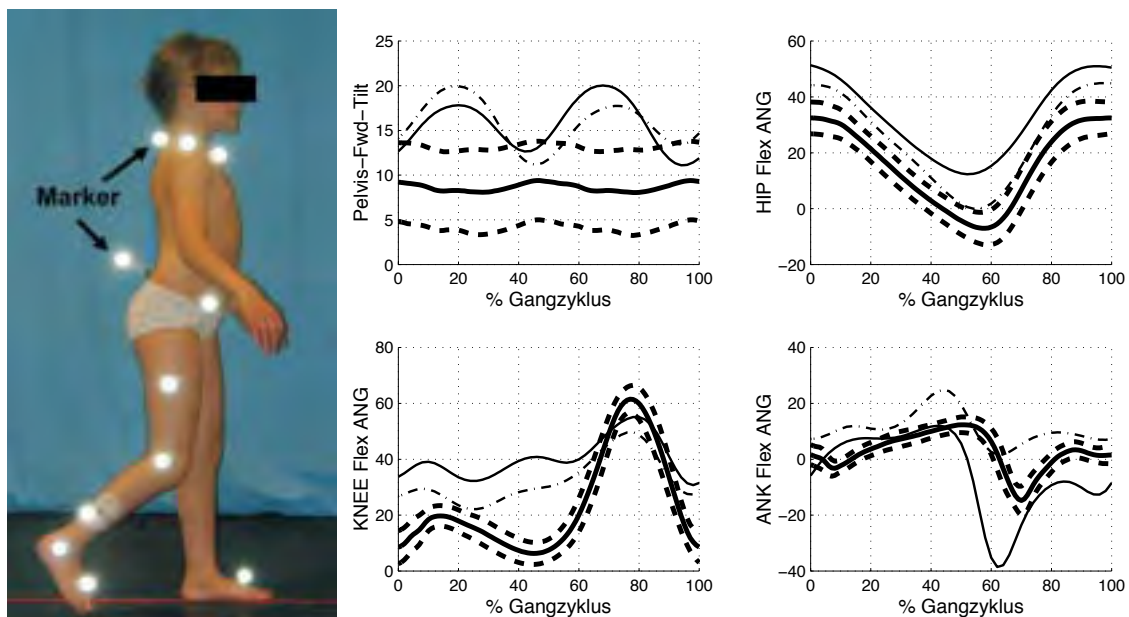


Bild 1: links: Messaufnahme, rechts: Gelenkwinkelverläufe der sagittalen Ebene (Beckenkipplung, Hüft- und Kniebeugung sowie Sprunggelenksflexion). Die dünn dargestellten Zeitreihen sind die rechte (durchgezogen) und die linke Seite (strichpunktiert) eines Patienten. Als Vergleich dienen die fett gedruckten Zeitreihen des Mittelwerts (durchgezogen) und der Standardabweichung (gestrichelt) des gesunden Probanden-Kollektivs bei mittlerer Geschwindigkeit.

In Bild 1 ist ein Kind bei der Messaufnahme der Gelenkwinkel während des Gehens gezeigt sowie ein Auszug von (insgesamt 12) gemessenen Gelenkwinkel-Zeitreihen.

Übliche Abkürzungen sind hier „Pelvis-Fwd-Tilt“ für die Beckenkipfung, „Hip Flex ANG“ für die Hüftbeugung, „Knee Flex ANG“ für die Kniebeugung und „Ank Flex ANG“ für den sagittalen Winkelverlauf des Sprunggelenks.

Der Mediziner schätzt den Patienten durch direktes Beobachten oder durch Hinzunahme dieser Charts ein, um Diagnose- und Therapieentscheidungen zu treffen. Dabei bewertet er aufbauend auf seinen Erfahrungen die Kurvenverläufe nach Form, Abstand zum Normverhalten oder durch Merkmale wie Bewegungsspannen, Maxima, Kurvenanstiege usw.

Diese Herangehensweise ist durch ein entscheidungsunterstützendes System formalisiert nachzubilden, um einen einheitlichen, problemübergreifenden Rahmen für verschiedene klinische Problemstellungen bereitzustellen. Aus den gemessenen Zeitreihen $u_i[k]$ (Verlauf des i -ten Gelenkwinkels mit $k = 0, \dots, 100$ entsprechend der prozentualen Aufteilung des Gangzyklus) werden in einem mehrstufigen Vorgehen interpretierbare Ergebnisse erzielt (Bild 2, [12]):

1. Erzeugung zusätzlicher Zeitreihen, um enthaltene Informationen besser auswerten zu können (z. B. Winkelgeschwindigkeiten oder -beschleunigungen, Seitendifferenzen zwischen linker und rechter Körperseite, Normabweichung)
2. Berechnung von potenziell relevanten Merkmalen aus den originalen und den zusätzlichen Zeitreihen (Extrema und deren Positionen im Gangzyklus, Mittelwert, Bewegungsspanne bzw. Range of Motion, Korrelation zum mittleren Normverhalten) über den gesamten Schritt oder zeitliche Abschnitte wie Stand- bzw. Schwungperiode und deren weitere Unterteilung durch die in der Ganganalyse üblichen acht Schritphasen [18]
3. Relevanzbewertung der erzeugten Merkmale hinsichtlich der jeweiligen Problemstellung (Diagnose, Therapieauswahl oder -validierung)
4. Klassifikatorentwurf.

Anschließend werden die Ergebnisse auf verschiedene Arten präsentiert.

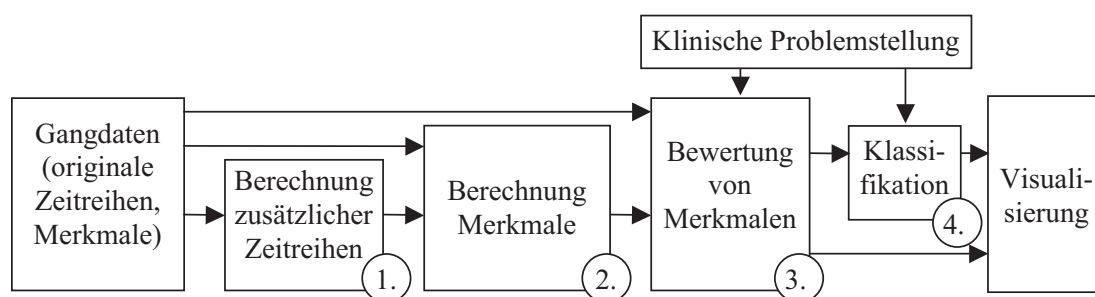


Bild 2: Prinzipieller Auswerteablauf

Bei der Berechnung von Merkmalen über originale und zusätzliche Zeitreihen entstehen viele potenzielle Merkmale. Beispielsweise ergeben sich aus 12 originalen Gelenkwinkel-Zeitreihen und den jeweils zusätzlich berechneten 4 Zeitreihen (insgesamt 60 Zeitreihen) durch die Berechnung von je 7 Merkmalen über 11 Schrittabchnitte insgesamt 4620 Merkmale.

Die nachfolgenden Auswertungen (Bewertung von Merkmalen, Klassifikation) setzen die Formalisierung der klinischen Problemstellung als Klassifikationsproblem voraus. Beispiele sind die Erkennung von Therapieeffekten (Unterscheidung prä- und posttherapeutisch, 2 Klassen), die Charakterisierung von Patienten- und Probandengruppen (2 oder mehrere Klassen) und die Therapieprognose (erfolgreich bzw. nicht erfolgreich, 2 oder mehrere Klassen). In dieser Arbeit wird beispielhaft ein 4-Klassen-Problem (ICP-Patienten sowie Normprobanden mit mittlerer, langsamer und schneller Gehgeschwindigkeit) für die klinische Problemstellung „durch welche Merkmale und Regeln sind die ICP-Patienten im Vergleich zum gesunden Gehverhalten charakterisiert?“ behandelt.²

Eine Relevanzbewertung der Merkmale kann mit statistischen Standardverfahren wie einer univariaten (z. B. ANOVA: ein Merkmal) oder multivariaten (MANOVA: Gruppe von Merkmalen) Analyse oder informationstheoretischen Maßen erfolgen [13, 14]. Für die Klassifikation können Fuzzy-Verfahren oder statistische Klassifikatoren verwendet werden.

Im Ergebnis entstehen eine nach Relevanz geordnete Liste von Merkmalen, deren Messwerte sich für die Ausgangsklassen des festgelegten Problems möglichst deutlich unterscheiden, sowie Regelbasen bzw. Klassifikatoren zur Erklärung des Problems. Diese Informationen können auch grafisch dargestellt werden.

3 Informationsgewinnung beim Klassifikatorentwurf

3.1 Überblick

Das Fuzzy-System wird datenbasiert durch

- den automatisierten Entwurf von Zugehörigkeitsfunktionen anhand der Datenverteilung inkl. einer voll automatisierten Benennung der Terme [19],
- die Induktion von Entscheidungsbäumen sowie die Bewertung und Auswahl von linguistischen Variablen (hier die oben berechneten Merkmale),
- die Extraktion von Regelhypothesen aus den Entscheidungsbäumen,
- das Regelpruning (Löschen linguistischer Variablen sowie ODER-Verknüpfung benachbarter linguistischer Terme in den Regelprämissen) mit der Regelbewertung,
- die Auswahl von Regeln als Regelbasis

entworfen [13, 14]. Die im Folgenden verwendeten $k = 1, \dots, K$ Regeln haben allgemein die Struktur

$$R_k : \text{WENN } \underbrace{x_1 = A_{1,Rk}}_{\text{Teilprämisse } P_{k1}} \text{ UND } \dots \text{ UND } \underbrace{x_s = A_{s,Rk}}_{\text{Teilprämisse } P_{ks}} \text{ DANN } \underbrace{y = B_{Rk}}_{\text{Konklusion } C_k}$$

Prämisse P_k

wobei x_l ($l = 1, \dots, s$) die l -te linguistische Eingangsvariable (Merkmal) mit den

²Dazu werden die Daten von 43 ICP-Patienten und 10 gesunden Probanden mit unterschiedlichen Gehgeschwindigkeiten verwendet [12].

zugehörigen linguistischen Termen $A_{l,i}$ ($i = 1, \dots, m_l$) ist und y die linguistische Ausgangsvariable mit den Termen B_i (z. B. für das 4-Klassen-Problem B_1 : ICP-Patienten, B_2 : Norm mittel, B_3 : Norm langsam, B_4 : Norm schnell). Die Prämisse P_k besteht aus UND-Verknüpfungen der Teilprämissen P_{k1} bis P_{ks} . Der Term $A_{l,Rk}$ enthält entweder nur einen linguistischen Term oder mehrere bzw. alle m_l ODER-verknüpften linguistischen Terme des Merkmals x_l . Bei ODER-Verknüpfung aller Terme beeinflusst die Teilprämisse die Regelaktivierung nicht, weil wegen der hier verwendeten dreieck- bzw. trapezförmigen Zugehörigkeitsfunktionen die Teilprämisse immer erfüllt ist.

Aufgrund der hohen Anzahl von linguistischen Variablen (Merkmalen) ist eine automatisierte Festlegung der Namen der linguistischen Terme $A_{l,i}$ und der Stützpunkte ihrer Zugehörigkeitsfunktionen $a_{l,i}$ sinnvoll. Die Bestimmung der Stützpunkte basiert auf einer angestrebten Gleichverteilung der Beispiele für die verschiedenen Terme, wobei auch Interpretierbarkeiten von Stützpunkten berücksichtigt werden [13, 19]. Die Benennung der Terme orientiert sich an den Werten der Stützpunkte (Tabelle 1). Ein Term mit $a_{l,i} = 0$ wird mit NU bezeichnet. Die Bezeichnung von Termen mit $a_{l,i} > 0$ richtet sich nach deren Anzahl (1 Term: POS, 2 Terme: PK-PG, 3 Terme PK-PM-PG, 4 Terme PK-PM-PG-PSG, 5 Terme PSK-PK-PM-PG-PSG). Bei Termen mit $a_{l,i} < 0$ wird analog vorgegangen. Allerdings geht bei diesem automatischen Entwurf die subjektiv empfundene, intuitive Zuordnung verloren. Die Alternative einer manuellen Zuordnung ist aber nicht akzeptabel.

$\mu(a_{l,i}) = 1, a_{l,i} > 0$	$\mu(a_{l,i}) = 1, a_{l,i} < 0$	$\mu(a_{l,i}) = 1, a_{l,i} = 0$
POS: positiv	NEG: negativ	NU: ungefähr null
PSK: sehr klein	NSK: etwas kleiner als null	-
PK: klein	NK: leicht negativ	-
PM: mittel	NM: mäßig negativ	-
PG: groß	NG: stark negativ	-
PSG: sehr groß	NSG: sehr stark negativ	-

Tabelle 1: Kurz- und Langbezeichnung der linguistischen Terme in Abhängigkeit von den Stützpunkten der Zugehörigkeitsfunktionen $a_{l,i}$

3.2 Modifikationen durch A-priori-Merkmalrelevanzen

Bei der Induktion von Entscheidungsbäumen sowie der Bewertung und Auswahl von Merkmalen wird zusätzlich zur problembezogenen Merkmalsrelevanz M_l [13, 14] auch eine problemunabhängige A-priori-Merkmalrelevanz $M_{l,ap}$ in die Berechnung der Gesamt-Merkmalrelevanz $M_{l,ges}$ einbezogen. Die Motivation besteht darin, im Falle problembezogen nahezu gleichwertiger Merkmale diejenigen Merkmale zu bevorzugen, die sich beispielsweise durch eine bessere Interpretierbarkeit, einen geringeren Messaufwand, eine bessere Berechnungseffizienz oder eine höhere messtechnische Zuverlässigkeit auszeichnen. Die Verknüpfung beider Komponenten erfolgt multiplikativ:

$$M_{l,ges} = M_l \cdot M_{l,ap}. \quad (1)$$

Diese Berechnungsvorschrift wird sowohl zur Merkmalsselektion als auch zur Festlegung der Auftrennungsmerkmale in den Knoten der Entscheidungsbäume herangezogen. In beiden Fällen ist M_l das Verhältnis der Transinformation zur Ausgangsentropie.

Im Beispiel der Ganganalyse dominieren die klinische Akzeptanz (Erhöhung der Interpretierbarkeit) und die Zuverlässigkeit der Messungen bei der Festlegung der A-priori-Merkmalsrelevanzen. Um diese nicht für jedes Merkmal manuell festlegen zu müssen, werden Kategorien zur Einteilung von Merkmalen herangezogen:

1. Art der Zeitreihe (originale oder zusätzliche Zeitreihe, z. B. Gelenkwinkel, -winkelgeschwindigkeiten, Normabweichungen) – $M_{ap,ZR}$,
2. Gelenke (wie Becken, Hüfte, Knie, Fuß) – $M_{ap,Gel}$,
3. Ebene (sagittal, frontal, transversal) – $M_{ap,Eb}$,
4. zeitlicher Abschnitt (ganzer Schritt, nur Stand- oder Schwungperiode, Schrittphasen) – $M_{ap,Zeit}$,
5. Art des Merkmals (Minimum, Maximum, Mittelwert usw.) – $M_{ap,AM}$.

Ein Merkmal nimmt pro Kategorie genau einen Wert an und lässt sich durch die Kombination der Werte aller Kategorien eindeutig beschreiben. Die A-priori-Merkmalsrelevanzen der Werte aller Kategorien sind Eins mit Ausnahme

- der frontalen und transversalen Ebene mit je $M_{ap,Eb} = 0.8$, die bei der klinischen Analyse eine etwas geringere Rolle als die sagittale Ebene spielen,
- der in der klinischen Praxis weniger gebräuchlichen Winkelgeschwindigkeiten mit $M_{ap,ZR} = 0.8$ sowie
- der zeitlichen Abschnitte der Stand- und Schwungperiode mit je $M_{ap,Zeit} = 0.8$ und der Schrittphasen mit je $M_{ap,Zeit} = 0.6$, die aufgrund der automatischen Zeiteinteilung fehleranfälliger sind.

Zusätzlich werden Merkmale mit einer messtechnisch unzuverlässigeren (ungünstige Signal-Stör-Verhältnisse) Kombination von Kategorien $M_{ap,Kombi}$ unterdrückt. Beispielsweise sind Messwerte in der frontalen und transversalen Ebene für das Knie und in der frontalen Ebene für das Fußgelenk aus anatomischer Sicht während des Gehens nahezu irrelevant (originale Zeitreihen der Gelenkwinkel jeweils $M_{ap,Kombi} = 0.6$ sowie deren Winkelgeschwindigkeiten jeweils $M_{ap,Kombi} = 0.1$). Somit kann auch die Auswahl von klinisch kaum akzeptierten Merkmalen wie Range of Motion von Geschwindigkeiten vermieden werden ($M_{ap,Kombi} = 0.1$). Die Verknüpfung erfolgt für jedes Merkmal über eine Multiplikation der jeweiligen A-priori-Merkmalsrelevanzen der Kategorien:

$$M_{l,ap} = M_{ap,ZR}(l) \cdot M_{ap,Gel}(l) \cdot M_{ap,Eb}(l) \cdot M_{ap,Zeit}(l) \cdot M_{ap,AM}(l) \cdot M_{ap,Kombi}(l). \quad (2)$$

Beispiel: Das berechnete Merkmal „Mittlere Winkelgeschwindigkeit der Knieflexion in der Schwungperiode“ (abgekürzt: „MEAN SW Knee Flex ANG V“, vgl. auch Abschnitt 2) nimmt in den Kategorien folgende Werte an: Art der Zeitreihe: Geschwindigkeit ($M_{ap,ZR} = 0.8$), Gelenk: Knie ($M_{ap,Gel} = 1$), Ebene: sagittal ($M_{ap,Eb} = 1$), zeitlicher Abschnitt: Schwungperiode ($M_{ap,Zeit} = 0.8$), Art des Merkmals: Mittelwert ($M_{ap,AM} = 1$), spezielle Kombinationen von Kategorien: keine ($M_{ap,Kombi} = 1$). Daraus folgt mit (2) die A-priori-Merkmalsrelevanz $M_{l,ap} = 0.64$.

3.3 Relevanzinformationen beim Klassifikatorentwurf

Für das 4-Klassen-Problem aus Abschnitt 2 wird eine Regelbasis bestehend aus 7 Einzelregeln gefunden. Eine 10-fache Cross-Validierung schätzt eine mittlere Fehlklassifikation für dieses Problem von ca. 3.5%. Die wichtigste Einzelregel

R_1 : WENN ROM STRI Pelvis-Fwd-Tilt = PM ODER PG ODER PSG
 UND MAX STRI Knee Flex ANG V = PSK ODER PK ODER PM
 DANN Person = ICP-Patient

charakterisiert das ICP-Verhalten, Bild 3. Die dabei verwendeten Abkürzungen beziehen sich auf die Kategorien Art der Zeitreihe („V“ für Winkelgeschwindigkeit, ohne Zusatz für originale Zeitreihen), Gelenk („Pelvis“ für Becken und „Knee“ für Knie), Ebene („Fwd-Tilt“ und „Flex“ für sagittale Ebene), zeitlicher Abschnitt („STRI“ für gesamten Schritt) und Art des Merkmals (Bewegungsspanne „ROM“, Maximum „MAX“). Die linguistischen Terme sind Tabelle 1 zu entnehmen.

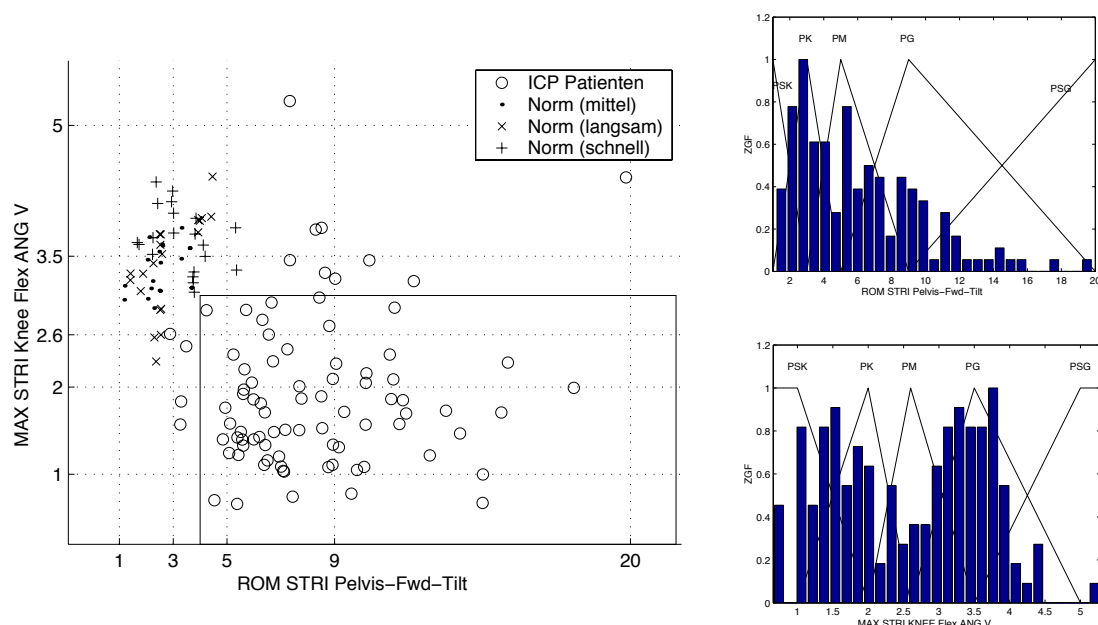


Bild 3: Wichtigste Regel zur Charakterisierung des ICP-Verhaltens als 2D-Darstellung (links) und Zugehörigkeitsfunktionen mit Häufigkeitsverteilung (rechts)

Diese Regel ist aufgrund der verwendeten Merkmale aus Abschnitt 2 prinzipiell klinisch interpretierbar. In diesem Beispiel deutet die *große* Bewegungsspanne im Becken über den gesamten Schritt auf das *meistens* vorhandene ICP-typische „double bump pattern“ (Beckenverlauf mit einer erhöhten Amplitude und zwei charakteristischen Maxima, Bild 1) hin. Die *kleine* maximale Knie-Winkelgeschwindigkeit bedeutet ein eher steifes Knie, Bild 1. Allerdings ist die formale Regeldarstellung gewöhnungsbedürftig und nicht dem natürlichen Sprachgebrauch angepasst. Weiterhin sind weder die Relevanz der Regel und der Merkmale (Fehlklassifikationen, erfasste Datensätze) noch die Aufteilung in die weiteren Ausgangsklassen (hier die 3 Norm-Klassen) auf Anhieb ersichtlich. Die merkmalspezifische Festlegung der linguistischen Terme ist aus dem Regeltext ebenfalls nicht zu entnehmen. Diese Informationen können bestenfalls mit Hilfe einer Grafik abgeschätzt werden, Bild 3.

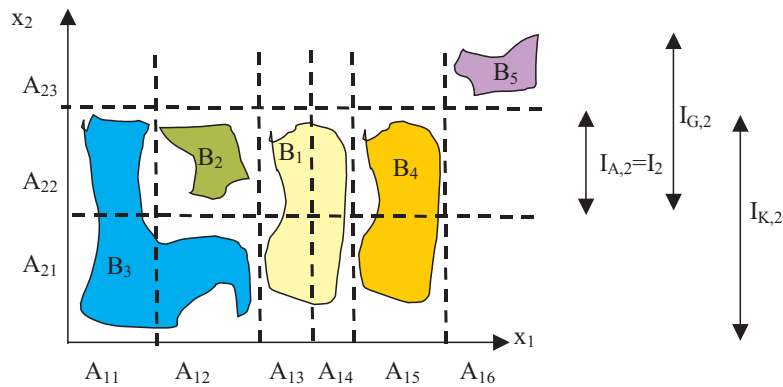


Bild 4: Exemplarische Beziehungen zwischen zwei linguistischen Variablen x_1, x_2 , der Abdeckungsbereich (α -Schnitt bei 0.5) ihrer zugehörigen linguistischer Terme A_{11}, \dots, A_{16} , bzw. A_{21}, \dots, A_{23} , die Abdeckungsbereiche der fünf Ausgangsklassen $B_{1, \dots, 5}$ und Beispiel für die interessierenden Bereiche $I_{A,2}, I_{G,2}, I_{K,2}$ zum Vergleich der Ausgangsklasse B_2 mit allen anderen Ausgangsklassen für das Merkmal x_2

Ansätze zur systematischen Einbeziehung solcher Zusatzinformationen werden im Folgenden anhand eines illustrativen Beispiels mit zwei Merkmalen und fünf Ausgangsklassen erläutert, Bild 4. Offene Fragen, die sich auf solche Zusatzinformationen beziehen, sind:

1. Wie relevant ist ein verwendetes Merkmal oder eine Kombination von Merkmalen zur Klassifizierung bzw. Beschreibung von allen oder einzelnen Ausgangsklassen? Beispiel: Durch die Variable x_1 können mehrere Ausgangsklassen eindeutig unterschieden werden (B_1, B_4, B_5), x_1 ist also relevanter als x_2 (nur Erkennung von B_5). Allerdings ist die eindeutige Detektion *aller* Ausgangsklassen, insbesondere die Unterscheidung der Klassen B_2 und B_3 , nur durch die Kombination beider Variablen $\{x_1, x_2\}$ möglich.
2. Lassen sich anhand der Ausgangsklassen Rückschlüsse auf einzelne linguistische Terme einer Variablen der Regelprämisse ziehen (Beziehung WENN Ausgangsklasse DANN linguistischer Term des Merkmals)? Beispiel: WENN $y = B_4$ DANN $x_1 = A_{15}$ im Gegensatz zur irrelevanten Regel WENN $y = B_1$ DANN $x_1 = A_{13}$.
3. Wieviele Datensätze der jeweiligen Ausgangsklasse in der Konklusion werden durch die Regelprämisse abgedeckt? Beispiel: alle Datensätze der betreffenden Ausgangsklasse in WENN $x_1 = A_{12}$ UND $x_2 = A_{22}$ DANN $y = B_2$ im Vergleich zu WENN $x_1 = A_{11}$ DANN $y = B_3$.
4. Wieviele Fehlklassifikationen werden durch eine Regel verursacht und welche Ausgangsklassen sind betroffen? Beispiel: Durch alleinige Betrachtung von $x_1 = A_{12}$ lassen sich die beiden Ausgangsklassen B_2 und B_3 nicht eindeutig trennen, eine Regel mit einer solchen Prämisse verursacht immer Fehlklassifikationen.
5. Existieren alternative Merkmale anstatt des verwendeten Merkmals in der Regelprämisse? Beispiel: WENN $x_1 = A_{16}$ DANN $y = B_5$ kann ohne Einschränkung durch WENN $x_2 = A_{23}$ DANN $y = B_5$ ersetzt werden.
6. Ist jede einzelne Teilprämisse notwendig? Beispiel: Beide Teilprämissen WENN

$x_1 = A_{12}$ UND $x_2 = A_{22}$ DANN $y = B_2$ sind im Gegensatz zu redundanten Teilprämissen wie WENN $x_1 = A_{16}$ UND $x_2 = A_{23}$ DANN $y = B_5$ erforderlich.

7. Wie unterscheiden sich die Messwerte eines Merkmals für eine Ausgangsklasse qualitativ von denen für andere Ausgangsklassen? Beispiel: Merkmal x_1 hat für Klasse B_3 meistens kleinere, für B_5 immer größere und für B_1 immer andere Werte als die anderen Ausgangsklassen.

Die Beantwortung dieser Fragen basiert auf den (relativen) bedingten Häufigkeiten und informationstheoretischen Maßen, die während des Entwurfs (Entscheidungsbaum, Regelpruning) berechnet werden [13, 14]:

1. Die Transinformation als ein Maß für den Informationsfluss zwischen Ein- und Ausgangsvariable beschreibt die Relevanz der Eingangsvariablen zur Klassifikation der Ausgangsklassen.
2. Die bedingte Häufigkeit eines linguistischen Terms bei gegebener Ausgangsklasse $h(x_l = A_{l,i}|y = B_j)$ charakterisiert typische Werte dieses Merkmals für die jeweilige Ausgangsklasse.
3. Die bedingte Häufigkeit einer Prämisse bei gegebener Konklusion $h(P_k|C_k)$ ist ein Indikator für den Abdeckungsbereich der Regelprämisse.
4. Die bedingte Häufigkeit der Konklusion bei gegebener Prämisse $h(C_k|P_k)$ misst die Korrektheit einer Regel, d. h. die relative Anzahl von richtig klassifizierten Datensätzen. Falls Fehlklassifikationen vorkommen ($h(C_k|P_k) < 1$), dann gibt $h(y = B_j|P_k) \neq 0$ die betroffenen Ausgangsklassen $B_j \neq B_{Rk}$ an.
5. Die Transinformation zwischen verschiedenen Eingangsvariablen gibt Beziehungen zwischen den Variablen an, d. h. ob sie gegeneinander ersetzt werden können.
6. Die Relevanz einer Teilprämisse wird durch die Anzahl von Fehlklassifizierungen bei einer geprunten Regel ohne diese Teilprämisse gemessen.
7. Zum qualitativen Vergleich werden drei Hypothesen untersucht, ob die Werte eines Merkmals x_l für die Ausgangsklasse B_j signifikant größer (G), kleiner (K) oder anders (A) im Vergleich zu allen anderen Ausgangsklassen sind. Dazu werden zunächst für $y = B_j$ die relevanten Terme mit $I_l = \{i \in [1, m_l] | h(x_l = A_{l,i}|y = B_j) > \alpha\}$ (hier $\alpha = \frac{1}{m_l}$) gesucht und die interessierenden Bereiche für die Hypothesen *größer* (alle Terme vom kleinsten relevanten Term bis zum größten Term: $I_{G,l} = \{\min(I_l), \dots, m_l\}$), *kleiner* (alle Terme vom kleinsten Term bis zum größten relevanten Term: $I_{K,l} = \{1, \dots, \max(I_l)\}$) und *anders* (alle relevanten Terme: $I_{A,l} = I_l$) festgelegt. Als Maß für die Hypothesen wird im interessierenden Bereich die Summe der Häufigkeiten der Klasse B_j mit der Summe der Häufigkeiten für die negierte Ausgangsklasse \bar{B}_j verglichen:

$$V_{H_l} = \sum_{i=I_{H,l}} h(x_l = A_{l,i}|y = B_j) - h(x_l = A_{l,i}|y = \bar{B}_j), H = \{G, K, A\}. \quad (3)$$

Die beste der drei Hypothesen H_l^* weist den größten Wert $V_{H_l}^*$ auf. Bei $V_{H_l} = 1$ liegen alle Beispiele der Ausgangsklasse B_j und kein Beispiel der negierten Klasse im interessierenden Bereich. Wenn es im interessierenden Bereich keine Unterschiede gibt, gilt $V_{H_l} = 0$. Bei $V_{H_l} < 0$ ist die Hypothese falsch. Ein

Beispiel für die interessierenden Bereiche $I_{A,2}, I_{G,2}, I_{K,2}$ der Ausgangsklasse B_2 für das Merkmal x_2 zeigt Bild 4, mögliche Schlussfolgerungen H_2^* sind je nach konkreter Datenverteilung *anders* bzw. *kleiner*.

Aufgrund der graduellen Zuordnung werden die bedingten Häufigkeiten $h(A|B)$ über Diskretisierung (α -Schnitt bei $\alpha = 0.5$), Verwendung des Produkt- bzw. Summen-Operators für UND- bzw. ODER-Verknüpfungen oder über ein restringiertes Optimierungsproblem geschätzt [19, 20].

Für die oben ermittelte Regel R_1 zur Charakterisierung von ICP-Patienten ergeben sich durch Diskretisierung die bedingten Häufigkeiten

$$h(P_1|C_1) = 0.85, \quad h(C_1|P_1) = 1.00 \quad (4)$$

$$h(x_{298} = A_{298,1...5} | y = B_1) = (0.00 \quad 0.05 \quad 0.37 \quad 0.51 \quad 0.07) \quad (5)$$

$$h(x_{540} = A_{540,1...5} | y = B_1) = (0.36 \quad 0.39 \quad 0.15 \quad 0.08 \quad 0.02). \quad (6)$$

Somit deckt die Regel 85% der Beispiele für ICP-Patienten ab. Es kommen keine Fehlklassifizierungen vor (4). Der linguistische Term $A_{298,4}$ tritt mit einer relativen Häufigkeit von 0.51 bei Betrachtung der ICP-Patienten auf (5) usw.

4 Automatisierte Textgenerierung für Regeln

Einige der in Abschnitt 3.1 ermittelten Informationen werden nun als Bausteine zur Erweiterung der Regeln R_k eingebunden. Neben der Vermittlung von Zusatzinformationen werden weitere Ziele wie eine Annäherung an den menschlichen Sprachgebrauch sowie eine sprachliche Komprimierung unter Beibehaltung des Informationsgehalts verfolgt.

Die Zahlenwerte der bedingten Häufigkeiten werden in linguistische Terme umgesetzt, siehe Tabelle 2 und [21, 22]. Hierbei wird zwischen der geschätzten Wahrscheinlichkeit und der relativen Anzahl der Beispiele unterschieden.

Relative Häufigkeit h	linguistischer Term (geschätzte Wahrscheinlichkeit)	linguistischer Term (relative Anzahl von Beispielen)
[0, 0.025]	nie	keine
(0.025, 0.20]	selten	wenige
(0.20, 0.50]	gelegentlich	einige
(0.50, 0.80]	häufig	viele
(0.80, 0.975]	meistens	fast alle
(0.975, 1.00]	immer	alle

Tabelle 2: Linguistische Terme zur Beschreibung der relativen Häufigkeiten

Zur Erklärung der Regel R_k werden in einem ersten Textrahmen zunächst die in der Regelprämisse vorkommenden Merkmale erläutert:

Die Beispiele für AUSGANGSKLASSE B_{R_k} sind dadurch charakterisiert, dass MERKMAL x_l MASZ_HYPOTHESE $V_{H_l}^$ HYPOTHESE H_l^* als sonst ist: BEDINGTE_HÄUFIGKEIT $h(x_l = A_{l,L_1} | C_k)$ TERM A_{l,L_1} und ... und BEDINGTE_HÄUFIGKEIT $h(x_l = A_{l,L_v} | C_k)$ TERM A_{l,L_v} .*

Die Bezeichnungen der Merkmale werden stückweise aus den Bezeichnungen der originalen Zeitreihen und je nach Kategorie um Textbausteine für zusätzliche Zeitreihen, die Art des Merkmals und den zeitlichen Abschnitt zusammengesetzt und als Langtext ausgeschrieben. Die Bezeichnungen der linguistischen Terme sind aus Tabelle 1 zu entnehmen. Zur Verbesserung der Lesbarkeit werden die relevanten Terme aus I_l nach absteigender relativer Häufigkeit $h(x_l = A_{li}|C_k)$ sortiert (L_1, \dots, L_v) und Terme mit einer relativen Häufigkeit kleiner 0.2 weggelassen.

Als weiterer Textbaustein wird die beste der drei Hypothesen H_l^* mit der Bezeichnung *größer*, *kleiner* bzw. *anders* verwendet. Für das zugehörige Maß wird die entsprechende Bezeichnung für Häufigkeiten aus Tabelle 2 eingesetzt. Die Vergleichsinformation wird nur für $V_{H_l^*} > 0.2$ angezeigt. Ansonsten wird diese Information ersatzlos weggelassen und ein leicht modifizierter Textbaustein verwendet.

Eine verbesserte Erklärung der Bedeutung der Terme bewirkt die zusätzliche Angabe des Abdeckungsbereiches anhand des α -Schnitts ($\alpha = 0.5$). Für Randterme werden die Bausteine *kleiner als* und *größer als*, für Mittelterme *zwischen ... und ...* genutzt. Die anhand der Rohdaten und der Kategorien ermittelten physikalischen Einheiten werden optional als weitere Textbausteine angehängt. Beispielsweise haben die Gelenkwinkel die Einheit [°] und Winkelgeschwindigkeiten die Einheit [°/°].

Für eventuell folgende Merkmale wird die erste Zeile dieses Textrahmens durch „*Weiterhin gilt für diese Beispiele, dass*“ ersetzt.

Der zweite Textrahmen beschreibt die eigentliche Regel:

Aus diesen Aussagen lässt sich eine Regel aufbauen, die BEDINGTE_HÄUFIGKEIT_DER_PRÄMISSE $h(P_k|C_k)$ Fälle von AUSGANGSKLASSE B_{Rk} beschreibt. Wenn TEILPRÄMISSE P_{k1} ist und ... TEILPRÄMISSE P_{ks} ist, folgt daraus BEDINGTE_HÄUFIGKEIT_DER_KONKLUSION $h(C_k|P_k)$ AUSGANGSKLASSE B_{Rk} .

Zur Komprimierung des Inhalts werden ODER-verknüpfte hintereinander stehende Terme (*erster Term*, *zweiter Term*, *dritter Term*, ... oder *letzter Term*) durch *erster Term* bis *letzter Term* inklusive des daraus resultierenden Abdeckungsbereiches abgekürzt.

Somit wird für die gefundene Regel aus Abschnitt 3.3 der Erklärungstext generiert:

Die Beispiele für ICP-Patient sind dadurch charakterisiert, dass der Range of Motion der Beckenkipfung während des Schrittes (x_{298}) meistens größer als sonst ist: häufig groß (zwischen 7° und 14.5°) und gelegentlich mittel (zwischen 4° bis 7°). Weiterhin gilt für diese Beispiele, dass der maximale Wert der zeitlichen Änderung der Kniegelenksflexion bzw. -extension während des Schrittes (x_{540}) häufig kleiner als sonst ist: gelegentlich sehr klein (kleiner als 1.5°/%) und gelegentlich klein (zwischen 1.5°/% und 2.3°/%).
Aus diesen Aussagen lässt sich eine Regel aufbauen, die fast alle Fälle von ICP-Patient beschreibt: Wenn der Range of Motion der Beckenkipfung während des Schrittes mittel bis sehr groß (größer als 4°) ist und der maximale Wert der zeitlichen Änderung der Kniegelenksflexion bzw. -extension während des Schrittes sehr klein bis mittel (kleiner als 3.05°/%) ist, folgt daraus immer ICP-Patient.

Analog dazu können alle Regeln als automatisiert generierter Text dargestellt werden. Damit werden dem Benutzer zusätzliche relevante Informationen auf eine verständliche Weise vermittelt, die durch die alleinige textuelle Darstellung der WENN-DANN-Regel wie in Abschnitt 3.3 nicht erfassbar sind.

5 Zusammenfassung und Ausblick

In dieser Arbeit wird ein Ansatz zur automatisierten Generierung von Erklärungstexten am Beispiel der Instrumentellen Ganganalyse vorgestellt. Ausgehend von einem datenbasierten Entwurf eines Fuzzy-Klassifikators werden die Regeln durch verschiedene Zusatzinformationen ergänzt, die auf bedingten Häufigkeiten und informationstheoretischen Maßen beruhen. Zusätzlich wird eine verbesserte Auswahl von Merkmalen über A-priori-Merkmalrelevanzen erreicht. Durch die Einführung von Vergleichsoperationen gelingt es, die Beispiele der betrachteten Klasse zu den restlichen Ausgangsklassen in Relation zu setzen. Damit wird erfolgreich gezeigt, dass ausgehend von einer Vielzahl interpretierbarer Merkmale grundlegende Eigenschaften der betrachteten Ausgangsklasse kurz und verständlich formalisiert werden können.

Weitere Zusatzinformationen, wie die Regelrelevanz auf der Basis von Binomialverteilungstests und zusammengefasste Aussagen über Kategorien von Merkmalen, lassen sich in ähnlicher Art und Weise einbinden. Mit dem letztgenannten Strategieelement können abstrakte Aussagen wie „das Beckenverhalten unterscheidet Patient von Norm“ generiert werden, die über die Aussagen von einzelnen Merkmalen hinausgehen. Zur weiteren Verbesserung der Lesbarkeit ist die Monotonie der Textbausteine zu überwinden, die nach dem hier vorgestellten Schema beliebig aneinander gereiht werden können bis zur vollständigen Erklärung aller Fälle. Bestimmte grammatikalische Formulierungen zur Kopplung linguistischer Terme und Variablen (gerade themenspezifische Bezeichnungen) in den Textrahmen steigern zudem die Lesbarkeit.

Die hier vorgestellte Vorgehensweise eignet sich besonders gut für Probleme, die eine Merkmalsextraktion aus Zeitreihen erfordern, und lässt sich auf andere klinische und nichtklinische Anwendungen übertragen.

Literatur

- [1] Adlassnig, K.-P.: Fuzzy Set Theory and Fuzzy Logic in Medicine. In: *Proc. 7th Europ. Congr. on Intelligent Techniques and Soft Computing EUFIT'99, Sep. 13-16, 1999*. Aachen. 1999.
- [2] Seising, R.: Eine kleine Geschichte der Fuzzy-Systeme in der Medizin. In: *Proc. 12. Workshop Fuzzy Systeme, Dortmund*. Forschungszentrum Karlsruhe, FZKA 6767. 2002.
- [3] Chau, T.: A Review of Analytical Techniques for Gait Data. *Gait and Posture* 13 (2001), S. 49–66 (Part 1); 102–120 (Part 2).

- [4] Bertani, A.; Cappello, A.; Benedetti, M. G.; Simoncini, L.; Catani, F.: Flat Foot Functional Evaluation Using Pattern Recognition of Ground Reaction Data. *Clinical Biomechanics* 14 (1999), S. 484–493.
- [5] Deluzio, K.; Wyss, U.; Costigan, P.; Sorbie, C.; Zee, B.: Gait Assessment in Unicompartamental Knee Arthroplasty Patients: Principal Component Modelling of Gait Waveforms and Clinical Status. *Human Movement Science* 18 (1999), S. 701–711.
- [6] Schutte, L.; Narayanan, U.; Stout, J.; Selber, P.; Gage, J.; Schartz, M.: An Index for Quantifying Deviations from Normal Gait. *Gait and Posture* 11 (2000), S. 25–31.
- [7] Kelly, I.; O'Regan, M.; Jenkinson, A.; O'Brian, T.: The Quality Assessment of Walking in Cerebral Palsy. *Gait and Posture* 5 (1997), S. 70–74.
- [8] Davids, J.; Bagley, A.; Bryan, M.: Kinematic and Kinetic Analysis of Running in Children with Cerebral Palsy. *Developmental Medicine and Child Neurology* 40 (1998), S. 528–535.
- [9] Sutherland, D.; Kaufman, K.; Wyatt, M.; Chambers, H.; Mubarak, S.: Double-Blind Study of Botulinum A Toxin Injections Into the Gastrocnemius Muscle in Patients with Cerebral Palsy. *Gait and Posture* 10 (1999), S. 1–9.
- [10] Olney, S.; Griffin, M.; McBride, I.: Multivariate Examination of Data From Gait Analysis of Persons With Stroke. *Physical Therapy* 78 (1998), S. 814–828.
- [11] Zanchi, V.; Papic, V.; Cecic, M.: Quantitative Human Gait Analysis. *Simulation Practice and Theory* 8 (2000), S. 127–140.
- [12] Loose, T.; Malberg, H.; Mikut, R.; Dieterle, J.; Schablowski, M.; Wolf, S.; Abel, R.; Döderlein, L.; Rupp, R.: Ein modulares Verfahren zur automatisierten Auswertung von Ganganalysedaten. *Biomedizinische Technik* 47, Ergänzungsband 1 (2002), S. 700–703.
- [13] Mikut, R.; Malberg, H.; Peter, N.; Jäkel, J.; Gröll, L.; Bretthauer, G.; Abel, R.; Döderlein, L.; Rupp, R.; Schablowski, M.; Siebel, A.; Gerner, H.-J.: *Diagnoseunterstützung für die Instrumentelle Ganganalyse (Projekt GANDI)*. Forschungszentrum Karlsruhe, FZKA 6613, <http://bibliothek.fzk.de/zb/berichte/FZKA6613.pdf>. 2001.
- [14] Mikut, R.; Peter, N.; Bretthauer, G.; Rupp, R.; Abel, R.; Siebel, A.; Döderlein, A.; Gerner, H.-J.: Fuzzy-Regelgenerierung und Multivariate Statistische Verfahren zur Schritt-Phasenerkennung in der Instrumentellen Ganganalyse. In: *Proc. 10. Workshop Fuzzy Control des GMA-FA 5.22, Dortmund*, S. 112–125. Forschungszentrum Karlsruhe, FZKA 6509. 2000.
- [15] Zadeh, L. A.: From Computing with Numbers to Computing with Words - From Manipulation of Measurements to Manipulation of Perceptions. *IEEE Trans. on Circuits and Systems, I: Fundamental Theory and Applications* 45 (1999), S. 105–119.
- [16] Zadeh, L.; Kacprzyk, J. (Hg.): *Computing with Words in Information/Intelligent Systems, Vol. 1: Foundations, Vol. 2: Applications*. Heidelberg: Physica-Verlag. 2000.
- [17] Mikut, R.; Loose, T.; Jäkel, J.: Rule-Oriented Information Acquisition from Biological Time Series in Clinical Decision Making. In: *Proc. 10th Zittau Fuzzy Colloquium*, S. 300–307. Hochschule Zittau/Görlitz. 2002.

- [18] Perry, J.: *Gait Analysis. Normal and Pathological Function*. Thorofare: Slack Inc. 1992.
- [19] Mikut, R.; Jäkel, J.; Gröll, L.: Automatic Design of Interpretable Membership Functions. In: *Proc. 8th Zittau Fuzzy Colloquium*, S. 103–111. Hochschule Zittau/Görlitz. 2000.
- [20] Jäkel, J.; Gröll, L.: Schätzung der bedingten Wahrscheinlichkeit unscharfer Ereignisse. In: *Proc. 11. Workshop Fuzzy Control des GMA-FA 5.22, Dortmund*, S. 172–188. Forschungszentrum Karlsruhe, FZKA 6660. 2001.
- [21] Bonissone, P. P.; Decker, K. S.: Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-Off Precision and Complexity. In: *Uncertainty in Artificial Intelligence* (Kanal, L. N.; Lemmer, J. F., Hg.), S. 217–247. Amsterdam: North-Holland. 1986.
- [22] Renooij, S.; Witteman, C. L. M.: Talking probabilities: communicating probabilistic information with words and numbers. *International Journal of Approximate Reasoning* 22 (1999), S. 169–194.

Geometrische Veranschaulichung von Fuzzy-Klassifikationssystemen und Multilayer Perceptrons und Parallelen zwischen beiden Modellen

Birka von Schmidt

Institut für Flugführung
Deutsches Zentrum für Luft- und Raumfahrt
Lilienthalplatz 7
D-38108 Braunschweig
Tel.: 0531-295-2225
Fax.: 0531-295-2899
E-mail: birka.vonschmidt@dlr.de

In diesem Beitrag soll ein Überblick darüber gegeben werden, wie sich Fuzzy-Klassifikationssysteme ebenso wie Neuronale Netze geometrisch darstellen lassen, indem man den Raum mittels Hyperebenen in Sektoren einteilt und diese den Klassen zuordnet. Über diese geometrischen Betrachtungen wird es dann möglich, die beiden Systeme miteinander zu verknüpfen.

1 Fuzzy-Klassifikationssysteme

Fuzzy-Klassifikationssysteme unterscheiden sich von den Fuzzy-Reglern dadurch, daß sie keine kontinuierliche Ausgabe aufweisen, sondern die Eingabewerte einer endlichen Anzahl diskreter Werte zuordnen. Dabei repräsentiert jeder Ausgabewert eine Klasse.

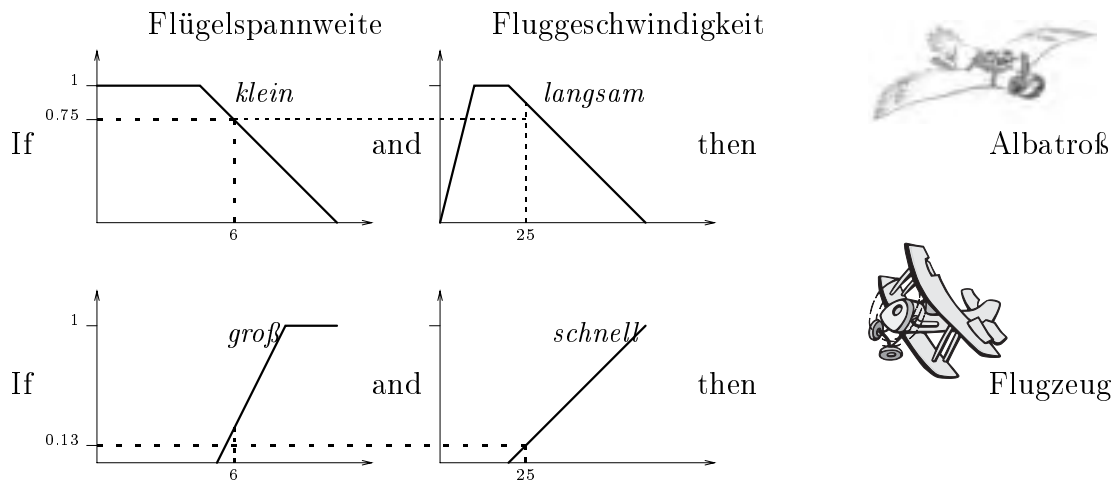
Wir nehmen an, daß das System nur zwischen zwei Klassen \mathcal{C}^+ und \mathcal{C}^- unterscheidet. Eine Generalisierung für beliebige Klassen ist dann leicht zu realisieren.

Ein Fuzzy-Klassifikationssystem setzt sich aus einer endlichen Menge von Regeln der Form

R : If x_1 is $\mu_R^{(1)}$ and ... and x_m is $\mu_R^{(m)}$ then class is \mathcal{C}_R

zusammen, wobei \mathcal{C}_R entweder \mathcal{C}^+ oder \mathcal{C}^- ist und die $\mu_R^{(i)}$ Fuzzy-Mengen jeweils für die i -te Koordinate sind.

Beispiel 1 *Betrachten wir ein Beispiel mit zwei Regeln, die als T-Norm das Minimum nutzen:*



Soll nun z.B. für Orville mit der Eingabe (6,25) die Klasse bestimmt werden. Die erste Regel feuert mit $\min\{0.75, 0.875\} = 0.75$, zur Klasse der Albatrosse gehört Orville nur mit $\min\{0.125, 0.25\} = 0.125$. Orville ist also ein Albatroß (mit Zugehörigkeitsgrad 0.75).

Im vorangegangenen Beispiel wurden für t-Norm und t-Conorm Minimum und Maximum gewählt. Diese verbreitete Methode führt zu einem Fuzzy-Klassifikationssystem, das - anschaulich gesprochen - den n-dimensionalen Datenraum durch (n-1)-dimensionale Hyperebenen in Sektoren einteilt, die dann den Klassen zugeordnet werden. Ein solches Klassifikationssystem entscheidet lokal nur aufgrund von zwei Koordinaten, was anschaulich bedeutet, daß die Hyperebenen jedoch parallel zu n-2 der Koordinaten sind ([3]). Die dadurch mögliche Klassifikation ist eingeschränkt, wenn sie auch dem menschlichen Denken entgegenkommt.

Andere Möglichkeiten für t-Normen und t-Conormen führen zu anderen Klassenabgrenzungen. In [2] sind verschiedene Formen für Klassengrenzen betrachtet worden.

1.1 Fuzzy-Klassifikationssysteme, die die Łukasiewicz-t-Norm nutzen

Die Łukasiewicz-t-Norm ist sehr viel universeller als das Minimum. Sie ist definiert durch

$$T_{Luk}(a_1, \dots, a_n) = \sum_{i=1}^n a_i + 1 - n$$

und mit ihr lassen sich beliebige linear separable Probleme lösen ([1]). Das bedeutet, daß sich der Datenraum durch Hyperebenen in die Klassen einteilen läßt ([4]).

Betrachten wir zunächst den zweidimensionalen Fall. In Abbildung 1 sind die Fuzzy-Mengen für eine lineare Klassifikation aufgezeigt.

Die in Abbildung 2 gezeigte Klassifikation löst das linear separable Problem im zweidimensionalen Fall unter Verwendung der Minimums-Norm. Da jedoch lokal nur zwei Variablen zur Klassifikation genutzt werden, ist in höheren Dimensionen nicht jedes linear separable Klassifikationsproblem lösbar. Die ist im Fall der Łukasiewicz-t-Norm anders. In Abbildung 3 ist gezeigt, daß der Feuerungsgrad einer Regel an einer Geraden mit 0 beginnt und dann ansteigt, bis der Feuerungsgrad 1 erreicht

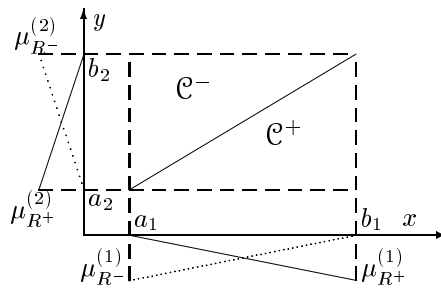


Abbildung 1: Die Fuzzy-Mengen für eine lineare Klassifikationsgrenze

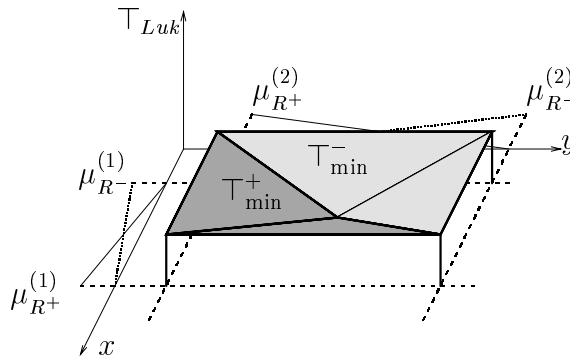


Abbildung 2: Veranschaulichung der t-Norm \top_{\min}

ist. Dies sieht im n -dimensionalen Fall dann so aus, daß an einer Hyperebene der Dimension $n - 1$ der Feuerungsgrad einer Regel 0 ist und von dort aus ansteigt bis zu dem Punkt, an dem alle Fuzzy-Mengen den Zugehörigkeitsgrad 1 haben, und dort mit 1 feuert.

Soll das Fuzzy-Klassifikationssystem nun die Klassengrenzen eines linear separablen Problems darstellen, so teilt man den Raum in Teilquader. Dabei enthält jeder Teilquader der Dimension n maximal n Trennräume der Dimension $n - 1$, die sich in einer Ecke des Quaders treffen. Ein Beispiel für den dreidimensionalen Raum ist in Abbildung 4 zu sehen.

Bemerkung 1 *Der Fall, daß sich mehr als n Trennräume treffen ist ein Sonderfall,*

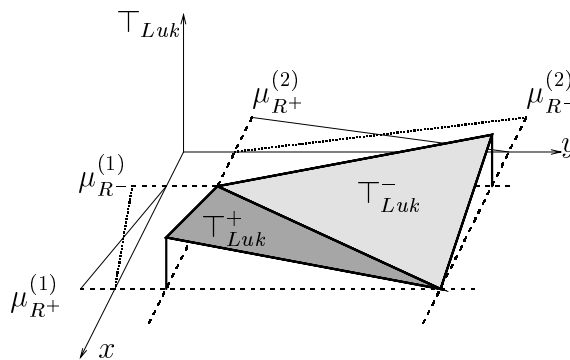


Abbildung 3: Veranschaulichung der t-Norm \top_{Luk}

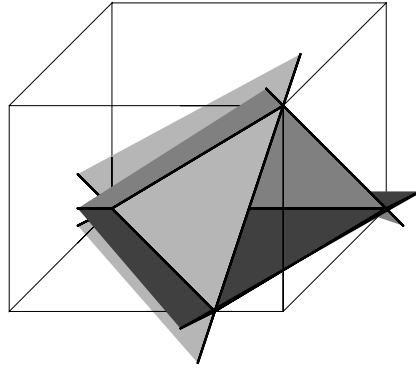


Abbildung 4: Ein Teilquader mit maximaler Ebenenzahl im dreidimensionalen Fall

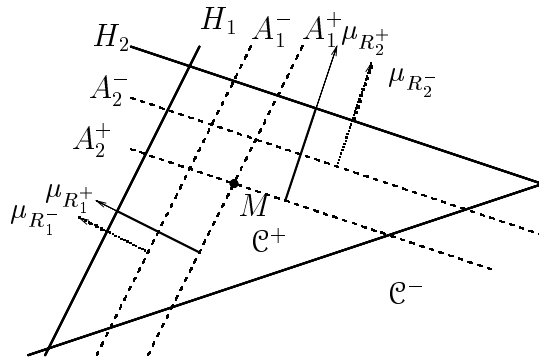


Abbildung 5: Hilfsebenen für die Regelbestimmung

der nur eine beliebig kleine Umgebung beeinflusst. Dabei ist zu beachten, daß die Anzahl der zu betrachtenden Teilquader desto größer wird, je kleiner man den Quader wählt, der den Sonderfall enthält.

Um ein Fuzzy-Klassifikationssystem zu konstruieren, das ein Klassifikationsproblem wie in Abbildung 4 löst, nutzen wir einige Hilfsebenen. Abbildung 5 stellt einen Schnitt durch alle drei Hyperebenen dar. M ist dabei die Gerade, die vom Schnittpunkt der drei Ebenen aus durch den inneren Sektor verläuft, z.B. durch dessen Schwerpunkt.

Zu jeder Hyperebene H_i gehören zwei Hilfsebenen A_i^+ und A_i^- . Mit Hilfe dieser Hilfsebenen werden nun Regeln konstruiert. Eine Regel $\mu_{R_i^+}$ feuert für die Klasse \mathcal{C}^+ innerhalb des Sektors, beginnt bei A_i^+ mit Feuerungsgrad 0 und steigt bis zur Ecke des Quaders an. Die zweite Regel $\mu_{R_i^-}$ beginnt an A_i^- mit Feuerungsgrad 0. Sie wird so konstruiert, daß sie doppelt so schnell ansteigt wie $\mu_{R_i^+}$. Dadurch überholt sie $\mu_{R_i^+}$ genau an H_i . Damit feuert innerhalb des durch die H_i bestimmten Sektors $\mu_{R_i^+}$ für \mathcal{C}^+ und außerhalb des Sektors $\mu_{R_i^-}$ für \mathcal{C}^- . In Abbildung 5 wurden die Hilfsebenen und die Richtungen, in denen die $\mu_{R_i^+}$ bzw. die $\mu_{R_i^-}$ ansteigen, eingezeichnet, der Übersichtlichkeit wegen allerdings nur für H_1 und $H - 2$.

Dieses Verfahren kann für beliebige Dimensionen angewandt werden. Man bestimmt den Sektor und den Schnittpunkt. M verbindet den Schnittpunkt mit dem Schwerpunkt des Sektors. Die Hilfsebenen A_i^+ beinhalten M und sind bezüglich des Schnitt-

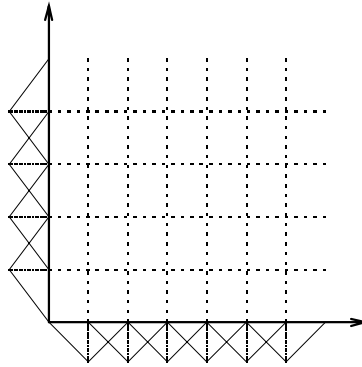


Abbildung 6: Einteilung des Raumes entsprechend der Fuzzy-Regeln

tes senkrecht zu M parallel zu H_i . Die Hilfsebenen A_i^- werden dann mittig zwischen H_i und A_i^+ gelegt. Nach der Konstruktion der Hilfsebenen können dann die Regeln konstruiert werden.

1.2 Fuzzy-Klassifikations Systeme und ihre geometrische Darstellung

Das Verfahren läßt sich auch umdrehen, d.h. man kann sich ein vorhandenes Klassifikationssystem geometrisch veranschaulichen.

Um die Klassifikation eines Fuzzy-Klassifikationssystems zu bestimmen, teilt man den Raum entsprechend der Mittelpunkte der Fuzzy-Mengen. Wenn ein Fuzzy-System nicht so regelmäßig ist wie in Abbildung 6, muß es doch durch Streckung und Stauchung in ein solches zu überführen sein.

Wenn die Regeln genau zur Hälfte überlappen wie in Abbildung 6, so entsteht bereits im dreidimensionalen Raum ein relativ großer nicht-klassifizierter Bereich (Abbildung 7). Dieser wird mit steigender Dimension im Verhältnis immer größer, was nicht der Fall ist, wenn sich die Fuzzy-Mengen weit genug überlappen. Daher macht es nur Sinn, ein System mit entsprechender Überlappung zu untersuchen. Wenn sie sich um $\frac{n-2}{n}$ der Breite des Cuboids ($\delta_c^{(i)}$ für die i -te Koordinate) überlappen, wie in Abbildung 9 gezeigt, so haben wir wieder den in Abbildung 8 gezeigten Fall, daß die Regeln genau im Mittelpunkt des Quaders mit dem Feuerungsgrad 0 beginnen und dann bis zur Ecke linear auf 1 ansteigen.

Die Klassifikation wird für jeden Quader einzeln bestimmt. Dabei lassen wir solche Regeln außer acht, die den Feuerungsgrad 1 in einem anderen Quader erreichen, also noch mit geringen Feuerungsgraden in den gerade untersuchten Quader ragen. Wir gehen also davon aus, daß die Fuzzy-Mengen jeweils nur in zwei benachbarten Quadern feuern.

Bemerkung 2 Diese Annahme macht insofern Sinn, als daß wir ein Klassifikationsproblem betrachten. Im Falle eines Fuzzy-Reglers ist das Ziel dagegen eine interpolierende Funktion, zu der die kleinen Anteile der Fuzzy-Mengen in glättender Funktion beitragen.

Hier betrachten wir jedoch ein Klassifikationsproblem. Entweder es fließen aus anderen Regeln große Zugehörigkeitsgrade in die Berechnungen ein, so daß die kleinen

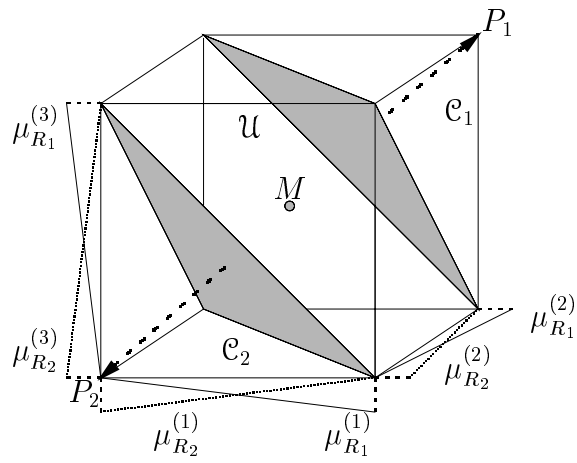


Abbildung 7: Der mittlere Bereich U bleibt unklassifiziert.

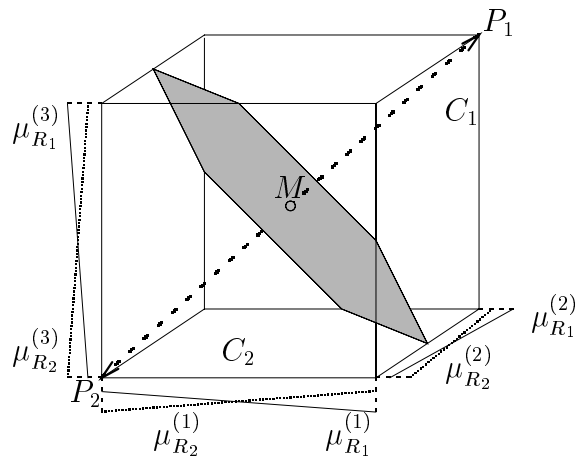


Abbildung 8: Bei entsprechender Überlappung der Fuzzy-Mengen wird der ganze Quader klassifiziert

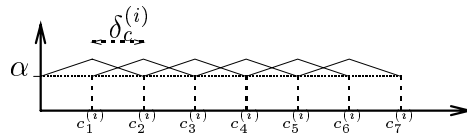


Abbildung 9: So sollten die Fuzzy-Mengen aussehen.

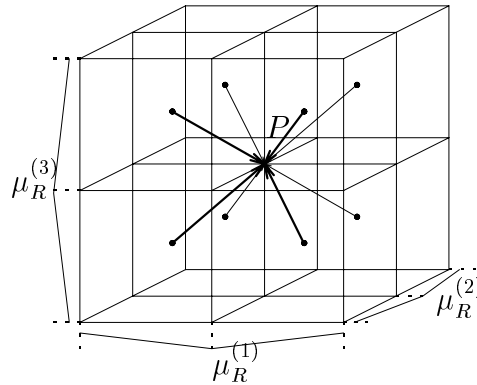


Abbildung 10: Aus $2^n = 8$ Richtungen steigt der Feuerungsgrad in Richtung der Ecke an

Zugehörigkeitsgrade das Ergebnis nicht ändern und bedenkenlos vernachlässigt werden können, oder sie bestimmen das Resultat als einzig vorhandene. Damit wäre dann das Gesamtergebnis klein, so daß die resultierende Klassifikation nicht deutlich differenzierte.

Setzt man nun mehrere dieser einzeln betrachteten Teilquader zusammen, so gehen auf eine Ecke 2^n solche Teilregeln zu wie in Abbildung 10 zu sehen, die alle für die gleiche Klasse feuern, so daß die Ecke eindeutig einer Klasse zugeordnet werden kann.

Die in benachbarten Quadern entstehenden Trennebenen ergänzen sich teilweise, so daß sie zur Vereinfachung zusammengefaßt werden können. In Abbildung 11 ist ein Beispiel für eine solche Fortsetzung zu sehen.

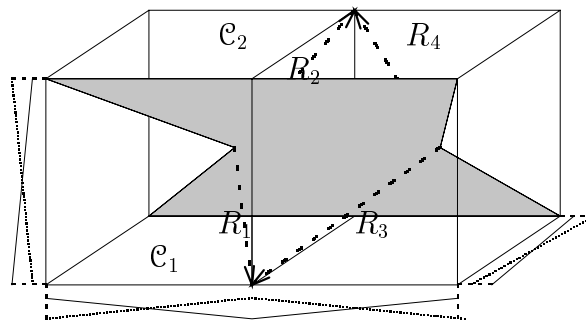
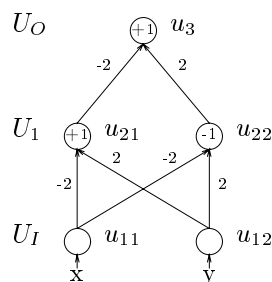


Abbildung 11: Trennungen lassen sich im Nachbarquader weiterführen

2 Multilayer Perceptrons

Ebenso wie ein Łukasiewicz Fuzzy-Klassifikationssystem kann ein Multilayer Perceptron (MLP) ein stückweise linear separables Problem lösen. Um die Funktionsweise eines MLP kurz zu erläutern, nutzen wir das klassische Beispiel des XOR-Problems:

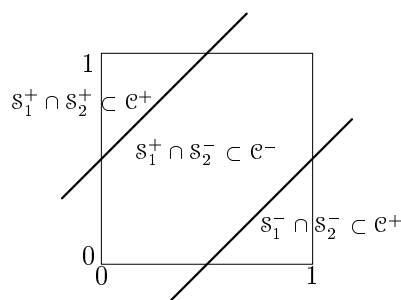
Beispiel 2 (Eine Lösung des XOR-Problems) Die folgende Graphik repräsentiert ein Multilayer Perceptron. An den Pfeilen sind die Gewichte der entsprechenden Verbindungen vermerkt, in den Kreisen die Schwellwerte der entsprechenden Neuronen.



Die Aktivierungen der Neuronen werden dann folgendermaßen berechnet:

$$\begin{aligned}
 a_{u_{11}} &= x \text{ and } a_{u_{12}} = y; & W_{I1} &= \begin{pmatrix} -2 & 2 \\ -2 & 2 \end{pmatrix}; \\
 a_{u_{21}} &= -2 \cdot a_{11} + 2 \cdot a_{12} + 1; & & \\
 a_{u_{22}} &= -2 \cdot a_{11} + 2 \cdot a_{12} - 1; & W_{1O} &= \begin{pmatrix} -2 \\ 2 \end{pmatrix}. \\
 a_{u_3} &= -2 \cdot a_{21} + 2 \cdot a_{22} + 1; & &
 \end{aligned}$$

Der Raum $[0, 1] \times [0, 1]$ wird durch dieses MLP folgendermaßen in drei Sektoren eingeteilt:



Dieses Beispiel zeigt eine von verschiedenen möglichen Lösungen für das XOR-Problem. □

Hier sind x und y die Eingabewerte. Die Aktivierung a_v der Neuronen in den anderen Schichten wird aus den Aktivierungen a_{u_i} der m Neuronen der jeweils vorhergehenden Schicht berechnet durch

$$a_v = f\left(\sum_{i=1}^m a_{u_i} - \Theta_v\right),$$

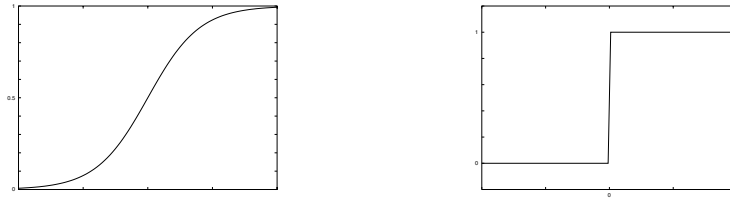


Abbildung 12: Sigmoide und Schwellwert-Aktivierungsfunktion

wobei Θ_v Schwellwert genannt wird und f i.a. eine sigmoide Funktion ist wie in Abbildung 12 links gezeigt. Ein Beispiel dafür wäre die logistische Funktion $f(x) = \frac{1}{1+e^{-\beta x}}$, $\beta > 0$, die sich asymptotisch an 0 und 1 nähert und den Wert 0.5 bei 0 erreicht.

Da die Netzeingabefunktion eines Neurons durch eine Summe bestimmt wird, liegt auch beim Multilayer Perceptron eine Einteilung des Raumes in Sektoren mittels Hyperebenen vor ([5]). Es kann also ebenso wie das Fuzzy-Klassifikationssystem mit Łukasiewicz-t-Norm linear separable Probleme lösen.

Nun kann die Ausgabe interpretiert werden als Zugehörigkeit zu einer Klasse (z.B. Ausgabe 0 entspricht der einen, Ausgabe 1 der anderen Klasse). Je näher die Ausgabe an einen der Werte herankommt, desto klarer gehört der Eingabepunkt zu der entsprechenden Klasse.

Die Steigung der Aktivierungsfunktion lässt sich verändern. Je steiler sie ist, desto näher kommen wir einer Schwellwertfunktion wie in Abbildung 12 rechts, die dann zu einer scharfen Klassifikation führt.

Dieses Verhalten zeigt ganz deutlich die Parallelen zum Fuzzy-System, denn auch dort führt ein Punkt sowohl zu einer Ausgabe für die eine als auch für die andere Klasse. Je weiter ein Punkt von den Klassengrenzen entfernt ist, desto größer ist der Unterschied zwischen den Ausgaben für die verschiedenen Klassen und desto klarer ist somit die Klassifikation.

Nun erhalten wir die Ausgabe 0 bzw. 1 allerdings nicht nur in der Ausgabeschicht, sondern auch in allen inneren Schichten. Das Neuronale Netz lässt sich daher als Boolescher Ausdruck interpretieren. Ein solcher Ausdruck entsteht im Beispiel 2.

Die Neuronen in der inneren Schicht repräsentieren die Trenngeraden im zweidimensionalen Raum wie gezeigt, u_{21} die Gerade links oben, u_{22} die rechts unten. Je nachdem, ob die Aktivierung größer oder kleiner als 0 ist, befindet sich der Eingabepunkt oberhalb oder unterhalb der Geraden. Der Übergang von der inneren Schicht zur Ausgabeschicht ordnet die entstandenen Sektoren den Klassen zu.

In einem komplizierteren Fall wie in Abbildung 13 sind zwei innere Schichten notwendig. Wir wollen den Aufbau eines solchen Multilayer Perceptrons anhand von Abbildung 14 erläutern:

Die Eingabeschicht erhält als Aktivierung die Koordinaten des Eingabepunktes X . Jedes Neuron der ersten inneren Schicht repräsentiert eine Hyperebene bzw. im zweidimensionalen Fall eine Gerade, die durch die Summenformel gegeben ist. Das Neuron erhält die Aktivierung 0 oder 1 wenn der Punkt sich unterhalb oder oberhalb der Ebene befindet.

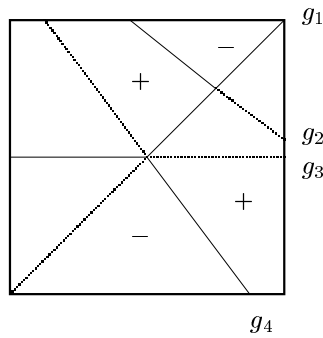


Abbildung 13: Ein Beispiel für eine Klassifikation im zweidimensionalen Raum.

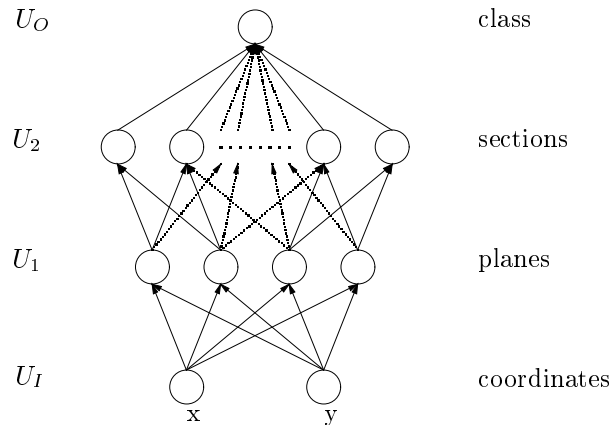


Abbildung 14: Ein Beispiel für ein MLP, das das Klassifikationsproblem aus Abbildung 13 löst.

Der Übergang von der ersten zur zweiten inneren Schicht repräsentiert ein AND, und jedes Neuron der zweiten inneren Schicht steht für einen Sektor.

IF X unterhalb von g_3 AND oberhalb von g_4 , THEN X in Sektor S_1 .

Der Übergang zur Ausgabeschicht schließlich repräsentiert ein OR.

IF X in Sektor S_1 OR X in Sektor S_2 , THEN X Klasse \mathcal{C}^+ .

Wenn mehr als zwei Klassen vorhanden sind, so bekommt jede Klasse ein Ausgabeneuron zugewiesen, das aktiviert wird, wenn X Element der Klasse ist. Bei zwei Klassen ist ein Neuron ausreichend.

Nehmen wir an, daß tatsächlich eine Schwellwertfunktion als Aktivierungsfunktion dient. Betrachtet man diese AND-OR-Verknüpfung, so fällt auf, daß die durch das MLP repräsentierte Boolesche Funktion in Form einer Disjunktiven Normalform vorliegt. Da ein jedes MLP mit Schwellwertfunktion einen Booleschen Ausdruck repräsentiert, läßt es sich auch in ein Netz mit zwei inneren Schichten transformieren, das identische Ausgaben bei identischen Eingaben liefert.

Es ist ebenso möglich, eine Konjunktive Normalform zu erzeugen, doch entfällt dann die anschauliche Interpretation durch Ebenen und Sektoren.

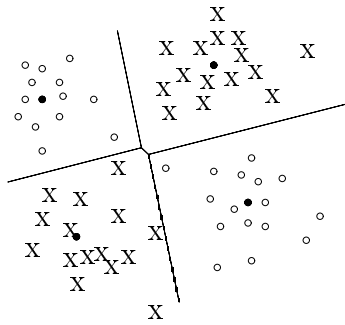


Abbildung 15: Ein Beispiel für Clustering Prototypen und die daraus entstehenden Klassengrenzen

3 Zusammenfassung und Ergänzungen

Da sowohl MLPs als auch Fuzzy-Klassifikationssysteme den Raum mittels Hyperebenen in Sektoren einteilen, die dann wiederum den Klassen zugeordnet werden, bildet diese geometrische Veranschaulichung eine Verbindung zwischen beiden Systemen.

Es ist möglich, ein bestehendes Neuronales Netz in ein Fuzzy-System mit gleichen Ergebnissen zu transformieren. Das Ergebnis sind interpretierbare Fuzzy-Regeln, die es möglich machen, das Black-Box-Verhalten des MLPs aufzulösen.

Umgekehrt kann Expertenwissen genutzt werden, um ein Fuzzy-System zu konstruieren. Konstruiert man nun aus diesem Fuzzy-System ein MLP, so kann die entstandene Initialisierung des Neuronalen Netzes durch einen Lernvorgang weiter verbessert werden. Das MLP kommt viel schneller zu einem guten Ergebnis als es das ohne gute Initialisierung gekonnt hätte.

Eine weitere Möglichkeit, diese Erkenntnisse zu nutzen bringt uns zum Fuzzy Clustering. Auch aus dem Ergebnis eines Fuzzy-c-means Algorithmus lassen sich Trennebenen (bzw. Trennräume der Dimension $n - 1$) extrahieren, indem man in der Mitte zwischen jeweils zwei Prototypen eine Trennebene zieht, die genau senkrecht zu ihrer Verbindungsline verläuft (siehe Abbildung 15). Ein Punkt, der sich auf der einen Seite dieser Trennung befindet, hat einen geringen Abstand zu dem entsprechenden Prototypen und hat somit auch eine größere Zugehörigkeit zu diesem als zu dem jeweils anderen.

Auf diese Weise entstehen Sektoren, die wiederum den Klassen zugeordnet werden können, zu denen auch die entsprechenden Prototypen gehören.

Das sonst oft genutzte Verfahren, mittels Projektion Fuzzy-Regeln aus den Clustern zu formen, die die Cluster abbilden, schafft kein identisches Abbild; es geht Information verloren. Die Möglichkeit mittels Sektoren die Regeln zu konstruieren, erhält alle in den Clustern gegebenen Informationen.

Insgesamt wird deutlich, daß die Geometrische Veranschaulichung ein Bindeglied zwischen den einzelnen Verfahren ist, das es möglich macht, die Vorteile der unterschiedlichen Methoden zu kombinieren.

Literatur

- [1] F. Klawonn, E.P. Klement: Mathematical analysis of fuzzy classifiers. In: X. Liu, P. Cohern, M. Berthold (eds.: Advances in intelligent data analysis. Springer, Berlin (1997), 359-370
- [2] A.Nürnbergger, A.Klose, and R.Kruse: "Discussing Cluster Shapes of Fuzzy Classifiers"; In: Proc. International Conference of the North American Fuzzy Information Processing Society '99 (NAFIPS'99), pp.546-550, New York, 1999.
- [3] B. von Schmidt, F. Klawonn: Fuzzy max-min classifiers decide locally on the basis of two attributes. *Mathware and Soft Computing* 6 (1999), 91-108
- [4] B. von Schmidt, F. Klawonn: Construction of fuzzy classification systems with the Łukasiewicz-t-norm. Proc. 19th Conf. of the North American Fuzzy Information Processing Society (NAFIPS'00), Atlanta (2000), 109-113
- [5] B. von Schmidt, F. Klawonn: Construction of a multilayer perceptron for a piecewise linearly separable classification problem. Proc. Joint 9th International Fuzzy Systems Association World Congress and 20th North American Fuzzy Information Processing Society International Conference, Vancouver (2001)

Zur Problematik der Gewichtsverteilung in Künstlichen Neuronalen Netzen

Kästner, W.; Traichel, A.; Förster, T.; Hampel, R.

Institut für Prozeßtechnik, Prozeßautomatisierung und Meßtechnik (IPM)
FG Meßtechnik/ Prozeßautomatisierung
an der Hochschule Zittau/Görlitz (FH)
D-02763 Zittau, Theodor-Körner-Allee 16
Tel: +49-(0)3583-61-1547, Fax: +49-(0)3583-61-1288
E-mail: ipm@hs-zigr.de

Kurzfassung

Künstliche Neuronale Netze (KNN), insbesondere Multilayer Perzeptren (MLP), werden im Rahmen der Modellbildung für die Nachbildung nichtlinearer Zusammenhänge zwischen Ein- und Ausgangsgrößen erfolgreich eingesetzt. Dabei wird das KNN im Allgemeinen als Black-Box-Modell interpretiert. Als Kriterien für die Bewertung der Güte des KNN dienen Fehlerkenngrößen des Trainings- bzw. Testfehlers.

Motivation für die im Beitrag beschriebenen Untersuchungen ist der Aspekt, daß das MLP die globalen Prozeßzusammenhänge in einer sich ausbildenden Gewichtstopologie speichert. Ausgangspunkt der Untersuchungen ist die Fragestellung, ob sich zwischen dem nachzubildenden Prozeßzusammenhang und der sich einstellenden Gewichtsverteilung des MLP Relationen ableiten lassen.

Für die Untersuchungen wurden bekannte analytische Zusammenhänge definiert und mittels eines MLP einfacher Topologie nachgebildet. Die sich einstellende Gewichtsverteilung wurde analysiert und in Beziehung zu den definierten Prozeßzusammenhängen gesetzt. Im Beitrag wird auf die Analyse ausgewählter Beispiele und deren Ergebnisse eingegangen und der mögliche Nutzen dieser Ergebnisse als ein Kriterium für die Bewertung der Netzgüte zur Diskussion gestellt.

1 Motivation

Künstliche Neuronale Netze (KNN), insbesondere Multilayer Perzeptren (MLP), werden im Rahmen der Modellbildung für die Nachbildung nichtlinearer Prozeßzusammenhänge zwischen Ein- und Ausgangsgrößen erfolgreich eingesetzt. Dies gilt vor allem für Prozesse, die mit analytischen Modellen nur unzureichend oder gar nicht beschrieben werden können. Dabei steht die möglichst exakte Beschreibung des globalen Zusammenhanges zwischen den Eingangsgrößen (X_i) und den Ausgangsgrößen (Y_j) des Prozesses im Vordergrund. Die Kenntnis der internen Parameter und Zustandsgrößen ist für die Modellierung mittels KNN nicht unbedingt erforderlich. Deshalb werden KNN im Allgemeinen als Black-Box-Modell interpretiert [2, 3].

Als Kriterien für die Bewertung der Güte der KNN dienen Fehlerkenngrößen des Trainings- bzw. Testfehlers. Dabei wird bewertet, mit welcher Genauigkeit das KNN die Trainings- bzw. Testdatensätze widerspiegeln kann (Bild 1).

Motivation für die im Beitrag beschriebenen Untersuchungen ist der Aspekt, daß das MLP die globalen Prozeßzusammenhänge in einer sich ausbildenden Gewichtstopologie speichert, welche sich im Verlaufe des Trainings einstellt.

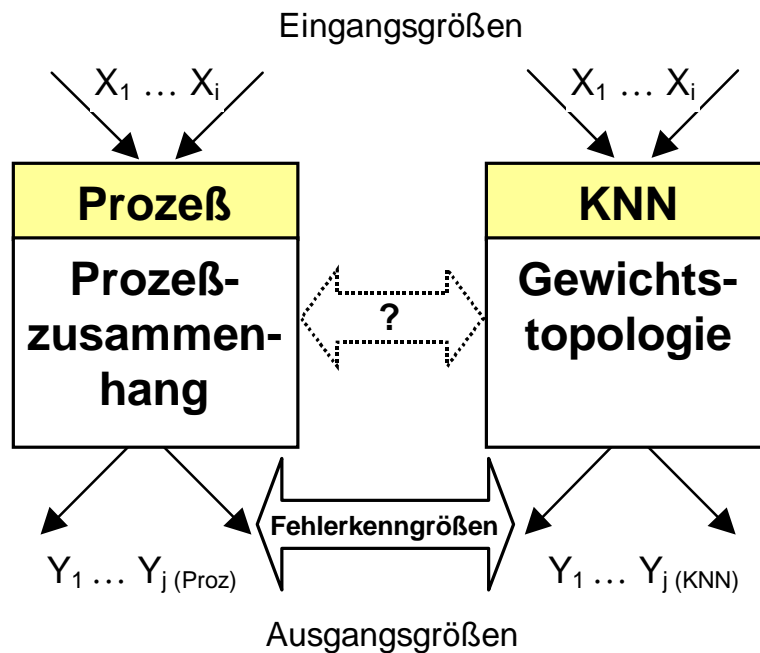


Bild 1: Nachbildung des Prozeßzusammenhanges mittels MLP

Die Gewichtstopologie war bei den bisherigen methodischen Betrachtungen zu KNN ein Untersuchungsgegenstand unter folgendem Blickwinkel:

- ⇒ bei der Gewichtsinitialisierung zu Beginn des Trainingsprozesses (symmetry breaking problem),
- ⇒ in Verbindung mit dem Lernverfahren (Backpropagation), wenn die Gewichte im Zuge des Trainingsprozesses adaptiert werden,
- ⇒ bei der Vergleichmäßigung der Beträge der Einzelgewichte (weight decay),
- ⇒ bei der Bewertung der Relevanz von Einzelgewichten (pruning).

Die Frage nach einem Zusammenhang zwischen einer sich ausbildenden Einstellung der Gewichte und dem trainierten Prozeß stand bisher nicht im Blickpunkt der Untersuchungen. Gerade für sicherheitsrelevante Anwendungen gewinnt dieser Aspekt an Bedeutung, da in diesem Sinne mehr Transparenz des Black-Box-Modells als Voraussetzung für die Evaluierung der Netzgüte gefordert wird [4].

Daraus leitet sich folgende Fragestellung ab:

Lassen sich zwischen dem nachzubildenden Prozeßzusammenhang und der sich einstellenden Gewichtsverteilung des MLP Relationen ableiten?

Die Anregung zu diesen Untersuchungen geht auf [1] zurück.

2 Generierung von Bewertungskenngrößen für die Gewichtsverteilung

2.1 Analyse der Einzelgewichte

Die Gewichtstopologie eines MLP setzt sich aus der Struktur der Einzelgewichte entsprechend der Netzstruktur, dem Betrag und dem Vorzeichen der Einzelgewichte zusammen. Die Einzelgewichte zwischen den Schichten des MLP können in sogenannten Gewichtsmatrizen zusammengefaßt werden. Zunächst wird untersucht, ob aus den Einzelgewichten Aussagen bezüglich des Prozeßzusammenhanges abgeleitet werden können.

Bei der Analyse des Trainingsprozesses zeigt sich, daß infolge der Gewichtsinitialisierung mit Zufallswerten unterschiedliche endgültige Gewichtsverteilungen generiert werden können. Die Unterschiede in den Einzelgewichten betreffen sowohl den Betrag als auch das Vorzeichen. Durch die daraus resultierenden unterschiedlichen Netze wird der globale Zusammenhang zwischen Ein- und Ausgangsgrößen in vergleichbarer Güte bezüglich des Trainings- bzw. Testfehlers reproduziert. Bild 2 veranschaulicht diesen Effekt für ein einfaches MLP mit acht Einzelgewichten anhand von Patch-Diagrammen.

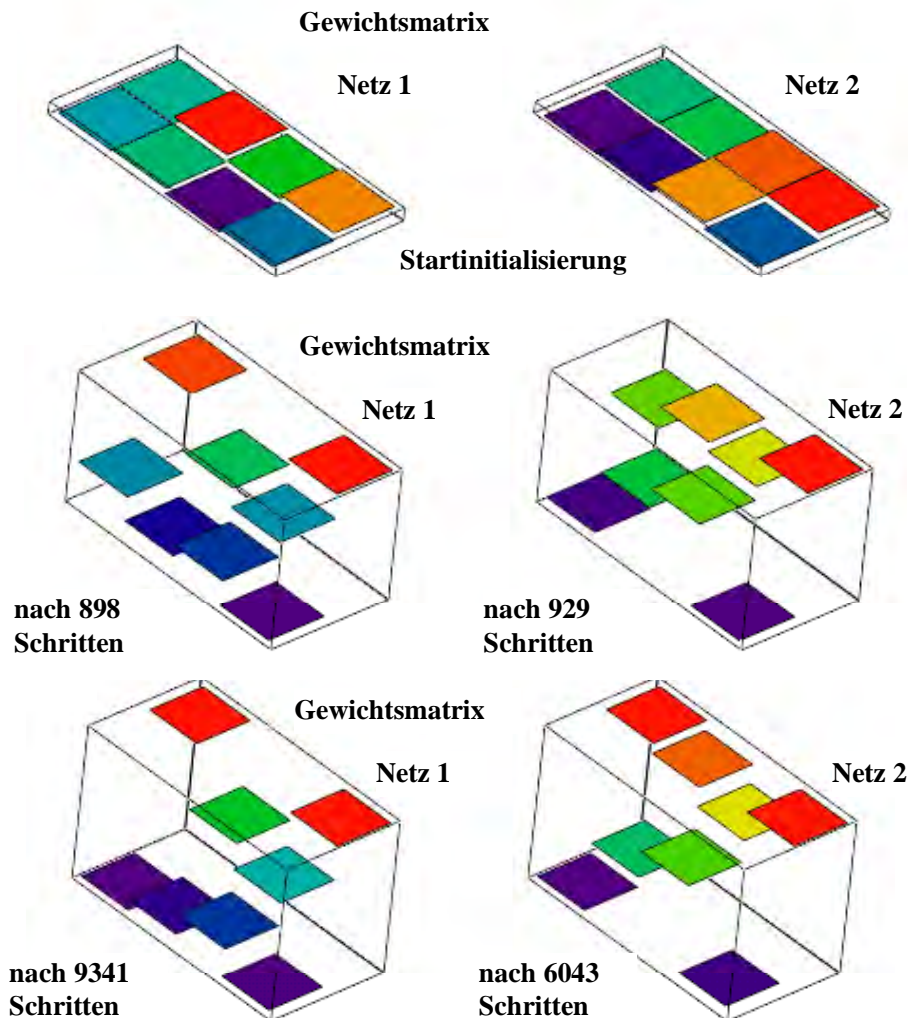


Bild 2: Trainingsergebnisse zwei gleicher Netze bei zufälliger Gewichtsinitialisierung (Darstellung der Gewichtsmatrix der acht Einzelgewichte eines MLP)

Im linken und rechten Teil des Bildes 2 wird anhand von drei Etappen der Trainingsprozeß zweier Netze dargestellt, die auf der gleichen Netzstruktur beruhen und für ein und dieselbe Trainingsaufgabe zur Anwendung kommen. Die zwei Netze unterscheiden sich lediglich in der Gewichtsinitialisierung (oberer Teil des Bildes), d. h. in Betrag und Vorzeichen der Einzelgewichte zu Beginn des Trainingsprozesses, veranschaulicht durch die unterschiedlichen Farben der Einzelgewichte.

Der mittlere und der untere Teil des Bildes zeigen die anhand des Trainingsfehlers als optimal bewerteten Netzzustände nach einer entsprechenden Anzahl von Epochen. Bei Erreichen eines definierten Fehlerwertes wurde das Training abgebrochen. Die unterschiedlichen per Zufall generierten Startinitialisierungen der Gewichtsmatrizen führen infolge des Trainingsprozesses zur Ausbildung unterschiedlicher Gewichtsverteilungen.

Als Ergebnis der Vorbetrachtungen läßt sich ableiten, daß die Einzelgewichte an sich für die Analyse des gesuchten Zusammenhanges ungeeignet erscheinen. Als Schlußfolgerung daraus sind aus den Einzelgewichten Kenngrößen zu generieren, die die gewünschten Relationen zur Bewertung des MLP liefern.

2.2 Definition der entkoppelten Korrelationsgewichte als Bewertungskenngrößen

Als Bewertungskenngrößen für die Gewichtsverteilung werden sogenannte entkoppelte Korrelationsgewichte eingeführt. Ihre Herleitung wird in diesem Abschnitt beschrieben und basiert auf folgenden Vereinfachungen [5]:

- ⇒ Entkopplung der Netzstruktur,
- ⇒ Linearisierung der Transferfunktion.

Entkopplung der Netzstruktur

Im Vordergrund steht das Ziel, den zwischen Eingangsgrößen X_i und Ausgangsgrößen Y_j existierenden Zusammenhang nachzubilden. Aus diesem Grund wird eine Entkopplung der Netzstruktur bezüglich der Eingangsgrößen vorgenommen, d. h. es werden den einzelnen Eingangsgrößen zugeordnete Teilnetze gebildet. Diese Vereinfachung hat zur Folge, daß die Ausgangsgröße nur noch in Abhängigkeit der jeweiligen Eingangsgröße betrachtet wird und der Einfluß der anderen Eingangsgrößen (Vernetzung) unberücksichtigt bleibt.

Bild 3 demonstriert die Entkopplung eines einfachen MLP mit 3 Eingangsgrößen und einer Ausgangsgröße sowie zwei Neuronen im Hiddenlayer in Teilnetze ($i = 3$), die den Zusammenhang der jeweiligen Eingangsgröße X_i zur Ausgangsgröße Y repräsentieren.

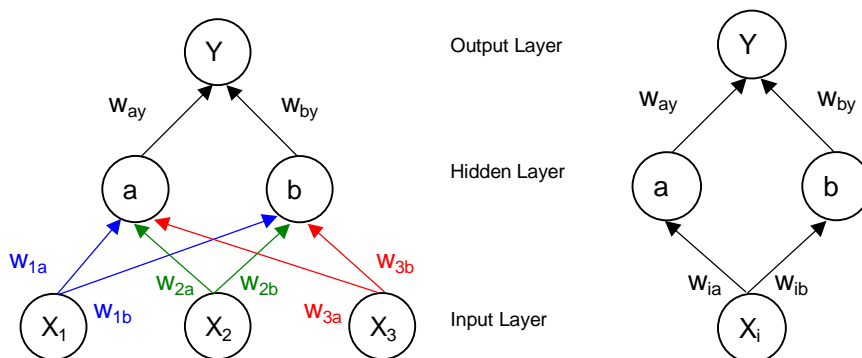


Bild 3: Zerlegung der Netzstruktur in Teilnetze, die den jeweiligen Eingangsgrößen zugeordnet sind

Die Gewichte w_{ia} und w_{ib} bewerten für dieses Beispiel die Verbindung zwischen Inputlayer und Hiddenlayer (bestehend aus zwei Neuronen) und die Gewichte w_{ay} und w_{by} die Verbindung zwischen Hiddenlayer und Outputlayer (bestehend aus einem Neuron).

Linearisierung der Transferfunktion

Um zu möglichst einfachen mathematischen Zusammenhängen zu gelangen, wird als zweite Vereinfachung quasi eine Linearisierung der Transferfunktion vorgenommen. Dazu wird die Transferfunktion durch eine Reihenentwicklung ersetzt, wobei nur die erste Näherung der Reihe Berücksichtigung findet.

Für das Beispiel der Transferfunktion $\tanh(x)$ gilt (1):

$$\tanh(z) \approx z \quad (1)$$

Mit diesen Vereinfachungen können aus Bild 3 lineare Zusammenhänge entsprechend Gleichung (2) abgeleitet werden, die die Abhängigkeit der Ausgangsgröße Y von der jeweiligen Eingangsgröße X_i entsprechend (3) charakterisieren.

$$G_i = (w_{i,a} \cdot w_{a,y} + w_{i,b} \cdot w_{b,y}) \quad (2)$$

$$Y(X_i) = G_i \cdot X_i \quad (3)$$

Die Größen G_i werden als entkoppelte Korrelationsgewichte bezeichnet. Sie kennzeichnen die Intensität (den Anstieg) und die Richtung (das Monotonieverhalten) des linearen Zusammenhanges zwischen der Eingangsgröße und der Ausgangsgröße.

Die entkoppelten Korrelationsgewichte G_i dienen bei den nachfolgenden Untersuchungen als Bewertungskenngröße.

3 Sensibilitätsanalyse anhand spezieller analytischer Beispielfunktionen

3.1 Ausgangspunkt der Sensibilitätsanalyse

Für die Sensibilitätsanalyse wurden funktionale Zusammenhänge definiert und mittels eines MLP einfacher Architektur reproduziert. Die sich einstellende Gewichtsverteilung wurde anhand der entkoppelten Korrelationsgewichte analysiert und in Beziehung zu den definierten Prozeßzusammenhängen gesetzt. Als Prozeßbeispiele dienten funktionale Zusammenhänge f zwischen den drei Eingangsgrößen X_1 , X_2 und X_3 und der Ausgangsgröße Y der Form

$$Y = f(m_i \cdot X_i^{a_i}) \quad (4)$$

wobei m den Anstieg und a den Exponenten der Funktion charakterisiert.

Dabei wurden zwei Formen der Nichtlinearität gewählt:

- ⇒ nichtlineare Funktionen für die Abhängigkeit der Ausgangsgröße Y von den Eingangsgrößen X_i , ($a \neq 1$)
- ⇒ nichtlineare Verknüpfungen von Eingangsgrößen.

Weiterhin wurde das Vorzeichen des Anstieges m variiert.

Für den Trainingsprozeß wurde ein MLP mit einer Hiddenlayer-Schicht verwendet, die in der Minimalvariante aus zwei Neuronen bestand. Zum Einsatz kamen:

- ⇒ die Transferfunktion *tanh*,
- ⇒ die Lernregel *Backpropagation*,
- ⇒ die Gewichtsinitialisierung per Zufallsgenerator mit $N(\mu, \sigma) = (-0,1; 0,1)$.

Als Trainingsdatensatz diente ein Datensatz bestehend aus den 3^3 Kombinationsvarianten des Zahlentripels [1,1; 1,5; 1,9] für die Eingangsgrößen X_1, X_2, X_3 und den sich aus dem funktionalen Zusammenhang ergebenden Werten für die Ausgangsgröße Y . Der Trainingsprozeß wurde als abgeschlossen betrachtet, wenn ein vorgegebener Wert des Trainingsfehlers unterschritten wurde. Die Simulation erfolgte mittels des Tools DataEngine.

3.2 Auswertung der Sensibilitätsanalyse

Die vergleichende Betrachtung zwischen dem Prozeßzusammenhang, charakterisiert durch die analytischen Funktionen und den sich ausbildenden entkoppelten Korrelationsgewichten erfolgte vorerst für die Richtung des linearen Zusammenhanges zwischen der Eingangsgröße und der Ausgangsgröße (Monotonieverhalten).

$$Y'(X) > 0 \quad \text{monoton wachsend} \quad (5a)$$

$$Y'(X) < 0 \quad \text{monoton fallend} \quad (5b)$$

Analysiert man das Monotonieverhalten der betrachteten analytischen Zusammenhänge entsprechend Gleichung (4), ist hinsichtlich des Einflusses des Exponenten a und des Anstieges m zu unterscheiden.

Einfluß des Exponenten a :

Der Exponent beeinflusst das Monotonieverhalten (monoton wachsend, monoton fallend) durch sein Vorzeichen (PV), das die direkte Proportionalität (Vorzeichen (+)) und die indirekte Proportionalität (Vorzeichen (-)) zwischen Eingangs- und Ausgangsgröße charakterisiert.

$$Y(X) = X^a \quad Y'(X) = a \cdot X^{a-1} \quad (6)$$

Einfluß des Anstieges m :

Der Anstieg m beeinflusst das Monotonieverhalten (monoton wachsend, monoton fallend) ebenfalls durch sein Vorzeichen (Vorzeichen des Terms TV).

$$Y(X) = m \cdot X \quad Y'(X) = m \quad (7)$$

Die Verknüpfung der Einflüsse von PV und TV wird durch deren Produkt entsprechend Gleichung (8) beschrieben.

$$Y(X) = m \cdot X^a \quad Y'(X) = (m \cdot a) \cdot X^{a-1} \quad (8)$$

Wenn sich die grundlegende Richtung des linearen Zusammenhanges zwischen Ein- und Ausgangsgrößen des Prozesses in der Gewichtstopologie, d. h. in den entkoppelten Korrelationsgewichten G_i abbildet müssen deren Vorzeichen das grundlegende Monotonieverhalten widerspiegeln.

3.3 Ergebnisse der Sensibilitätsanalyse

Tabelle 1 listet ausgewählte analytische Funktionen entsprechend [5] auf und stellt den funktionalen Zusammenhang des Prozesses in Beziehung zu den antrainierten entkoppelten Korrelationsgewichten des MLP. Ausgewertet werden hierbei nur die Vorzeichen (d. h. das Monotonieverhalten, die Richtung des linearen Zusammenhanges), nicht die Beträge der Gewichte (die Intensität des linearen Zusammenhanges).

Die Spalten 2 bis 6 charakterisieren die vorgegebenen funktionalen Zusammenhänge und die Spalten 7 und 8 enthalten Angaben zu den mit Hilfe des MLP antrainierten entkoppelten Korrelationsgewichten G_i und deren Vorzeichen GV.

Sp 1	Sp 2	Sp 3	Sp 4	Sp 5	Sp 6	Sp 7	Sp 8
Fall	Gleichung	Proportionalität	PV	TV	$\Pi(PV,TV)$	GV	G_i
1	$Y = -X_1 - X_2 - X_3$	$Y \sim X_1$ $Y \sim X_2$ $Y \sim X_3$	+	-	-	-	$G_1 = -2,13$ $G_2 = -2,13$ $G_3 = -2,13$
2	$Y = -\frac{1}{X_1} - \frac{1}{X_2} - \frac{1}{X_3}$	$Y \sim 1/X_1$ $Y \sim 1/X_2$ $Y \sim 1/X_3$	-	-	+	+	$G_1 = +1,37$ $G_2 = +1,37$ $G_3 = +1,33$
3	$Y = X_1^2 + X_2 + \frac{1}{X_3}$	$Y \sim X_1$ $Y \sim X_2$ $Y \sim 1/X_3$	+	+	+	+	$G_1 = +4,0$ $G_2 = +1,6$ $G_3 = -0,9$
4	$Y = \frac{1}{X_1^2} + \frac{1}{X_2} + X_3$	$Y \sim 1/X_1$ $Y \sim 1/X_2$ $Y \sim X_3$	-	+	-	-	$G_1 = -2,9$ $G_2 = -1,9$ $G_3 = +2,7$
5	$Y = -\frac{X_1}{X_2} + X_3$	$Y \sim X_1$ $Y \sim 1/X_2$ $Y \sim X_3$	+	-	-	+	$G_1 = -1,5$ $G_2 = +1,5$ $G_3 = +2,4$
6	$Y = \frac{X_1}{X_2} - X_3$	$Y \sim X_1$ $Y \sim 1/X_2$ $Y \sim X_3$	+	+	+	+	$G_1 = +2,1$ $G_2 = -2,4$ $G_3 = -2,2$

Tabelle 1: Entkoppelte Korrelationsgewichte G_i des MLP für ausgewählte analytische Funktionen und Zuordnung der funktionalen Zusammenhänge

Legende:

- PV - Vorzeichen des Exponenten der Zustandsgröße X_i (kennzeichnet die direkte bzw. indirekte Proportionalität zwischen Ausgangsgröße Y und Eingangsgröße X_i)
- TV - Vorzeichen des Terms (des Anstieges) des funktionalen Zusammenhanges für Zustandsgröße X_i
- $\Pi(PV,TV)$ - Vorzeichen resultierend aus dem Produkt von PV und TV für Zustandsgröße X_i
- GV - Vorzeichen des entkoppelten Korrelationsgewichtes der Zustandsgröße X_i , generiert mit MLP
- G_i - Entkoppeltes Korrelationsgewicht der Zustandsgröße X_i , generiert mit MLP

Spalte 2 der Tabelle 1 enthält die vorgegebenen funktionalen Zusammenhänge zwischen den Eingangsgrößen X_i und der Ausgangsgröße Y .

Für die einzelnen Zustandsgrößen wird in Spalte 3 dargestellt, ob die Ausgangsgröße Y direkt oder indirekt proportional von der jeweiligen Eingangsgröße X_i abhängt.

Für die einzelnen Eingangsgrößen wurde in Spalte 4 anhand der Vorzeichen der Exponenten ausgewiesen, wie sie das Monotonieverhalten der analytischen Funktion beeinflussen.

Spalte 5 enthält das Vorzeichen des Terms (TV), mit dem der Einfluß des Anstieges auf das Monotonieverhalten der Funktion beschrieben wird.

In Spalte 6 wird das Vorzeichen angegeben, was sich aus der Verknüpfung der Einflüsse von PV und TV ergibt.

Aus den Einzelgewichten des MLP wurden entsprechend Gleichung (2) die entkoppelten Korrelationsgewichte berechnet. Spalte 7 gibt Auskunft über das Vorzeichen des entkoppelten Korrelationsgewichtes und Spalte 8 gibt die Werte für G_i an.

Es zeigt sich, daß die Vorzeichen die aus dem funktionalen Zusammenhang extrahiert wurden (Spalte 6) mit den Vorzeichen der entkoppelten Korrelationsgewichte des MLP (Spalte 7) übereinstimmen.

Aus den Ergebnissen der bisherigen Untersuchungen lassen sich anhand der Vorzeichen der entkoppelten Korrelationsgewichte Relationen zwischen funktionalem Zusammenhang des Prozesses und Gewichtsverteilung des MLP ableiten, die in Form von Einzelregeln bzw. einer Regelmatrix formuliert werden können.

Einzelregeln

Für die Beschreibung der Abhängigkeit des Vorzeichens des entkoppelten Korrelationsgewichtes G_i von den Vorzeichen, die den funktionalen Zusammenhang charakterisieren, gelten folgende Einzelregeln:

- für das Vorzeichen PV bezüglich des Exponenten (des proportionalen Zusammenhanges)

$$\text{IF } Y \sim |X_i| \quad \text{THEN } G_i = (+) \quad (9a)$$

$$\text{IF } Y \sim |1/X_i| \quad \text{THEN } G_i = (-) \quad (9b)$$

- für das Vorzeichen TV bezüglich des Anstieges (des Einflusses des Terms)

$$\text{IF } m_i = (+) \quad \text{THEN } G_i = (+) \quad (10a)$$

$$\text{IF } m_i = (-) \quad \text{THEN } G_i = (-) \quad (10b)$$

Regelmatrix

Aus den möglichen Kombinationen der Vorzeichen PV und TV des funktionalen Prozeßzusammenhanges ergibt sich auf der Grundlage der Basisregel die in Tabelle 2 dargestellte Abhängigkeit für das Vorzeichen des entkoppelten Korrelationsgewichtes GV des MLP in Form einer Regelmatrix:

GV		PV	
		+	-
TV	+	+	-
	-	-	+

Basisregel
IF PV AND TV THEN GV

Tabelle 2: Vorzeichen des entkoppelten Korrelationsgewichtes GV in Abhängigkeit der Vorzeichen PV und TV des funktionalen Zusammenhanges

Ausgehend von der Kenntnis des funktionalen Zusammenhanges in Form der Vorzeichen PV und TV kann das Vorzeichen des sich einstellenden entkoppelten Korrelationsgewichtes GV im MLP vorausgesagt werden.

Im Umkehrschluß gibt das Vorzeichen des entkoppelten Korrelationsgewichtes Auskunft darüber, wie die Eingangsgröße X_i die Ausgangsgröße Y beeinflusst. Zur Zeit sind ohne zusätzliche Information keine Rückschlüsse darüber möglich, welche Kombination von PV und TV die Ausbildung eines positiven oder negativen Vorzeichens für GV verursacht.

$$\begin{array}{llllll} \text{IF} & \text{GV} = (+) & \text{THEN} & \text{PV} = (+) & \text{AND} & \text{TV} = (+) \\ & & & \text{OR} & \text{PV} = (-) & \text{AND} & \text{TV} = (-) \end{array} \quad (11)$$

$$\begin{array}{llllll} \text{IF} & \text{GV} = (-) & \text{THEN} & \text{PV} = (+) & \text{AND} & \text{TV} = (-) \\ & & & \text{OR} & \text{PV} = (-) & \text{AND} & \text{TV} = (+) \end{array} \quad (12)$$

Schlußfolgerung:

Die entkoppelten Korrelationsgewichte G_i geben Auskunft darüber, wie sich die den Trainingsdaten innewohnende Struktur (der Prozeßzusammenhang) in der Gewichtstopologie abbildet. Die entkoppelten Korrelationsgewichte sind hinsichtlich ihrer Aussagekraft für das MLP vergleichbar mit der für die Analyse von Daten zur Anwendung kommenden statistischen Kenngröße Korrelationskoeffizient r_{xy} [6].

4 Validierung der Ergebnisse

Zur Validierung der Ergebnisse der Sensibilitätsanalyse wird ein Prozeßbeispiel herangezogen, für das ein datenbasierter Zusammenhang existiert. Es handelt sich um die Abhängigkeit des Massenfüllstandes h im Plenum eines Siedewasserreaktors (Ausgangsgröße) von den Einflußgrößen Drehzahl der Zwangsumwälzpumpen n und Leistung des Reaktors P (Eingangsgrößen) [7].

$$h = f(n, P) \quad (13)$$

Das Kennfeld im Bild 4 zeigt den auf Daten basierenden Zusammenhang.

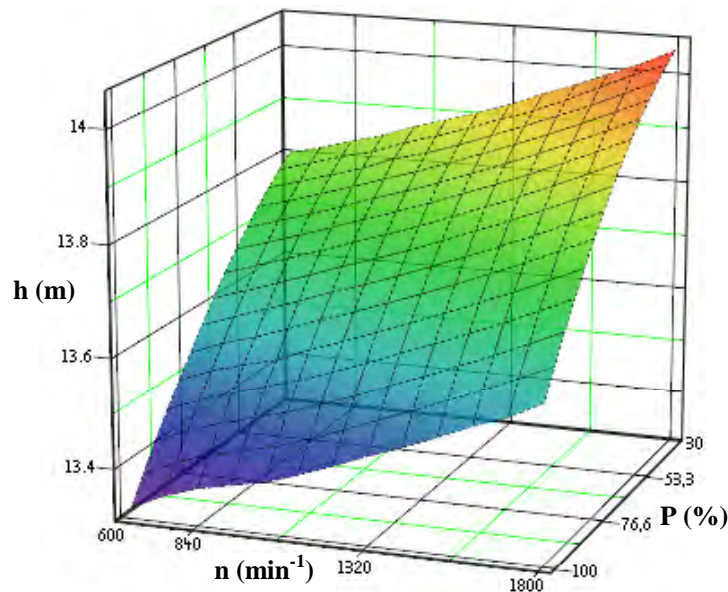


Bild 4: Kennfeld zur Beschreibung der Abhängigkeit des Massenfüllstandes h (m) von der Drehzahl der Zwangsumwälzpumpen n ($1/\text{min}$) und der Reaktorleistung P (%)

Werden die Daten einem MLP präsentiert und mit ausreichender Güte für den Trainings- und den Testfehler gelernt, lassen sich aus den Einzelgewichten des MLP die entkoppelten Korrelationsgewichte G_p für die Leistung und G_n für die Drehzahl ermitteln.

Das Training des MLP wurde wiederholt. Infolge der zufallsgenerierten Initialisierung der Startgewichte bildeten sich bei den Netzen MLP 1 und MLP 2 unterschiedliche Verteilungen der Einzelgewichte aus (Bild 5). Die entkoppelten Korrelationsgewichte blieben jedoch nahezu gleich (Tabelle 3).

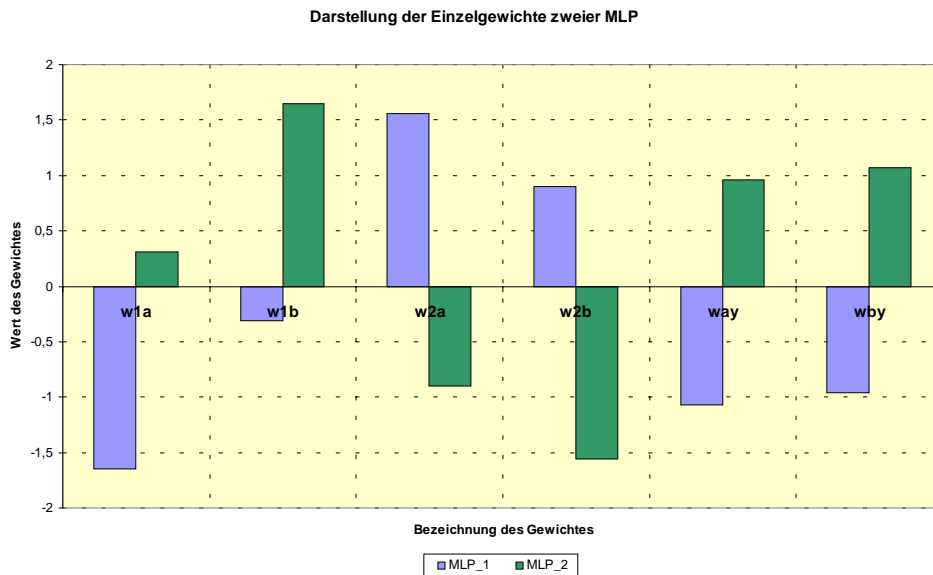


Bild 5: Vergleichende Darstellung der Einzelgewichte eines MLP bei Wiederholung der Trainingsaufgabe (Beispiel: Füllstandskennfeld)

Tabelle 3 stellt die Ergebnisse der Prozeßanalyse und der Analyse der Gewichtsverteilung gegenüber.

Prozeßzusammenhang aus Kennfeldanalyse		Korrelationskoeffizient r_{xy} aus Datenanalyse	Entkoppeltes Korrelationsgewicht G_i (MLP 1)	Entkoppeltes Korrelationsgewicht G_i (MLP 2)
IF	$n = \text{konst.}$	$r_{ph} = -0,86$	$G_p = -2,5329$	$G_p = -2,5308$
THEN	$h \sim 1/P$			
IF	$P = \text{konst.}$	$r_{nh} = +0,49$	$G_n = +2,0743$	$G_n = +2,0689$
THEN	$h \sim n$			

Tabelle 3: Ergebnisse der Prozeßanalyse und der Analyse der Gewichtsverteilung für das Prozeßbeispiel

Die entkoppelten Korrelationsgewichte, abgeleitet aus der Gewichtsverteilung des MLP geben den linearen Zusammenhang des Prozesses (Intensität und Richtung) wieder. Sie bilden sich analog der Kenngröße Korrelationskoeffizient aus, die aus der Datenanalyse des Prozesses abgeleitet werden kann. Die Ergebnisse der Sensibilitätsanalyse werden bestätigt.

5 Schlußfolgerungen

Die aus den Einzelgewichten eines MLP ermittelten entkoppelten Korrelationsgewichte geben Auskunft über die Intensität und die Stärke des linearen Zusammenhanges zwischen Eingangs- und Ausgangsgrößen.

Die den Prozeßdaten innewohnende Struktur wird somit auch in der Gewichtsverteilung des MLP abgebildet.

Anhand der entkoppelten Korrelationsgewichte kann überprüft werden, ob das MLP die Lerndaten und damit den zu trainierenden Prozeß nicht nur „gelernt“ sondern auch „verstanden“ hat.

Die entkoppelten Korrelationsgewichte können neben den bekannten Fehlerkenngrößen für die Evaluierung der Netzgüte eingesetzt werden.

Um Aussagen bezüglich der Eindeutigkeit und der Allgemeingültigkeit der oben dargestellten Ergebnisse abzuleiten sind weitere Untersuchungen erforderlich. Dies betrifft vor allem die Problematik des mehrfachen Auftretens von Zustandsgrößen im funktionalen Zusammenhang und die Auswirkung von stark unterschiedlichen Größenordnungen für die Werte der Eingangsgrößen. Weiterhin ist zu prüfen, ob es zu weiteren Kompensationseffekten bezüglich der Vorzeichen bzw. der Beträge der entkoppelten Korrelationsgewichte kommen kann.

Ungeachtet dessen stimmen die ersten Ergebnisse optimistisch, das sogenannte „Black-Box-Model“ des Multilayer Perzeptron mit Hilfe der Analyse der Gewichtsverteilung in ein „Grey-Box-Model“ zu überführen.

Die Untersuchungen wurden im Rahmen eines vom BMWi geförderten Forschungsprojektes (Projekt-Nr. 150 1248) durchgeführt.

Literatur

- [1] Boger: *Applications of ANN in Process Control Systems*. Vortrag im Rahmen des Seminars „Artificial Intelligence in Energetics“, Prag; 30.05.2000
- [2] Zell, A.: *Simulation Neuronaler Netze*. Addison-Wesley Publishing Company, Bonn; 1994
- [3] Minsky, M., Papert, S.: *Perceptrons*. The MIT Press; 1969
- [4] Kästner, W.: *Ein Beitrag zur Analyse der Gewichtsverteilung Neuronaler Netze*. IPM-Arbeitsbericht, Zittau; November 2000
- [5] Förster, T.: *Untersuchungen zur Gewichtsverteilung innerhalb eines Künstlichen Neuronalen Netzes (Multilayer Perzeptron)*. Diplomarbeit, Hochschule Zittau/Görlitz (FH); 2002
- [6] Storm, R.: *Wahrscheinlichkeitsrechnung mathematische Statistik und statistische Qualitätskontrolle*. Fachbuchverlag Leipzig-Köln; 1995
- [7] Schefter, S, Gocht, U.: *Generierung eines Kennfeldes zur Beschreibung der Abhängigkeit des Füllstandes im RDB von der Reaktorleistung und der Drehzahl der Zwangsumwälzpumpen mit dem ATHLET-Code*. Statusbericht, BMWi-Projekt Nr. 150 1248, Hochschule Zittau/Görlitz (FH), IPM; Dezember 2001

Anwendung von Neuro-Fuzzy-Methoden zur multisensoriellen Schadensdiagnose in Abwasserkanälen

Ch. Frey, H.-B. Kuntze, R. Munser

Fraunhofer-Institut für Informations- und Datenverarbeitung IITB
Fraunhoferstraße 1, 76131 Karlsruhe
Tel. (+49-721)-6091-310
Fax (+49-721)-6091-413
E-Mail: kuntze@iitb.fraunhofer.de

Zusammenfassung

Eine qualifizierte Schadensdiagnose und –sanierung in Abwasserkanälen erfordert den Einsatz von intelligenten Robotern, die mit TV-Technik und mit Multisensorik ausgerüstet sind. Für die Detektion und Vermessung von sichtbaren Schäden innerhalb des Kanalrohres bzw. unsichtbaren Anomalien (z. B. Leckagen) hinter der Rohrwand wurden am IITB im Rahmen eines DFG-Verbundvorhabens zwei leistungsfähige Sensoren sowie eine neuro-fuzzy-basierte Software zur intelligenten Fusion und Auswertung der Sensormerkmale entwickelt und unter realen Praxisbedingungen erprobt. Während sich frühere Beiträge der Autoren primär der Sensor-Hardware widmeten [2,5,6] wird sich der vorliegende Beitrag primär mit dem neuro-fuzzy-basierten Lösungskonzept der Multisensor-Fusion und dessen Anwendung auf die Schadensdiagnose in Abwasserkanälen beschäftigen.

Abstract

Qualified damage diagnosis and rehabilitation in sewer pipe networks requires more and more the introduction of mobile robots equipped with smart multi-sensorics besides conventional TV technology. For the detection and measurement both of visible damages inside the tube and non-visible anomalies behind the tube wall (e. g. leakages) within the framework of a joint research project funded by the Deutsche Forschungsgemeinschaft DFG at the IITB two efficient sensors and neuro-fuzzy-based software for smart sensor fusion and evaluation have been developed and tested under real practice conditions.

1 Motivation

Der Inspektion und Sanierung von Abwasserkanälen kommt eine zunehmende ökologische und ökonomische Bedeutung zu. Defekte Abwasserkanäle können z.B. erheblich zur Kontamination des Bodens und Grundwassers beitragen. Beim gegenwärtigen Stand der Technik werden zur Inspektion bisher ausschließlich fern gesteuerte mobile Roboter mit TV-Kamera verwendet. Auf der Grundlage dieser Technologie lassen sich natürlich nur gut sichtbare Schäden detektieren. Verdeckte Schäden (z.B. Hohlräume oder Wasseransammlungen hinter der Rohrwand) sowie

durch bauliche Sanierungsmaßnahmen verdeckte Hohlräume (z.B. durch Liner-Sanierung verdeckte Rohrabzweigungen) bleiben unentdeckt. Selbst prinzipiell sichtbare Schäden werden durch die subjektive Auswertung von TV-Bildern vom Inspektionspersonal häufig falsch oder unzureichend bewertet.

Eine umfassendere qualifiziertere Schadensdiagnose in kommunalen und industriellen Rohrnetzen ist nur dann möglich, wenn die auf den Inspektions- oder Sanierungsrobotern vorhandene TV-Technik, die nur eine sehr eingeschränkte subjektive Schadensdiagnose zulässt, durch eine objektiv messende schadensrelevante Multisensorik ergänzt wird. Am Markt stehen solche sensorbasierten Rohrinspektionssysteme gegenwärtig noch nicht zur Verfügung. Anders sieht es im Bereich der Forschung und Entwicklung aus. Ein erstes multisensorisches Schadensdiagnosesystem dieser Art wurde Mitte der neunziger Jahre vom IITB in Kooperation mit dem Forschungszentrum Karlsruhe und Industriepartnern im Rahmen des vom BMBF geförderten KARO-Projektes entwickelt und prototypisch realisiert [1, 2]. Im Gegensatz zu anderen sensorbasierten Kanalrobotersystemen in denen ausschließlich unter Verwendung optischer und Ultraschallmessverfahren Schäden im Rohrinernen inspiziert werden, geht das am IITB verfolgte multisensorielle Inspektionskonzept davon aus, sowohl sichtbare Schäden innerhalb des Rohres als auch unsichtbare Schäden außerhalb des Rohres zu detektieren und diagnostizieren.

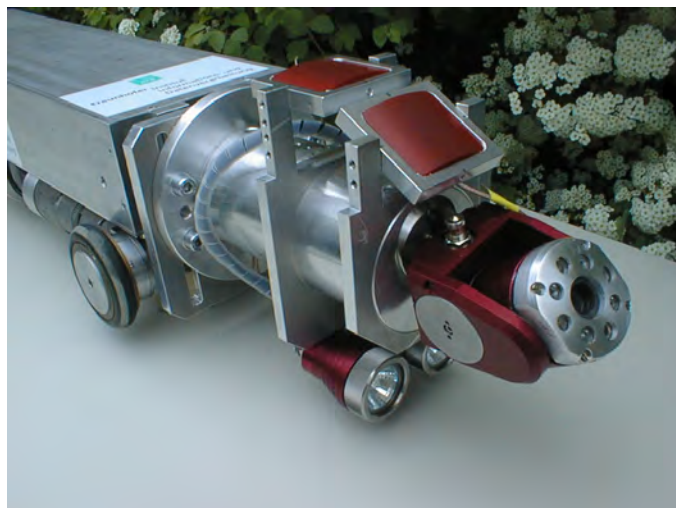


Abb. 1: Roboter mit MRS und LSS mit je einer positionierbaren Sende- und Empfangsantenne

Die Weiterentwicklung der im KARO-Projekt entstandenen Multisensorik des Mikrowellen-Rückstreu-Sensors (MRS) zur Detektion und Lokalisierung von verdeckten Anomalien hinter der Kanalrohrwand, des axialen 3D-Lichtschnitt-Sensors (LSS) zur Detektion und Vermessung von geometrischen Anomalien (z.B. Verformungen) innerhalb des Kanalrohres sowie die Schaffung einer intelligenten Neuro-Fuzzy-basierten Auswertungssoftware für eine große Bandbreite von Schadensklassen sind Gegenstand eines noch laufenden DFG-Verbundvorhabens mit der Universität Karlsruhe und dem Tiefbauamt der Stadt Rastatt [3-6]. Über neueste Ergebnisse dieses Vorhabens, die sich nicht nur auf Inspektions- sondern auch auf Sanierungsaufgaben beziehen wird im Folgenden berichtet (Abb. 1).

2 Rohrrinneninspektion mit axialem 3D-Lichtschnittsensor

Im Gegensatz zu den räumlich sehr platzaufwendigen konventionellen Lichtschnittverfahren beruht das am IITB entwickelte LSS-Verfahren darauf, dass die optischen Achsen des Strukturprojektors und der Kamera z. B. mittels eines Strahlteilers virtuell zur Deckung gebracht werden. Die für die Messgenauigkeit entscheidende Abstandsstrecke der Komponenten (Triangulationsbasis) liegt auf dieser optischen Achse (Abb. 2). Mit entsprechenden Anordnungen lässt sich dadurch eine genauere Vermessung im Weitwinkelbereich, eine minimale Baugröße senkrecht zur Inspektionsrichtung sowie eine Symmetrie zur optischen Achse erreichen.

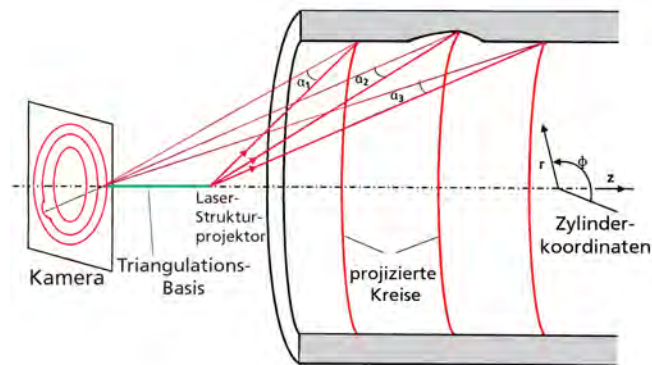


Abb. 2: Funktionsprinzip und TV-Kamera mit integriertem axialem 3D-Lichtschnitt Sensor LSS

Unter Verwendung entsprechender Hologramme kann der Strukturprojektor sowohl Kreise als auch Quadrate in das Rohr projizieren. Zahl und Öffnungswinkel der sich ergebenden Laserkegel bzw. -pyramiden sind einstellbar [3]. Der LSS liefert ein für jeden Schadenstyp charakteristisches 2D-Strukturmuster (Primärklassifikation) sowie die 3D-Maße von Schäden und Geometrien (Rissbreite, Muffenversatz, Rohrdurchmesser etc.). Die von den optischen Konstruktionsparametern abhängige Messgenauigkeit des breitbandig anwendbaren LSS ist erheblich höher als die in der Praxis von der ATV (Abwassertechnische Vereinigung) geforderte Auflösung von ca. 1% des Rohrdurchmessers.

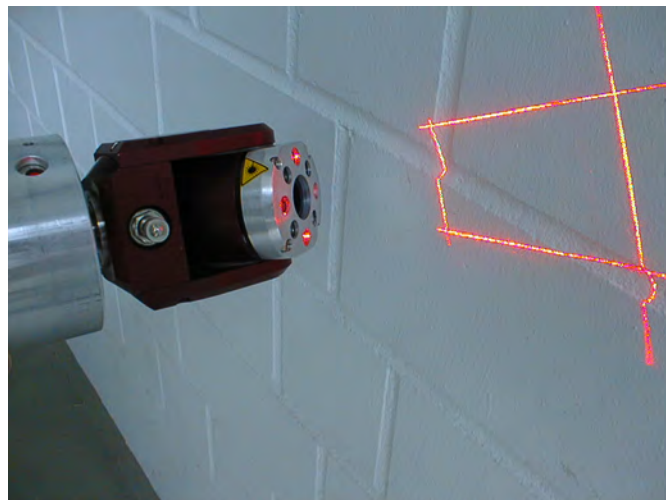


Abb. 3: Integration des LSS in eine marktübliche (IPEK) Inspektionskamera

Durch die geringe Baugröße lässt sich der LSS relativ leicht in marktübliche Kameraköpfe integrieren. Bei der in Abb. 3 dargestellten Realisierung sind z. B. 4 Weißlichtdioden (Beleuchtung) und 4 Laserdioden (Strukturprojektor) wechselseitig überlappend in den Blendenring der Kamera integriert. Das projizierte Laserquadrat ergibt an einer ebenen Ziegelwand linienförmige Lichtschnitte mit charakteristischen Brechungen in den Ziegelfugen (Abb. 3).

Die mit dem integrierten LSS in Kanalrohren aufgenommenen Lichtschnitte in Abb. 4 zeigen klar die symmetrischen Hyperbeln beim intakten Rohr sowie sehr deutlich die Muffenspalt- bzw. Riss-Erkennung und -Vermessungsmöglichkeit. In Abb. 5 sind das Weißlichtbild links und das "Lichtschnittbild" rechts mit gleicher Kameraposition aufgenommen worden. Deutlich sichtbar ist der mit dem LSS mögliche direkte Vergleich aus genau gleicher Perspektive. So können genau die Punkte, die der Bediener in seinem Bild z.B. anklickt, mit dem LSS vermessen werden. Anstatt des Ein-Ausschaltens des Projektors kann auch eine ständig eingeschaltete Lichtquelle im nicht sichtbaren Spektralbereich verwendet werden, für die die Messkamera empfindlich ist.

Nicht nur zur Detektion und Lokalisierung von Schäden und Hohlräumen, sondern auch zur Qualitätskontrolle von Sanierungsarbeiten lässt sich der LSS sehr gut einsetzen. Im Fall einer Liner-Sanierung (vgl. Abschnitt 5) lassen sich z.B. Ausbeulungen bei schlecht anliegenden Kunststoff-Liner-Rohren oder unzulässige Fugen im Bereich von Muffen und Hausanschlüssen detektieren und vermessen.

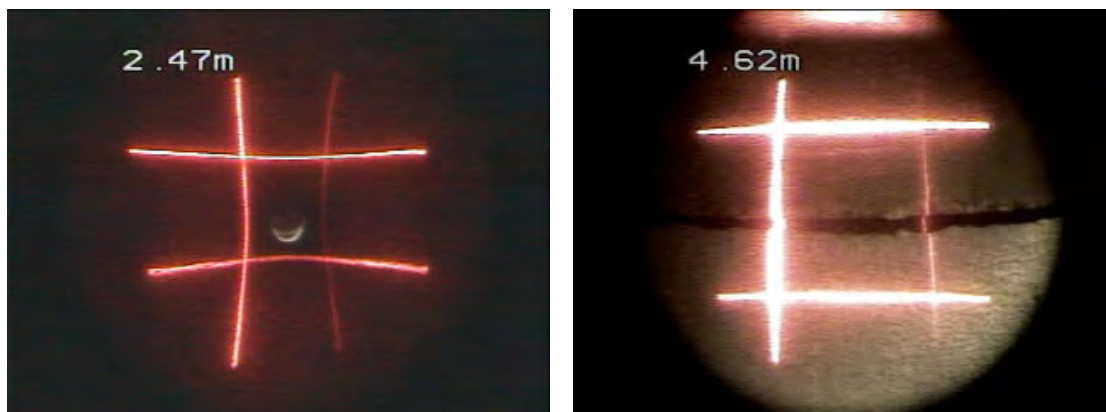


Abb. 4: Gemessene Lichtschnitte in der Kanalteststrecke des IITB (DN 300): links intaktes Rohr geradeaus, rechts Schwenk auf eine Muffe

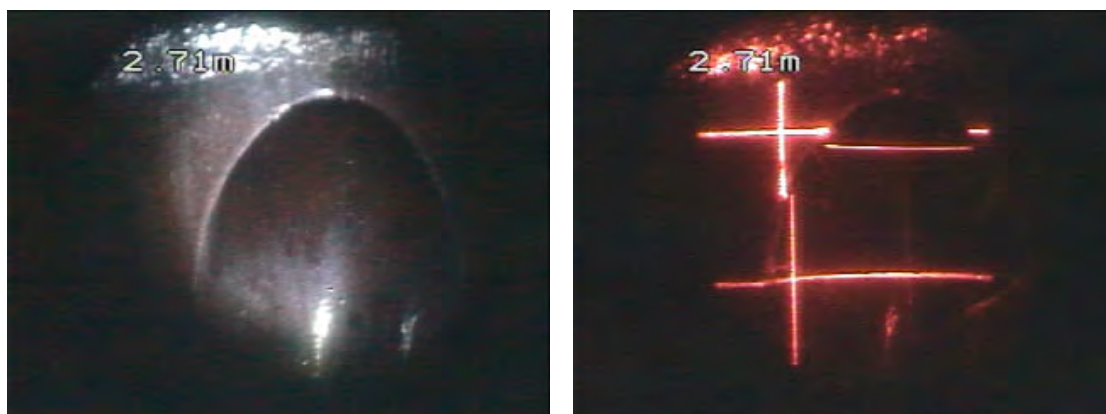


Abb. 5: Hausanschluss im Weißlicht sowie mit Lichtschnitt-Beleuchtung

3 Rohraußeninspektion mit Mikrowellen-Rückstreusensor

Mit Mikrowellen ist es möglich, durch die Wand von Steinzeug-, Beton- und Kunststoff-Rohren hindurch die Rohr Umgebung zu inspizieren. Bei 2-3 GHz durchdringen sie eine nichtmetallische Wand und trockene Umgebung bis maximal 0,3 m und reagieren auf Änderungen der relativen Dielektrizitätskonstante ϵ . Bei der Rückstreu-Anordnung des am IITB entwickelten und patentierten Mikrowellen-Rückstreu-Sensors (MRS) wird analog zu optischen Sensoren die an den Rohrwänden reflektierte Strahlung weitgehend unterdrückt, so dass besonders die durch Anomalien gebildeten Streuzentren detektiert werden (Abb. 6) [7]. Zur Ermittlung der optimalen Antennenanordnung wurden im Testbett des IITB systematische Messfahrten bei unterschiedlichen Neigungs- und Polarisationswinkeln der Sende- und Empfangsantenne unternommen.

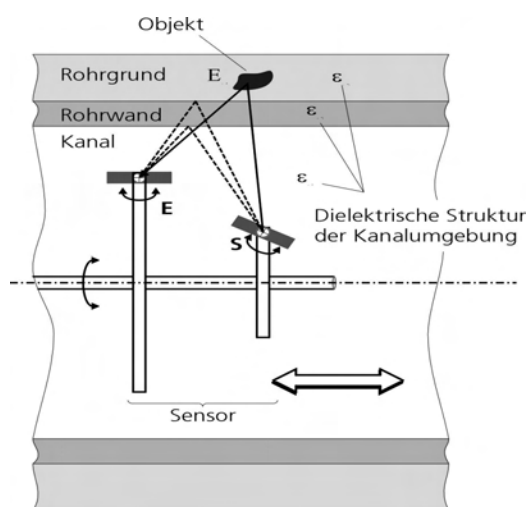


Abb. 6: Aufbau und Messprinzip

Unter diesen optimalen Winkeln können die Antennen auch in Form von 2 Gürteln um den Umfang des Trägerfahrzeugs fest angebracht werden, so dass die konstruktiv aufwendige und verschleißanfällige mechanische Bewegung durch eine schnelle elektronische Umschaltung ersetzt wird (Abb. 7). Die Gürtel mit je 4 Antennen sind um 45° gegeneinander versetzt, so dass das Rohr bzgl. des Rollwinkels in 8 Segmente unterteilt wird.

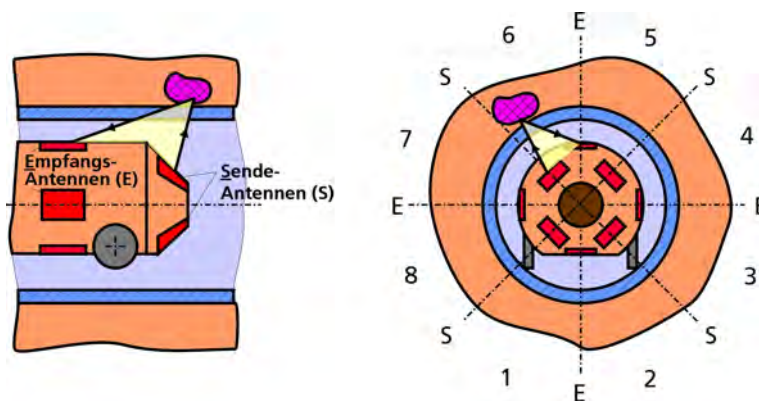


Abb. 7: Gürtelförmige Anordnung der MRS-Antennen des MRS

Das charakteristische MRS-Systemverhalten bei Fahrten entlang der Rohrsohle mit konstantem Rollwinkel des Antennenpaares wird durch die Messsignalverläufe in Abb. 8 vermittelt. Neben der sehr guten Reproduzierbarkeit lassen die 5 gemessenen Amplituden- und Phasen-Signalverläufe sehr deutlich vorhandene Wasserstellen W in der Rohrbettung, Muffen M, den Hausanschluss Q sowie den Materialübergang von Ton- zu Betonrohr bei 3,1 m erkennen.

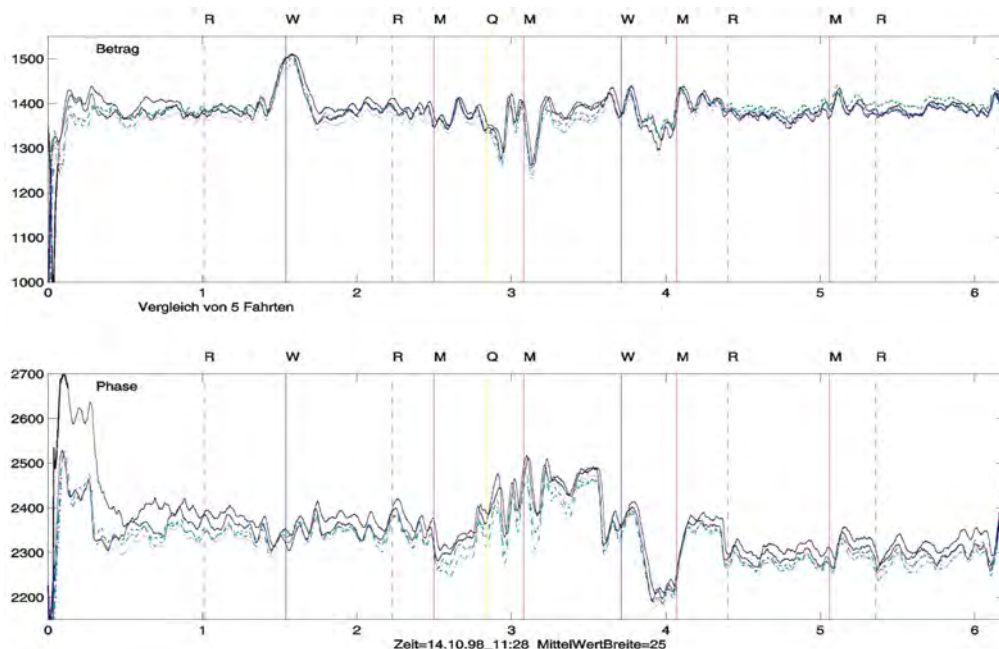


Abb. 8: Reproduzierbare Signalverläufe des MRS im IITB-Testbett

4 Ganzheitliche Schadensdiagnose durch Sensordatenfusion

Das am IITB entwickelte multisensorielle Online-Diagnoseverfahren, das auch zur Überwachung von Mechatroniksystemen (z. B. Bohrhämmer [8]) oder verfahrenstechnischen Industrieprozessen [10] eingesetzt wird, setzt sich aus den zwei aufeinander folgenden Schritten zusammen.

Der erste Schritt, die Merkmalgenerierung dient zur Aufbereitung der gemessenen Prozesssignale. Aus der Vielzahl der zur Verfügung stehenden Sensorsignale werden die relevanten Signalcharakteristika extrahiert und in einem Merkmalvektor geringerer Dimensionalität zusammengefasst. Zur Merkmalgenerierung werden häufig signalgestützte Verfahren (z. B. Fuzzy-basierte Grenzwert-, Trend- und Spektral- bzw. Waveletanalysen), modellgestützte Verfahren (z. B. Kalman-Filter) und wissensgestützte Verfahren (z. B. Expertensysteme) eingesetzt.

Der zweite Schritt, die Merkmalauswertung ist ein logischer Entscheidungsprozess, der quantitatives Wissen in qualitatives transformiert. Ziel ist es zu entscheiden, ob und zu welchem Zeitpunkt ein bestimmter Fehler oder eine Prozessphase aufgetreten ist. Hierzu können statistische Methoden (z. B. generalized likelihood ratio test), merkmalsbasierte Mustererkennungsmethoden (z. B. Bayes-Klassifikatoren, Fuzzy-Logik oder Neuronale Netze) eingesetzt werden.

Aufgrund der natürlichen Unschärfe der hier verwendeten Sensorinformationen bieten Neuro-Fuzzy-Methoden wesentliche Vorteile. Einerseits wird die Zugehörigkeit zu einer bestimmten Prozessphase in Form von Mitgliedsfunktionen $\mu(x)$ [unsharp beschrieben], andererseits wird der Zusammenhang zwischen den Merkmalen und den zu klassifizierenden Prozessphasen in Form linguistischer Regeln definiert, wie z.B.:

wenn	Merkmal 1	=	mittel
und	Merkmal 2	=	groß
und	=
dann	Schadensklasse	=	kleiner Hohlraum.

Ein weiterer Vorteil liegt in der Interpretierbarkeit der Fuzzy-Entscheidungsmodule, das Verhalten kann durch den Entwickler leicht modifiziert werden, zudem ist es möglich, vorhandenes Expertenwissen in Form von linguistischen Regeln einfach zu integrieren.

Zur Unterstützung bei der Implementierung der Fuzzy Auswertemodule wird der Neuro-Fuzzy-Ansatz NEFCLASS (NEuro-Fuzzy-CLASSifikation) genutzt [6, 9]. Das hybride Neuro-Fuzzy-Modell basiert auf dem generischen Modell eines dreischichtigen FL-Perceptrons, wodurch das Verhalten in Form von linguistischen Regeln interpretiert werden kann. Das in Abb. 9 dargestellte Modul beinhaltet beispielsweise drei Eingangsvariablen in der Eingabeschicht mit je drei Mitgliedsfunktionen $m(x)$, vier Regeln in der verdeckten Schicht sowie zwei Klassen in der Ausgabeschicht.

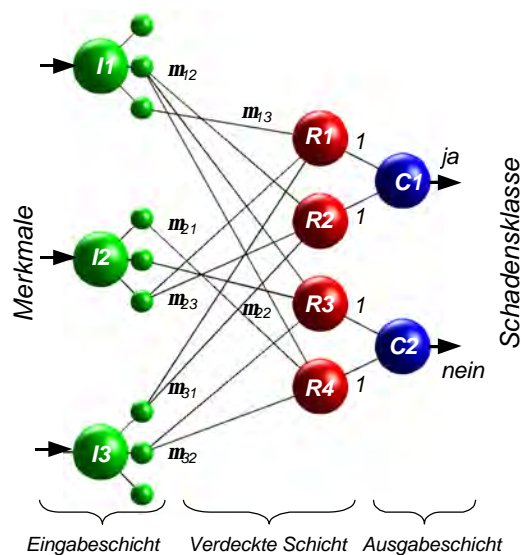


Abb. 9: Lernfähiges Neuro-Fuzzy-Modell zur Schadensdiagnose

Die gewichteten Verbindungen zwischen Eingabe- und Ausgabeschicht werden durch die Fuzzy-Mengen $m(x)$ hergestellt (Prämissen). Um semantische Probleme zu vermeiden, sind die Verbindungen der Regeln zur Ausgabeschicht als konstante Gewichte angenommen, es werden also keine gewichteten Regeln verwendet. Durch diese Vorgehensweise bleibt die Interpretierbarkeit des Modells erhalten, da die Regeln einfach ablesbar sind. Der überwachte Lernalgorithmus des NEFCLASS Modells ist in der Lage, sowohl die Regeln zu erlernen als auch die Parameter der den Regeln zugrunde liegenden Mitgliedsfunktionen $m(x)$ zu adaptieren. Die Bildung der Regelbasis kann zum einen auf der Basis von Expertenwissen erfolgen, zum anderen aber auch ohne Vorwissen inkrementell erzeugt werden.

Die bei der sensorbasierten Schadensdetektion verwendeten Schadensklassen werden anhand der Sensorfähigkeiten (Messgrößen, Messbereich, Auflösung etc.) und der

physikalischen Gegebenheiten des Abwasserkanals (Rohrtyp, Materialart, Bettungsmaterial, etc.) definiert und spezifiziert. Sie sind nicht identisch mit den standardisierten Schadensklassen der nationalen und europäischen Inspektionsstandards (ATV-Klassifikation bzw. EN 13508-Standard), die von der klassischen TV-Inspektion ausgehen.

Die in Abb. 10 dargestellten Diagnoseergebnisse wurden in der mit verschiedenen Anomalien präparierten Rohrteststrecke des IITB [6, 11] gewonnen. Es wird sichtbar, dass sowohl Anomalien (Luftlöcher, Wasserlöcher, etc.) als auch konstruktionsbedingte Inhomogenitäten (z.B. Muffen) zweifelsfrei diagnostiziert werden.

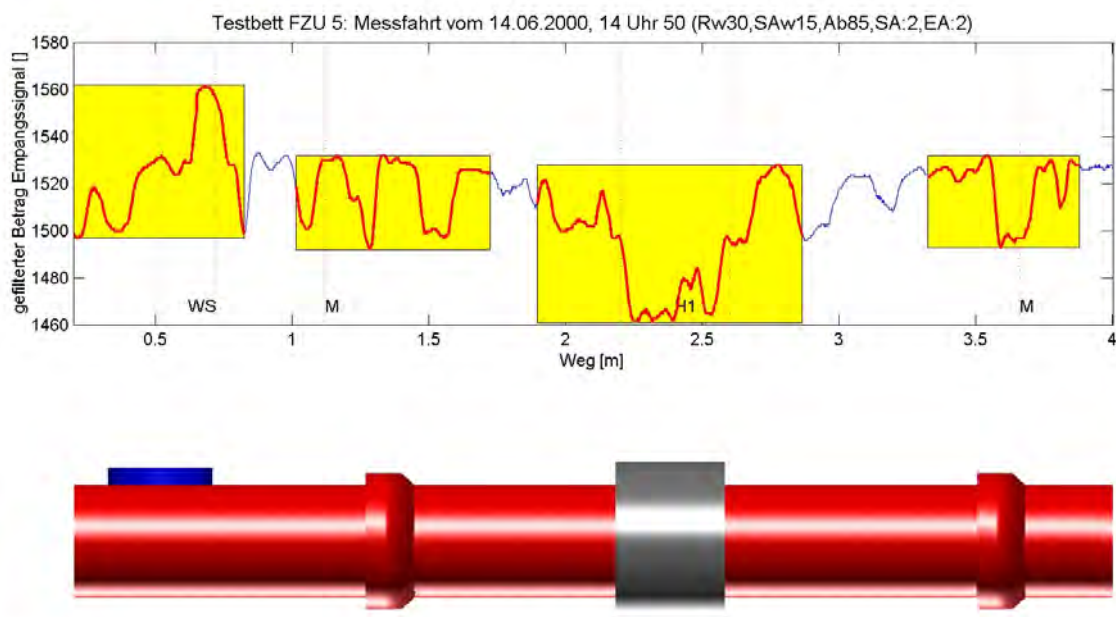


Abb. 10: Neuro-Fuzzy-basierte Schadensdiagnose-Diagnose mit MRS

Die mit Hilfe der Neuro-Fuzzy-Diagnosesoftware ermittelten Ergebnisse beinhalten zunächst nur qualitative Ergebnisse, d.h. das Auftreten vorher spezifizierter Schadensklassen (z.B. mittlerer Riss, großer Hohlraum, kleine Absickerung, etc.). Diesen qualitativen Aussagen lassen sich natürlich genaue quantitative Ergebnisse (z.B. Rissbreite 1,5 mm), die von den jeweiligen Sensoren gemessen wurden, ergänzend zuordnen.

5 Literatur

- [1] Kuntze, H.-B.; Schmidt, D.; Haffner, H.; Loh, M.: KARO - ein flexibel einsetzbarer Roboter zur intelligenten sensorbasierten Kanalinspektion. Berichte zur NO DIG'95-Konferenz in Dresden, 19.-22.9.1995, Hamburg-Messe & Congress GmbH, 1995, S. 357-365 (englische Fassung parallel erschienen)
- [2] Kuntze, H.-B; Haffner, H.: Experiences with the development of a robot for smart multisensoric pipe inspection. In: Proc. 1998 IEEE International Conference on Robotics and Automation, Leuven/Belgium, May 16-20, 1998, Vol. 2, Piscataway, NJ., 1998, S. 1773-1778
- [3] Hartrumpf, M.; Munser, R.: Verfahren zur dreidimensionalen Vermessung unzugänglicher Hohlräume. Deutsches Patentamt München, Aktenzeichen P4325542.6, Tag der Anmeldung: 29.07.1993, Tag der Offenlegung: 02.02.1995

- [4] Munser, R.; Kuntze, H.-B.; Hartrumpf, M.: Verfahren zum Wiederauffinden von vorgebbaren Stellen in Kanalsystemen und Rohrleitungen. Deutsches Patentamt München, Aktenzeichen P10014703.8, Tag der Anmeldung 24.03.2000
- [5] Munser, R.; Kuntze, H.-B.; Hartrumpf, M.: Eine modulare Multisensoreinheit für mobile Roboter zur intelligenten Leckage- und Schadensortung in Rohrleitungssystemen. VDI-Berichte 1552 Tagung Robotik 2000, Berlin, ISBN 3-18-091552-8, 29.-30.06.2000, S. 221-230
- [6] Munser, R.; Kuntze, H.-B.; Hartrumpf, M.; Frey, Chr. W.: Ein modulares Multisensorsystem für Rohrinspektions- und Rohrsanierungsroboter. In: Proceedings 16. Fachgespräch "Autonome Mobile Systeme" (AMS 2000), Karlsruhe, 20.-21.11.2000
- [7] Hartrumpf, M.; Munser, R.: Verfahren zur Detektion ganz oder teilweise verborgener Inhomogenitäten in undurchsichtigen Medien mittels Mikrowellenstrahlung. Deutsches Patentamt München Aktenzeichen P43400590, Tag der Anmeldung: 24.11.1993, Tag der Offenlegung: 01.06.1995, Patent erteilt am 05.01.1996 unter der Nr. 4340059
- [8] Jacubasch, A.; Frey, C.W.; Kuntze, H.-B.: "Intelligente neuro-adaptive Bewegungsregelung eines Bohrhammers". Proc. Robotik 2002, Ludwigsburg, 19./20.06.2002
- [9] Nauck, D.; Klawonn, R.; Kruse, R.: Neuronale Netze und Fuzzy Systeme. Vieweg Verlag, 1994
- [10] Frey, C.; Sajidman, M.; Kuntze, H.-B.: A Neuro-Fuzzy Supervisory Control System for Industrial Batch Processes. Proc. 9th IEEE Int. Conf. on Fuzzy Systems FUZZ'2000, May 7-10, 2000 San Antonio (USA), pp. 116-121
- [11] Munser, R., Kuntze, H.-B., Hartrumpf, M.: Eine modulare Multisensoreinheit zur intelligenten Leckage- und Schadensortung in Rohrleitungssystemen. IITB-Mitteilungen 1999, Fraunhofer IITB, S. 21-30
- [12] Stein, D.; Niederehe, W.: "Instandhaltung von Kanalisationen". Berlin: Ernst-Verlag für Architektur und techn. Wiss., 1992, ISBN 3-433-01177-X

Funktionswurzeln und Lernalgorithmen für abgetastete Systeme

Roland Kempf, Harald Staab

Technische Universität Darmstadt, Institut für Automatisierungstechnik,
Landgraf-Georg Straße 4, D-63482 Darmstadt
Telefon: (06151) 16-3542, Fax: (06151) 16-2507,
E-Mail: rdkempf@rtr.tu-darmstadt.de

Zusammenfassung:

Durch die zeitliche Abtastung eines zeit- und wertekontinuierlichen Systems erhält man ein zeitdiskretes System. Kann man während des Zeitintervalls den Eingangsgrößenvektor \mathbf{u} als konstant voraussetzen, so läßt sich ein funktionaler Zusammenhang zwischen der alten und neuen Zustandgröße angeben: $\mathbf{x}(k+1) = \mathbf{f}_{\mathbf{u}}(\mathbf{x}(k))$, wobei der Eingangsgrößenvektor \mathbf{u} als Scharparameter für die Funktionen $\mathbf{f}_{\mathbf{u}}$ betrachtet werden kann.

Die Menge der abgetasteten Systeme bildet allerdings nur eine kleine Teilmenge in der Menge der zeitdiskreten Systeme. Sie zeichnen sich unter anderem auch dadurch aus, dass die Transitionsfunktionen $\mathbf{f}_{\mathbf{u}}$ beliebige sogenannte Funktionswurzeln besitzen muß, d.h. es gibt Funktionen $\mathbf{f}_{\mathbf{u}}^{1/q}(\mathbf{x})$, so dass die q -fache Hintereinanderschaltung $\mathbf{f}_{\mathbf{u}}^{1/q} \circ \mathbf{f}_{\mathbf{u}}^{1/q} \circ \dots \circ \mathbf{f}_{\mathbf{u}}^{1/q}$ genau die Funktion $\mathbf{f}_{\mathbf{u}}$ ergibt.

Will man nun ein solches abgetastetes System identifizieren, so muß das Ergebnis diese Eigenschaft bzw. daraus abgeleitete Eigenschaften besitzen. Dies schränkt den Suchraum während einer Identifikation erheblich ein. Zum Teil können dadurch auch unerwünschte dynamische Verhaltensweisen, wie etwa chaotisches Verhalten, bei den Systemen vermieden werden, die sich andernfalls während des Identifikationsprozesses ergeben können. Besonders bei Lernalgorithmen, die ohne eine vollständige Beschreibung der internen Zustandsgrößen auskommen, wie z.B. "backpropagation through time", erhöht sich dadurch die Lerngeschwindigkeit und Erfolgsquote.

In diesem Beitrag werden einige Grundlagen im Bereich der Funktionswurzeln angesprochen und dargestellt wie man die Eigenschaften von Funktionen mit beliebigen Funktionswurzeln ausnutzen kann, um Lernverfahren zu modifizieren. Getestet wird ein modifizierter "backpropagation through time"-Algorithmus zur Identifikation eines einfachen Neuro-Fuzzy Systems.

1 Problemstellung

Bei der Identifikation abgetasteter wertkontinuierlicher Systeme ist das Ziel eine Funktion f zu finden, die den Übergang des Zustandes $x(k)$ eines Systems zum Zeitpunkt $k \cdot \Delta t$ auf den Zustand $x(k + 1)$ zum Zeitpunkt $(k + 1) \cdot \Delta t$ je nach Belegung der möglichen Eingabewerte $u(k)$ beschreibt, also:

$$x(k + 1) = f(x(k), u(k)), \text{ wobei } f \text{ stetig und } k \in \mathbb{N}. \quad (1)$$

Die Gleichung (1) beschreibt allerdings eine weit größere Funktionsklasse als die der abgetasteten wertkontinuierlichen Systeme. Betrachtet man verschiedenen stetige Funktionen f , so können nach Anwendung der Gleichung (1) sehr unterschiedliche Dynamiken entstehen. Insbesondere

- konvergentes Verhalten, z.B. für $f(x) = 0.5x + 0.25$,
- divergentes Verhalten, z.B. für $f(x) = 2x - 0.5$,
- oszillatorisches Verhalten, z.B. für $f(x) = 1 - x$, und sogar
- chaotisches Verhalten, z.B. für $f(x) = \begin{cases} 2x & \text{für } x \leq 1, \\ 2 - 2x & \text{für } x > 1. \end{cases}$

Abbildung 1 zeigt die Funktionen und Zeitreihen, die durch sie definierten dynamischen Systeme.

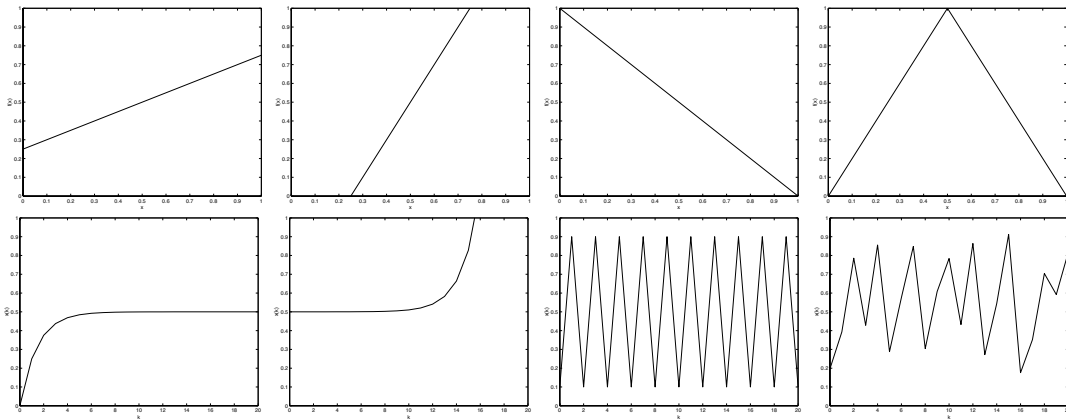


Abbildung 1: Verschiedene Funktionen (oben) erzeugen Zeitreihen (unten) mit qualitativ unterschiedliche Dynamiken.

Allerdings tauchen nicht alle dieser Verhaltensweisen auch in abgetasteten wertkontinuierlichen Systemen auf. Bei der Identifikation ist daher darauf zu achten, Funktionstypen mit solchen Verhalten auszuschließen. Dazu muß man die erlaubten Verhaltensweisen, bzw. die zugelassenen Funktionen f näher beschreiben. Wie im nächsten Abschnitt erläutert wird, besitzen abgetastete wertkontinuierliche Systeme, im Gegensatz zu vielen zeitdiskreten Systemen, beliebige sogenannter Funktionswurzeln.

2 Existenz beliebiger Funktionswurzeln

Betrachtet wird ein abgetastetes wertekontinuierliches System, dessen Eingabewerte u während eines Abtastschrittes als konstant vorausgesetzt wird. Dieses System wird eindeutig durch eine Gleichung der Form wie Gleichung (1) bzw. durch

$$x(T + \Delta t) = f_u(x(T)) \quad (2)$$

mit $T = k \cdot \Delta t$ beschrieben.

Tastet man das System mit der doppelten Abtastrate ab, so ergibt sich eine andere Funktion $f_u^{1/2}$ die, anstelle von f_u , das System beschreibt. Mit dieser Funktion gilt:

$$x(T + \frac{1}{2} \cdot \Delta t) = f_u^{1/2}(x(T)) \quad (3)$$

$$x(T + \Delta t) = f_u^{1/2}(x(T + \frac{1}{2} \cdot \Delta t)) \quad (4)$$

und somit

$$x(T + \Delta t) = f_u^{1/2}(f_u^{1/2}(x(T + \Delta t))) \quad (5)$$

Vergleicht man Gleichung (2) und Gleichung (5), so erhält man die folgende Beziehung zwischen den beiden Funktionen f_u und $f_u^{1/2}$:

$$f_u = f_u^{1/2} \circ f_u^{1/2}. \quad (6)$$

Bei q -facher Abtastrate erhält man analog Funktionen $f_u^{1/q}$ mit

$$f_u = \underbrace{f_u^{1/q} \circ \dots \circ f_u^{1/q}}_{q\text{-fach}}. \quad (7)$$

Die Funktionen $f_u^{1/q}$ nennt man auch q -te Funktionswurzeln von f_u , weil in Analogie zur Multiplikation ihre q -fache Anwendung der ursprünglichen Funktion f_u gleicht. Durch die p -fache Hintereinanderschaltung der Funktion $f_u^{1/q}$ sind dann auch die Funktionen $f_u^{p/q}$ festgelegt.

Für eine Übergangsfunktion f_u eines abgetasteten wertekontinuierlichen Systems muß es also für jeden positiven Bruch p/q stetige Funktionen $f_u^{p/q}$ geben.

3 Auswirkung der Existenz von Funktionswurzeln auf abgetastete wertekontinuierlicher Systeme

Exemplarisch werden die Auswirkungen der Existenz von Funktionswurzeln auf abgetastete wertekontinuierlicher Systeme nun anhand eines linearen, eindimensionalen System demonstriert. Anschließend werden nichtlineare, eindimensionale Systeme behandelt.

3.1 Lineares System

Ein lineares, eindimensionales System läßt sich bei in einem Abtastschritt festgesetzten Eingabewerten u , nach Transformation einer Ruhelage in den Ursprung allgemein wie folgt beschreiben:

$$x(T + \Delta t) = f_u(x(T)), \text{ mit } f_u(x) = a_1 \cdot x. \quad (8)$$

Die Übergangsfunktion der doppelt abgetasteten Systems wird nun durch eine Funktionswurzel von $f_u(x)$ beschrieben. Da das doppelt abgetastete System selbst eindimensional und linear sein muß, ist die Funktionswurzel $f_u^{1/2}$ durch folgenden Ansatz beschreibbar:

$$f_u^{1/2}(x) = a_2 \cdot x. \quad (9)$$

Nach Gleichung (5) ergibt sich dann

$$a_1 \cdot x = f_u(x) = f_u^{1/2}(f_u^{1/2}(x)) = a_2 \cdot a_2 \cdot x \quad (10)$$

und somit $a_1 = a_2^2$. Damit die lineare Funktionswurzel $f_u^{1/2}(x)$ existiert, muß daher $a_1 \geq 0$ sein. Analog muß auch $a_2 \geq 0$ sein, damit eine vierte lineare Funktionswurzel $f^{1/4}(x)$ von $f(x)$ existiert, die gleichzeitig eine zweite lineare Funktionswurzel von $f^{1/2}(x)$ darstellt. Somit ist der Parameter a_2 durch $a_2 = +\sqrt{a_1} = a_1^{1/2}$ eindeutig festgelegt.

Analog überlegt man sich, dass ungerade q -fache lineare Funktionswurzeln von f_u durch $f^{1/q}(x) = a_1^{1/q} x$ eindeutig festgelegt sind. Kombiniert man die Überlegungen zu geraden und ungeraden Funktionswurzeln von f_u so ergibt sich $f^{1/q}(x) = a_1^{1/q} x$ und $f^{p/q}(x) = a_1^{p/q} x$ für beliebige Brüche p/q , wobei $p, q \in \mathbb{N}$.

Da das Systemverhalten stetig in der Zeit ist, können in diesem Fall auch "reellen Funktionswurzeln" definiert werden durch:

$$f_u^{t/\Delta t}(x) = a_1^{t/\Delta t} x = e^{t/\Delta t \cdot \ln(a_1)} x = e^{t/\tau} x, \text{ wobei } \tau = \frac{\Delta t}{\ln(a_1)}. \quad (11)$$

Schreibt man Gleichung (11) um, so erhält man die bekannte Darstellung des Zeitverlaufs der Zustandsgröße $x(T + t)$ zwischen zwei Abtastzeiten T und $T + \Delta t$ durch:

$$x(T + t) = f_u^{t/\Delta t}(x(T)) = e^{t/\tau} x(T), \text{ mit der Zeitkonstanten } \tau = \frac{\Delta t}{\ln(a_1)}. \quad (12)$$

Durch die Forderung, dass alle Funktionswurzeln auch linear sein sollen, ist es somit möglich, den kontinuierlichem Zeitverlauf der Zustandsgröße eindeutig zu rekonstruieren. Bei den Untersuchungen wurde außer Acht gelassen, dass es womöglich nichtlineare Funktionswurzeln von linearen Funktionen gibt. Im nichtlinearen Fall ist diese Frage allerdings wichtig.

3.2 Nichtlineares System

Bei nichtlinearen Systemen ist die Bestimmung der Funktionswurzeln aufgrund eines fehlenden Ansatzes nicht so einfach möglich, wie die Bestimmung von linearen Funktionswurzeln von linearen Systemen. Funktionswurzeln sind in der Regel auch nicht eindeutig

festgelegt [4, 2]. Trotzdem lassen sich noch einige Aussagen über eindimensionale, nichtlineare abgetastete Systeme treffen. Es gilt folgender

Satz:

Eine eindimensionale, stetige Funktion f , die ein reelles, beschränktes und abgeschlossenes Intervall X in sich selbst abbildet besitzt nur dann beliebige stetige Funktionswurzeln f^r mit $r \in \mathbb{R}$ in X , wenn die Funktion die Funktion f monoton steigend auf X ist.

Beweis:

Angenommen die Funktion f ist nicht monoton steigend X . Dann gibt es Punkte $x_1 < x_2$ aus X mit $f(x_1) > f(x_2)$. Da $f^t(x)$ stetig in t ist, gibt es auch ein $t \in (0, 1)$, so dass $f^t(x_1) = f^t(x_2)$ und damit folgt $f(x_1) = f^{1-t}(f^t(x_1)) = f^{1-t}(f^t(x_2)) = f(x_2)$ im Widerspruch zu $f(x_1) > f(x_2)$. \square

Dieser Satz hat eine große Auswirkung auf mögliche Dynamikarten von eindimensionalen, nichtlinearen abgetasteten Systemen, welche sich durch eine solche Funktion f beschreiben lassen. Denn nach [1] sind solche Systeme entweder konvergent oder bestimmt divergent. Oszillationen oder gar chaotische Dynamiken sind ausgeschlossen.

4 Anwendungsbeispiele

Die Überlegungen aus den vorangegangenen Abschnitten wurden benutzt, um den Lernalgorithmus für ein rekurrentes NeuroFuzzy-System zu modifizieren und damit die Geschwindigkeit und Erfolgsquote für Lernvorgänge zu verbessern.

4.1 Aufbau der NeuroFuzzy-Systeme

Die benutzten NeuroFuzzy-Systeme haben je eine Eingangsgröße u und eine Zustandsgröße x . In der ersten Neuronenschicht findet die Fuzzifizierung der Größen x und u statt, indem die Zugehörigkeitsfunktionen $\mu_j^x(x)$ und $\mu_p^u(u)$ ausgewertet werden. Die nächste Neuronenschicht bestimmt durch Multiplikation der entsprechenden Zugehörigkeitwerte die Aggregation der Wahrheitswerte für die einzelnen Regeln. Die Implikation wird durch die Gewichte $r_{q,j}$ auf der Verbindungen zum Summationsneuron realisiert. Das Summationsneuron schließlich übernimmt Akkumulation und Defuzzifizierung gleichzeitig. Die Zugehörigkeitsfunktionen werden dabei so gewählt, dass der Nenner, der in der Defuzzifizierungsmethode "Center of Singletons" (CoS) [5] auftritt, den Wert 1 für alle Belegungen von x und u hat. Als Zugehörigkeitsfunktionen werden hier durchgehend normierte Dreiecksfunktionen verwendet. Der schematische Aufbau des NeuroFuzzy-Systems ist in Abbildung 2 zu sehen.

Der Ausgang v wird nun zeitverzögert als Eingangswert x zurückgeführt. Damit entsteht ein dynamisches System, welches in Anlehnung an Gleichung (1) folgende Gleichung realisiert:

$$x(k+1) = v(k) = f(x(k), u(k)), \text{ wobei } f \text{ stetig und } k \in \mathbb{N}. \quad (13)$$

Die Parameter $r_{q,j}$ dieses Systems sollen nun so bestimmt werden, dass es vorgegebene Zeitreihen approximiert. Die Bestimmung erfolgt mit Hilfe des Backpropagation-Algorithmus.

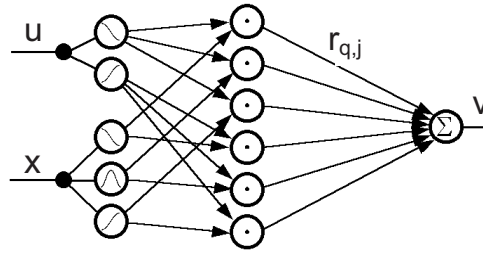


Abbildung 2: Schematischer Aufbau des NeuroFuzzy-Systems.

4.2 Backpropagation-Algorithmus

Die Funktion f läßt sich durch die Singletonpositionen $r_{q,j}$ parametrisieren und wie folgt schreiben:

$$f(x, u) = \sum_{q,j} r_{q,j} \cdot \mu_j^x(x) \cdot \mu_q^u(u). \quad (14)$$

In jedem Zeitschritt wird das Ergebnis $v(k) = f(x(k), u(k))$ mit dem gewünschten Ergebnis $\tilde{v}(k)$ verglichen und der quadratische Fehler $e(k) = (v(k) - \tilde{v}(k))^2$ über die Zeit zu einem Gesamtfehler e_{ges} aufsummiert. Anschließend werden die Singletonpositionen $r_{q,j}$ im l -ten Lernschritt nach der Gleichung

$$r_{q,j}(l+1) = r_{q,j}(l) - h \cdot \frac{\partial e_{ges}(r_{q,j})}{\partial r_{q,j}} \quad (15)$$

verändert. Mit Hilfe der Kettenregel ergibt sich einfach

$$r_{q,j}(l+1) = r_{q,j}(l) - h \cdot \sum_k (v(k) - \tilde{v}(k)) \cdot \mu_j^x(x(k)) \cdot \mu_q^u(u(k)). \quad (16)$$

Es wurde bewußt kein seriell-paralleles Lernen [3] verwendet, bei dem nach jedem Schritt der Zustand $x(k)$ im Modell auf den Wert $\tilde{v}(k)$ des zu identifizierenden Systems gesetzt wird anstatt den Modellausgang $v(k)$ zu benutzen, um das Verfahren dann auf Systeme anwenden zu können, deren Zustand sich zum Teil nicht beobachten läßt.

Die Schrittweite h wurde per SuperSAB-Verfahren adaptiert, weil dies zu besseren Ergebnissen als bei konstanter Schrittweite, bzw. mit einem "momentum term" oder "resilient backpropagation" [6] führte.

4.3 Diesel Turbolader

Für das erste Beispiel lagen Daten des Turboladers eines Dieselmotors vor. Die Einspritzmenge $v(k) = x(k+1)$ im Turbolader hängt stark von der Einspritzmenge $x(k)$ im vorangegangenen Zyklus und der Pedalstellung $u(k)$ ab. Die weiteren Einflußgrößen wurden vernachlässigt und die Daten wurden durch einen linearen Filter leicht geglättet und zwischen 0 und 1 skaliert. Aufgrund des transparenten Ansatzes des NeuroFuzzy-Systems

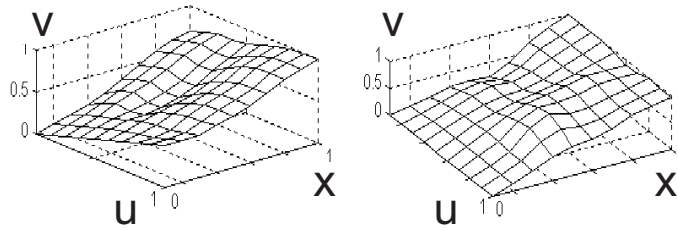


Abbildung 3: Übergangsfunktion $f(x, u)$ vor (links) und nach (rechts) dem Lernprozeß.

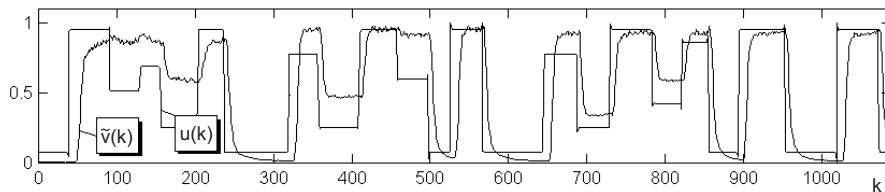


Abbildung 4: Normierte Einspritzmenge \tilde{v} und normierte Pedalstellung u über den Abtastschritten k .

war eine Anfangsinitialisierung der Parameter sehr leicht möglich. Abbildung 3 zeigt die Funktion $f(x, u) = v$ vor und nach dem Lernvorgang. Abbildung 4 zeigt die behandelten Daten und Abbildung 5 die Zeitreihen vor und nach der Anwendung des Lernalgorithmus.

Das gute Ergebnis dieses Lernvorganges liegt darin begründet, dass eine sehr gute Anfangsinitialisierung möglich war. Um den Effekt der Anfangsinitialisierung testen zu können, wurde eine einfacheres Systemverhalten, ein PT_1 -Verhalten zu erlernen und zwar von einer willkürlich initialisierten Regelbasis.

4.4 PT_1 -Verhalten und der modifizierter Backpropagation-Algorithmus

Bei einer willkürlichen Anfangsbelegung der Regelbasis als Ausgangspunkt für das Erlernen einer PT_1 -Verhaltens können prinzipiell konvergentes, oszillatorisches und sogar chaotisches Verhalten auftreten. Man kann die Beispielfunktionen 1, 3 und 4 aus dem ersten Abschnitt durch folgende Wahl der Singletonpositionen erreichen:

	$r_{1,1}$	$r_{1,2}$	$r_{1,3}$
Beispiel 1	0.25	0.5	0.75
Beispiel 3	1	0.5	0
Beispiel 4	0	1	0

Insbesondere können kleine Veränderungen in den Parametern $r_{q,j}$, wie in Abbildung 6 zu sehen ist, zu unterschiedlichem Verhalten führen. Die linke Funktion in dieser Abbildung hat drei Fixpunkte die rechte Funktion dagegen nur einen. Da die Fixpunkte zugleich Ruhelagen sind, hat das linke System zwei stabile und eine instabile Ruhelage, das rechte

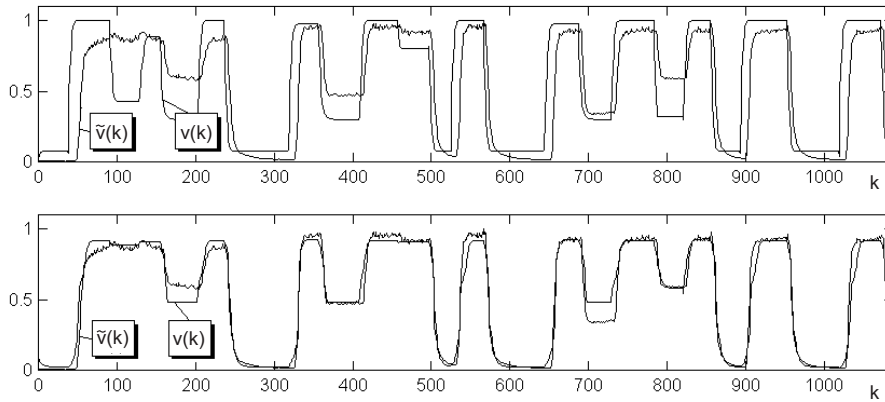


Abbildung 5: Vergleich der Daten \tilde{v} mit den Modellergebnissen v vor (oben) und nach (unten) dem Lernprozeß.

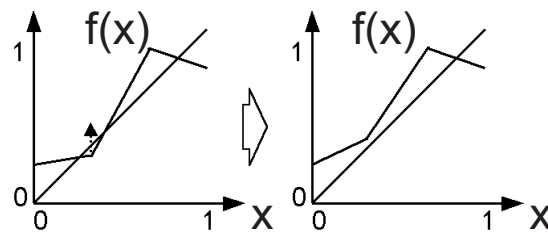


Abbildung 6: Kleine Veränderungen in den Parametern können zu einem sehr unterschiedlichen dynamischen Verhalten führen. Die linke Funktion erzeugt eine instabile und zwei stabile Ruhelagen, die rechte Funktion erzeugt nur eine einzige stabile Ruhelage.

System nur eine einzige, stabile Ruhelage. Für ein PT_1 -Verhalten wird eine eindeutige, stabile Ruhelage erwartet. Weiterhin sollen negative Steigungen ausgeschlossen werden, wie in Abschnitt 3 gezeigt wurde. Um diese Punkte zu erfüllen wird der Raum der zulässigen Singletonpositionen eingeschränkt.

Dazu wird der Eingabewert u so gewählt, dass nur eine der Zugehörigkeitsfunktionen $\mu_q^u(u)$ von Null verschieden ist. Diese Werte werden mit p_q^u bezeichnet. Damit ist die Funktion f in diesem Fall nur von den Singletonpositionen $r_{q,j}$ abhängig, wobei q fest und j beliebig ist. Wählt man nun auch die x -Werte so, dass nur eine der Zugehörigkeitsfunktion $\mu_j^x(x)$ von Null verschieden ist, so ist der Funktionswert gleich der speziellen Singletonposition $r_{q,j}$. Diese x -Werte werden mit p_j^x bezeichnet. Für je zwei benachbarte x -Werte p_j^x und p_{j+1}^x , werden die Singletonpositionen als Punkt in der Ebene dargestellt, wie in Abbildung 7 zu sehen ist. Wenn die Funktion die Winkelhalbierende schneidet, so darf sie nicht fallen. Diese entspricht dem verbotenen Bereich A in Abbildung 7. Ist die Steigung der Geraden beim Schnitt mit der Winkelhalbierenden größer als 1, gibt es einen

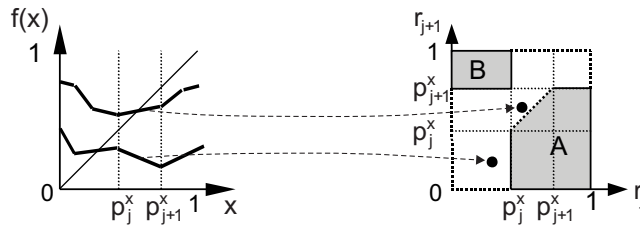


Abbildung 7: Nebeneinanderliegende Parameter p_q^x und p_{q+1}^x dürfen bei abgetasteten Systemen bestimmte Wertepaare nicht annehmen. Diese liegen in den grauen Bereichen A und B.

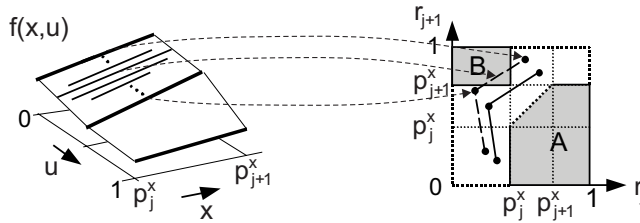


Abbildung 8: Die Verbindungslinie zwischen Parameterpaaren $(r_{q,j}, r_{q+1,j})$ und $(r_{q,j+1}, r_{q+1,j+1})$ dürfen nicht in die Bereiche A und B eindringen.

instabilen Fixpunkt und somit noch mindestens zwei stabile Fixpunkte im Intervall $[0, 1]$. Dieser verbotene Fall ist durch den Bereich B gekennzeichnet. Liegt die Eingabe u nun zwischen zwei Werten p_q^u und p_{q+1}^u , hat das den gleichen Effekt auf die Funktion f , wie die Nutzung von Singletonpositionen, die sich aus der Gewichtung der jeweiligen Singletonpositionen ergeben. Daher liegen die Funktionswerte für p_q^x und p_{q+1}^x in diesem Fall auf der Verbindungsgeraden zwischen $(r_{q,j}, r_{q+1,j})$ und $(r_{q,j+1}, r_{q+1,j+1})$ wie in Abbildung 8 zu sehen ist.

Vor jeder Berechnung der Zeitreihe und dem Vergleich mit den realen Daten werden nun die Singletonpositionen so verschoben, dass keine der Verbindungslinien in die verbotenen Bereiche A oder B durchkreuzen. Da die Verschiebung minimal sein soll, muß somit ein Optimierungsproblem mit Randbedingungen gelöst werden.

Das Ergebnis dieser Modifikation ist in Abbildung 9 zu sehen. Man erkennt, dass der Fehler mit dem modifizierten Algorithmus schneller abfällt und das Ergebnis nach einer festen Zahl von Schritten besser ist. Simulationen bei Systemen, denen Zustand aufgrund einer hinzugefügten nichtumkehrbaren Ausgabefunktion nicht eindeutig festzustellen war, zeigten, dass die Erfolgsquote bei Verwendung des modifizierten Backpropagation-Algorithmus höher ist als bei dem einfachen.

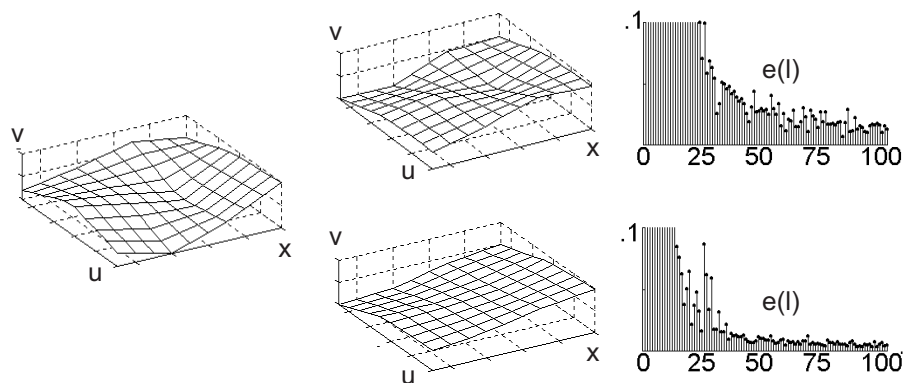


Abbildung 9: Die Übergangsfunktion einer PT_1 -Gliedes vor (links) und nach (Mitte) dem Lernprozeß. Die Übergangsfunktion oben wurde mit dem unmodifizierten Bachpropagation-Algorithmus erreicht, die Übergangsfunktion unten mit dem modifizierten.

5 Zusammenfassung

Abgetastete wertekontinuierliche Systeme sind nur eine Teilmenge der zeitdiskreten Systeme. Sie zeichnen sich durch die Existenz beliebiger Funktionswurzeln aus. Diese Eigenschaft und daraus abgeleitete Eigenschaften können dazu genutzt werden, den Suchraum bei Identifikationsaufgaben einzuschränken, um somit mit einer höheren Erfolgsquote zu erzielen und schneller zu einer Identifikation zu kommen.

Literatur

- [1] R. Kempf, On Ω -limit sets of discrete-time dynamical systems, angenommen zu Veröffentlichung im Journal of Difference Equations and Applications.
- [2] M. Kuczma, B. Choczewski, R. Ger, Iterative Functional Equations, Cambridge Univ. Press, Cambridge, 1990
- [3] O. Nelles, S. Ernst, R. Isermann, Neuronale Netze zur Identifikation nichtlinearer, dynamischer Systeme: Ein Überblick, at 45 (1997) 6, S. 251-262
- [4] P. Protzel et al., Anwendung und Entwicklung Neuronaler Verfahren zur Autonomen Prozess-Steuerung, Abschlussbericht des BMBF Verbundprojekt 01 IN 505 B, 1999
- [5] Fuzzy logic and fuzzy control - Terms and definitions, VDI/VDE-Richtlinien, VDI/VDE 3550 Blatt 2, Beuth Verlag GmbH, Berlin, 2000
- [6] A. Zell, Simulation neuronaler Netze, Oldenburg Verlag, München, 1997

An Evolutionary Algorithm for Semi-Supervised Fuzzy Classification

Aljoscha Klose

Department of Knowledge Processing and Language Engineering
University of Magdeburg, School of Computer Science
D-39106 Magdeburg, Germany
Tel: (0391) 6712700, Fax: (0391) 6712018
aljoscha.klose@cs.uni-magdeburg.de

1 Introduction

Fuzzy *if-then* rules have become a popular basis for classifiers. The use of linguistic variables is close to human thinking and thus eases the readability and interpretability of the rule base. Automatic induction of fuzzy rules from data is therefore an interesting topic. There is a variety of techniques that have been proposed for this task, e.g. neuro-fuzzy systems, evolutionary algorithms, or decision tree based methods. These approaches have in common that they solve a *supervised learning* task, i.e. they induce the rules from a database of examples with known class labels.

Getting the labels for the data is, however, a problem in many practical applications. Often, training samples have to be labeled manually, which can be a tedious task when there are many objects. Typical examples of such domains are image processing or document retrieval. If only unlabeled data is available for learning, i.e. if the class labels of all sample data are unknown, *unsupervised learning* algorithms can be used. This is a task for cluster analysis, which tries to find groups of similar objects in the data. Fuzzy extensions of cluster analysis represent the clusters by multidimensional fuzzy sets and can be used to extract fuzzy if-then rules from data. [6]. The main drawback of unsupervised learning is that it strongly depends on the given distance measures. It has usually problems in finding the right number of clusters. Usually, we cannot guarantee that the clusters in the data space correspond to meaningful classes of the objects.

As unsupervised learning without any guidance does in many cases not yield satisfactory results, in practice supervised learning is much more common. If it is not possible to label all objects, one usually confines the examples to a certain — hopefully representative — fraction of the data and leaves the unlabeled data aside. However, though unlabeled, the additional data might still bear valuable information on the true distribution of the objects in the input space. Thus, it is an appealing idea to use the remaining, otherwise discarded, data to support the learning of a classifier. Such *semi-supervised* learning could thus be expected to yield better generalization on unseen data with much less labeled examples.

There have been some efforts to combine supervised and unsupervised learning. In Sect. 2 we briefly review previous work on semi-supervised learning. However, these approaches deal with complex models like e.g. multi-dimensional cluster prototypes or support vector machines that are not easily interpretable and thus do give little insight into the data. Little has been done on the semi-supervised induction of more

convenient models like fuzzy rules from data. In this paper we present our work to enable fuzzy classification to deal with partially labeled data.

Our semi-supervised fuzzy classification system is based on an evolutionary algorithm. In Sect. 3 we describe how we incorporated labeled and unlabeled data in the definition of the fitness function. Details of the implementation are given in Sect. 4. The results of applying the classifier on example data are presented in Sect. 4.1. Sect. 5 summarizes our results.

2 Semi-Supervised Learning

The classical paradigms of machine learning can be separated into supervised and unsupervised approaches. Unsupervised learning techniques like cluster analysis seek for inherent structure in the distribution of the data, whereas supervised techniques try to generalize the regularities given by some sample data.

There have been some proposals how to combine and thus exploit the benefits of both approaches. In most publications, Pedrycz's approach from 1985 [11] is cited as the first work in the area of semi-supervised clustering. Since then, there have been a number of different ideas how to combine the information of labeled and unlabeled data in an semi-supervised learning algorithm. The underlying ideas include:

- Use the class information of the labeled points as seed points, and incrementally label the unlabeled points (e.g. [14]),
- learn prototypes from the labeled data and adjust their exact positions to fit labeled *and* unlabeled data (e.g. [2, 11]),
- learn from the labeled points, use low densities between classes to optimize class borders (e.g. [1, 4]),
- use the abundance of unlabeled data for more reliable estimation of the probability density function of all the data in the input space (e.g. [13]).

Successful applications of semi-supervised approaches are reported e.g. in the field of image processing [2] or text classification [8].

3 Measuring Rule Set Quality for Partially Labeled Data

Given a class of classifier models, the task of any learning algorithm is to choose an instance that maximizes some objective function. We will define a measure based on the Minimum Description Length Principle that allows to assess the quality of rule bases with varying numbers of rules in the context of partially labeled data.

The Minimum Description Length Principle (MDL) offers a well-founded method to compare different hypotheses about structure in data [12]. Generally, when models of different complexity are used, usually the most complex model (i.e. that with the most degrees of freedom) fits the data best. If competitive models are only

compared by means of an error measure, the more complex models tend to outwit the others. This can lead to over-fitting. Therefore, penalty terms are introduced that punish complexity. These are often subjectively chosen heuristics.

MDL tries to give some more objective means. The idea is that the data has to be transmitted from an (imaginary) sender to an (imaginary) receiver. Structure in the data can be used for more efficient codes that result in shorter messages. However, both sender and transmitter need to know the encoding scheme of the data. Thus the message is compound by first transmitting the coding scheme, and then the data, encoded using this scheme. Complex models need a longer coding scheme, as more free parameters have to be transmitted. However, the resulting data part of the message will usually be shorter. The model with the shortest overall message length is assumed to fit the data best and is chosen. MDL is equivalent to maximum likelihood for the estimation of models with a fixed number of free parameters, but additionally offers a possibility to compare objectively between models of different complexity.

To apply the MDL principle to our problem, we have to define an appropriate encoding scheme. The message consists of the following parts:

- **the rule base (the code)**, e.g. the number of dimensions, the number of rules, the fuzzy sets used in a rule. This information is then used to encode
- **the data tuples (the data itself)**, e.g. the index of the rule that is used to encode this tuple, the exact values of the tuple using a rule specific code, and the class labels.

According to Shannon's coding theorem, the lengths in bits of parts of a message are calculated as $-\log p_i$, where p_i is the probability of an instantiation i within all possible alternatives over some appropriately chosen probability distribution. To illustrate this, we present the equations for the message length for data tuples below.

If the t -norm \top_{prod} is used and some restrictions are placed on the fuzzy sets and rule weights, (neuro-) fuzzy classification systems can be seen as Naive Bayes classifiers [10]. We can use this interpretation of the fuzzy membership values as probability densities to transmit the tuple coordinates. Let \mathbf{x} be a tuple, r the rule which is used to encode it, i.e. the rule with the highest activation, and w_r the appropriate normalizing rule weight. Then the part of the message that transmits the coordinates of the d dimensions of a tuple has a length of

$$l(\mathbf{x}) = -\log_2 w_r - \sum_{i=1}^d \log_2 \mu_{r,i}(x_i).$$

This part is used for encoding both, labeled and unlabeled tuples. The closer a tuple lies to the center of the membership functions of its rule (i.e. the better the rule is adapted to the data), the higher the probabilities and thus the shorter the code length is.

Additionally, we have to transmit the class labels of the tuples. We associate each rule with the majority class of the tuples covered by it. Thus we can use the (already

transmitted) information which rule is used to encode a tuple. However, there might be tuples of several classes covered by one rule. Therefore, we have also to transmit the exceptions from the rule. We have to encode the class distribution in a rule and the explicit class labels for the tuples. Let us assume N_r tuples covered by a rule r , with $N_{r,c}$ tuples of class c . Let n_c be the number classes in the dataset. In that part of the message the $N_{r,?}$ unlabeled points play a role. We consider them as belonging to the majority class c_{\max} in that rule and thus transmit this class (the most probable class yields the shortest message length). For the encoding of the class information for all tuples covered by one rule r we thus get a length of

$$l(r) = \log_2 \frac{(N_r + n_c - 1)!}{N_r!(n_c - 1)!} + \log_2 \frac{N_r!}{\left(\prod_{c=1, c \neq c_{\max}}^{n_c} N_{r,c}!\right)(N_{r,c_{\max}} + N_{r,?})!}.$$

Misclassified labeled tuples make that part of the message longer, as the probability distribution of a rule becomes more heterogeneous.

This MDL based measure can be seen as an example of the second class of semi-supervised approaches, as we combine an error measure for the labeled data and a measure of adaptation to all data. One problem of this measure is that the relation of the lengths of the code part and the data part of the message depends on the number of tuples and the number of dimensions. It can therefore be important to weight the individual parts of the message. This can be interpreted as adjusting the precision of the transmitted tuple coordinates.

4 Implementation Details

Unfortunately, our MDL based measure does not directly allow optimization methods like gradient descent. In previous work we already successfully implemented simple evolutionary optimization of fuzzy sets [7]. Therefore, we extended this approach and implemented an evolutionary optimization for the fuzzy rule learning. We can then directly use the MDL measure as a fitness function.

A fuzzy classification system [6] consists of a fuzzy rule base and an inference engine, which evaluates the rule base for the datum to be classified. The rule base contains a set of fuzzy classification rules of the form:

if x_1 is μ_1 **and** x_2 is μ_2 **and** \dots **and** x_n is μ_n
then pattern (x_1, x_2, \dots, x_n) belongs to class c ,

where the μ_k are fuzzy sets describing regions in the corresponding dimensions. We use Gaussian functions, and additionally allow “don’t cares” in the antecedents, i.e. fuzzy sets that are always 1. Especially in higher dimensional datasets “don’t cares” are important to make the fuzzy rules more readable, the rule bases more compact and improve generalization ability.

Fuzzy membership functions in fuzzy rule base systems can be defined either locally or globally. Globally means that a set of membership functions is defined for the rule base, and that they are referred to by the antecedents of the rules. In the local case, each rule might have its own fuzzy set definitions. Globally partitioning the

axes into fuzzy sets makes it easier to assign linguistic labels to them. The global definition is thus more commonly used. However, in our implementation we chose to use locally defined fuzzy sets for two reasons. First, locally defined fuzzy sets allow more degrees of freedom, where rules with globally defined fuzzy sets often yield only axis parallel separation borders between classes (see, e.g., [9]). Secondly, the rule-wise definition eases the genetic operations like crossover and thus enables synchronous optimization of rules and fuzzy sets. By applying repair mechanisms we try to ensure linguistic interpretability of the resulting fuzzy sets.

In the implementation of the evolutionary learning we roughly follow the “Pittsburgh-style” approach described in [3]. Opposed to “Michigan-style” approaches, one chromosome represents a complete rule base (instead of cooperating individual rules).

As selection operator we chose the tournament selection [5], i.e. we randomly choose two chromosomes from the pool and take the fitter of the two. This operator is more robust compared to fitness proportional approaches, as it does not depend on the scaling of the fitness function, and computationally cheaper than rank based selection, which implies sorting the chromosomes by fitness.

According to the *building block theorem*, a good recombination operator should produce valid offsprings that preserve good partial solution from their parents. In our case, reasonable building blocks are obviously the rules, or combinations of them. In a former implementation we used the so-called *one-point ordered crossover* [3]. It chooses a split point on every axis and thus separates the rule space into two subspaces. In the child’s rule base we copy the rules of one parent falling in one, and of the other parent falling in the other subspace. The resulting rule base of the child chromosome is covered with rules from the whole rule space, and neighboring rules have a higher probability to survive together. The number of rules may change by this procedure. However, the subspaces can show some peculiarities if dimensionality is high. We thus perform the split differently by defining a random hyperplane and choose the rules that have their center on one or the other side.

The mutation operator changes structure as well as parameters of the fuzzy sets. Thus rules might be deleted or added, dimensions might be set to “don’t care”, and the fuzzy set parameters are overlaid with small Gaussian distributed noise.

To enable linguistic interpretation of the fuzzy sets after learning, a minimum requirement is that they are ordered. In that way we can assign the usual labels (e.g. *small, medium, large*) or mixtures of them (e.g. *small to medium*). We implemented a repair mechanism that checks the fuzzy sets of every dimension after crossover and mutation. If the constraints are violated, the fuzzy sets are appropriately shifted to restore them.

4.1 Empirical Results

In the following we present the results of applying our approach to the *Wisconsin Breast Cancer* dataset, that can be found in the UCI machine learning repository. This dataset contains 699 cases distributed into two classes (benign and malign), from which we excluded 16 cases with missing values. We performed 5 random splits into training and test data. From each of the training sets we randomly chose

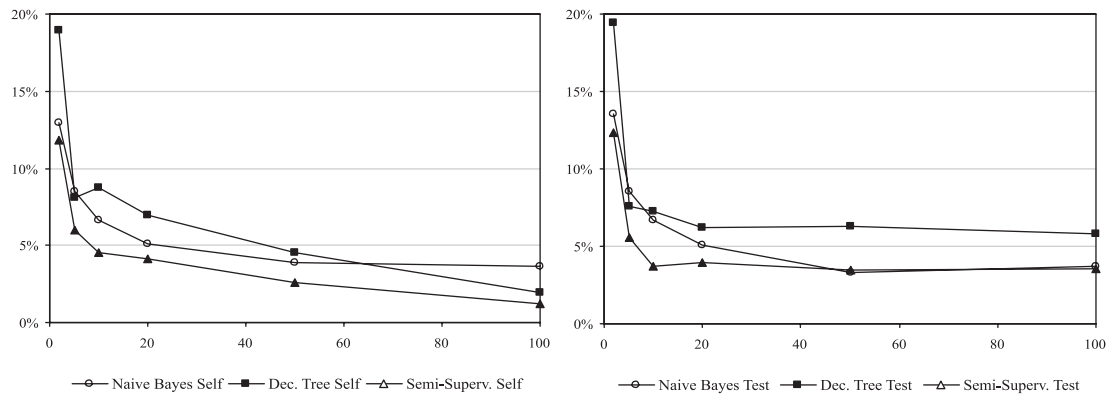


Figure 1: The error over the percentage of training data used for classification. The figures show the averaged percentage of misclassifications (a) on the training data, (b) on the test data.

different fractions (2%, 5%, 10%, 20% and 50%) and deleted the class labels of the remaining tuples. We presented only the chosen fraction to two fully supervised algorithms, i.e. naive Bayes classification and a decision tree inducer (an implementation of C4.5). We compared this to our algorithm that has been trained by the chosen labeled fraction and additionally the unlabeled remainder. Fig. 1 shows the results for the different fractions (averaged over the 5 splits) on the training data (left) and applied to the test data (right).

A first observation is that the decision trees did not perform well on this dataset. The difference between test and training data is quite big. Although we pruned the decision trees, they obviously seem to overfit the data. Moreover, they perform extremely bad for the small sample sizes. With the naive Bayes classifier the results on training and test data are very close, probably due to the rather low complexity and robustness of the classifier. With a growing percentage of labeled examples, the results constantly improve until 50% of the labels are used for learning. Our semi-supervised approach has a slight tendency to overfitting when compared to the naive Bayes classifier. The results on the training data are even better than that of the decision trees. However, we also get very good results on the test data. The results are similar to those of the naive Bayes classifier. However, by using the unlabeled data, these good results are already achieved with only 10% of the labels.

5 Conclusions

The state of the art in data mining is dominated by supervised approaches. However, in many real-world problems the assignment of labels for all objects is a severe problem and thus calls for semi-supervised methods.

We presented an evolutionary algorithm to induce fuzzy rules that allows to exploit labeled and unlabeled training data. The approach succeeded to outperform naive Bayes and decision trees, two purely supervised algorithms, in our examples.

References

- [1] K. P. Bennett and A. Demiriz. Semi-supervised support vector machines. In D. A. Cohn, M. S. Kearns, and S. A. Solla, editors, *Advances in Neural Information processing Systems*, pages 368–374, MIT Press, Cambridge, MA, 1998.
- [2] A. M. Bensaid, L. O. Hall, J. C. Bezdek, and L. P. Clarke. Partially supervised clustering for image segmentation. *Pattern Recognition*, 29(5):859–871, 1996.
- [3] B. Carse, T. C. Fogarty, and A. Munro. Evolving fuzzy rule based controllers using genetic algorithms. *Fuzzy Sets and Systems*, 80:273–293, 1996.
- [4] G. Fung and O. L. Mangasarian. Semi-Supervised Support Vector Machines for Unlabeled Data Classification. *Data Mining Institute, Technical Report 99-05*, 1999.
- [5] D. E. Goldberg and K. Deb. A comparative analysis of selection schemes used in genetic algorithms. In G. Rawlins, editor, *Foundations of Genetic Algorithms*. Morgan Kaufmann, 1991.
- [6] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis*. Kluwer Verlag, 1998.
- [7] A. Klose, R. Kruse, K. Schulz, and U. Thönnessen. Controlling asymmetric errors in neuro-fuzzy classification. In *Proc. ACM SAC'00*, pages 505–509, ACM Press, 2000.
- [8] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine learning*, 39(2/32), 103–134, 2000.
- [9] A. Nürnberger, A. Klose, and R. Kruse. Effects of Antecedent Pruning in Fuzzy Classification Systems In *Proc. of the Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies (KES 2000)*, pages 154–157, IEEE, Piscataway, 2000.
- [10] A. Nürnberger, C. Borgelt, and A. Klose. Improving naive bayes classifiers using neuro-fuzzy learning. In *Proc. ICONIP'99*, pages 154–159, Perth, Australia, 1999.
- [11] W. Pedrycz. Algorithms of fuzzy clustering with partial supervision. *Pattern Recognition Letters*, 3:13–20, 1985.
- [12] J. Rissanen. A universal prior for integers and estimation by minimum description length. *Annals of Statistics*, 11:416–431, 1983.
- [13] A. Skabar. Augmenting supervised neural classifier training using a corpus of unlabeled data. In *Proceedings of 25th German Conference on Artificial Intelligence (KI-2002)*, pages 174–185, Aachen, Germany, 2002.
- [14] A. Verikas, A. Gelzinis, and K. Malmquist. Using Unlabeled Data for Learning Classification Problems. In L. C. Jain and J. Kacprzyk, editors, *New Learning Paradigms in Soft Computing*, pages 368–403, Physica, Heidelberg, 2002.

Entwicklung einer Fuzzy-Control-Bibliothek in Modelica

Felix Felgner

Lehrstuhl für Automatisierungstechnik (Fachbereich EIT), Universität Kaiserslautern
Erwin-Schrödinger-Straße 12, D 67663 Kaiserslautern

Tel. (0631) 205-4208

Fax (0631) 205-4462

E-Mail: felgner@eit.uni-kl.de

1 Einleitung

Fuzzy-Control ist gemeinhin bekannt als ein Automatisierungsprinzip, dessen Entwurfsverfahren im Gegensatz zur Vorgehensweise der klassischen Regelungstechnik ohne ein mathematisches Prozessmodell auskommen. Voraussetzung hierfür ist jedoch, dass das zum Formulieren des Regelwerkes notwendige Expertenwissen bereits vorhanden ist. Es gibt Situationen, in denen der Regelungsingenieur diese Wissensbasis selbst finden muss: z.B. bei der Inbetriebnahme einer völlig neuen Anlage oder wenn eine vorhandene Regelung durch einen Fuzzy-Controller ersetzt werden soll oder wenn der Regelungsentwurf schon in der Entwicklung eines neuen Produktes erfolgt. In solchen Fällen ist die Simulation ein effektives Mittel, um eine komplizierte Prozessdynamik näher kennenzulernen.

Die Simulationssprache *Modelica* (<http://www.modelica.org>) bietet für derartige Modellierungsaufgaben bessere Voraussetzungen als die bislang noch weit verbreiteten signalflussorientierten Modellierungstechniken. Denn die objektorientierte Beschreibungsweise ohne Zwang zur Berechnungskausalität erlaubt die Definition von Teilsystemen, deren Schnittstellen sich nach den physischen Gegebenheiten des realen Prozesses richten. Um die Vorteile der *Modelica*-Modellierung auch beim Entwurf von Fuzzy-Controllern nutzen können, wurde am Lehrstuhl für Automatisierungstechnik eine Fuzzy-Control-Bibliothek in *Modelica* entwickelt. Diese ermöglicht es dem *Modelica*-Anwender, mit geringem Aufwand einen Fuzzy-Controller zu einem vorhandenen Prozessmodell zu entwerfen und dabei verschiedene Regelbasen, Fuzzy-Sets sowie Methoden für Inferenz und Defuzzifizierung zu testen.

2 Übersicht über die Bibliothek

2.1 Allgemeines

Die Modelle der Fuzzy-Control-Bibliothek wurden im Hinblick auf ihre Anwendung in der Simulationsumgebung des Programms *DYMOLA* (ab *Version 4.1a*) der Firma *Dynasim* (<http://www.dynasim.se>) entwickelt. Dies betrifft zum einen ihren graphischen Aufbau, soweit dieser für die Bedienung von Bedeutung ist. Zum anderen entspricht die Nutzung der FC-Modelle dem für *DYMOLA*-Anwender gewohnten Umgang mit gebrauchsfertigen *Modelica*-Modellen. Die FC-Modelle sind daher in einer strukturierten Bibliothek zusammengefasst, aus der der Anwender einzelne Komponenten entnimmt. Durch Laden der Bibliothek in ein beliebiges Verzeichnis stehen damit alle

Funktionen der FC-Modelle zur Verfügung, d.h. die Implementierung kommt ohne Änderungen an der installierten *DYMOLA*-Version aus.

In den folgenden Kapiteln wird ein Überblick über die Bibliothek gegeben. Dieser eignet sich als Bedienungsanleitung für den Aufbau benutzerdefinierter Fuzzy-Controller. Die neue Bibliothek („Package“ in der *Modelica*-Nomenklatur) existiert in zwei eigenständigen Versionen, die sich in der Bedienungslogik der Regeleingabe unterscheiden. Genaue Anleitungen zur Eingabe der Regeln finden sich in Kapitel 2.5.

2.2 Modellkomponenten

In **Version I** (Bild 2.1) erfolgt die Regeleingabe durch die Erstellung eines Schaltplanes, dessen Aufbau an Schaltungen aus der binären Logik erinnert. Es stehen Ein- und Ausgangsblöcke mit drei und fünf Zugehörigkeitsfunktionen zur Verfügung; in den Regelprämissen können UND- und ODER-Verknüpfungen sowie Negationen einzelner linguistischer Werte ausgedrückt werden; jeder Regel kann ein individuelles Gewicht zwischen 0 und 1 verliehen werden; für die Defuzzifizierung stehen die Höhenmethode und die erweiterte Schwerpunktmethode zur Verfügung.

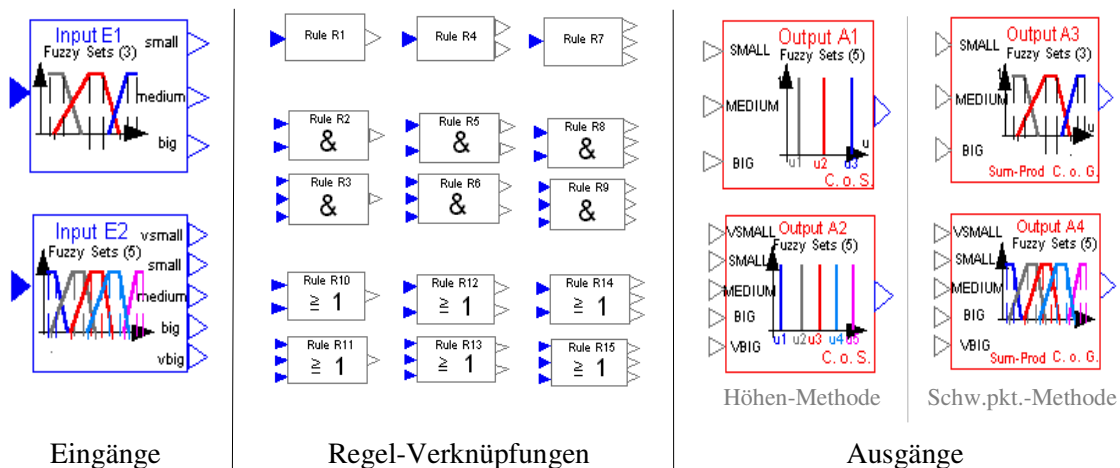


Bild 2.1: Komponenten der Version I

Version II (Bild 2.2) erlaubt eine Regeleingabe in Textform als Parameter in einen zentralen Regelblock, der eine Übersetzung der Regeln vornimmt. Zusätzlich zu Version I ist die Defuzzifizierung auch mit der Regelbasisfehler-Methode (vgl. [1]) möglich. Außerdem kann bei der Schwerpunktmethode zwischen drei Inferenzarten gewählt werden. Einschränkend gegenüber Version I sind in den Regelprämissen nur UND-Verknüpfungen erlaubt.

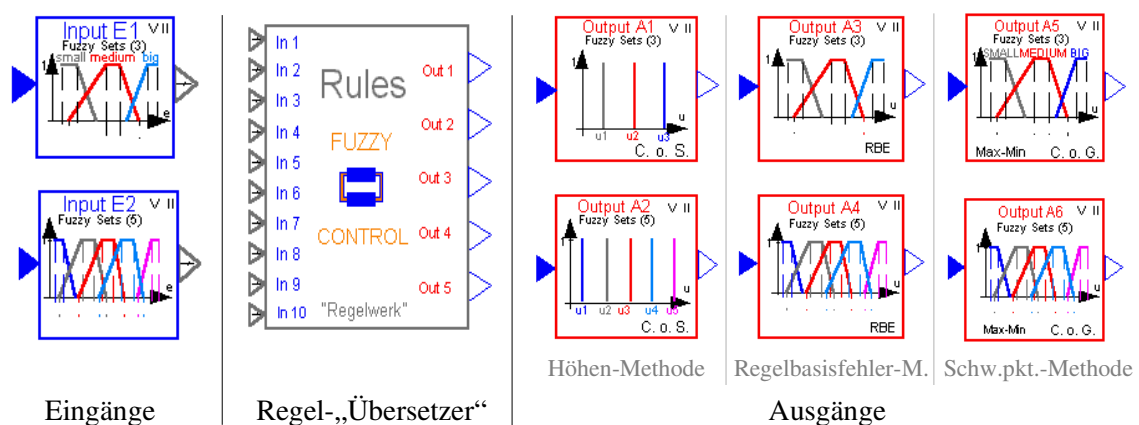


Bild 2.2: Komponenten der Version II

2.3 Wahl der Methode

Jeder MIMO-Fuzzy-Controller (MIMO: multiple input, multiple output) ist zerlegbar in je einen MISO-Fuzzy-Controller (MISO: multiple input, single output) pro Ausgang. Die Teil-Controller arbeiten dabei völlig unabhängig voneinander. Aus diesem Grunde war es naheliegend, die Methoden für Inferenz und Defuzzifizierung den Ausgangsblöcken zuzuordnen. Dabei können für unterschiedliche Ausgänge durchaus unterschiedliche Methoden¹ gewählt werden. Abhängig von der Defuzzifizierungsmethode gibt es weitere Einstellungsmöglichkeiten in der Parametertafel des gewählten Ausgangsblockes (Tabelle 2.1).

Defuzzifizierung	Version I		Version II	
	Inferenz	Zusätzliche Einstellungen	Inferenz	Zusätzliche Einstellungen
Erweiterte Schwerpunkt-methode (C. o. G.)	Summe-Produkt (Sum-Prod)	N Anzahl der Integrations-schritte	Summe-Produkt (Sum-Prod) Maximum-Minimum (Max-Min) Maximum-Produkt (Max-Prod)	N Anzahl der Integrations-schritte
Höhenmethode (C. o. S.)	(keine Einstell.)	(keine Einstell.)	(keine Einstell.)	(keine Einstell.)
Regelbasis-fehlermethode (RBE), nur Version II	—	—	siehe [1] (keine Wahlmögl.)	c Kompromiss-fähigkeit c_0 Kompromiss bei nicht eindeutigen globalen Minimum des RBF

Tabelle 2.1: Implementierte Methoden sowie wichtige Einstellungsmöglichkeiten

2.4 Definition der Fuzzy-Sets

Beim Aufbau eines *Modelica*-Fuzzy-Controllers hat der Anwender für jede Eingangsgröße einen Eingangsblock und für jede Ausgangsgröße einen Ausgangsblock aus der Bibliothek zu wählen bzw., genauer gesagt, zu instanziiieren. Dabei stehen Ein- und Ausgangsblöcke mit drei und mit fünf Fuzzy-Sets (Zugehörigkeitsfunktionen) zur Verfügung. Im Falle von drei Fuzzy-Sets handelt es sich um zwei rampenförmige Zugehörigkeitsfunktionen an den Rändern und eine trapezförmige Zugehörigkeitsfunktion im mittleren Bereich. Bei fünf Fuzzy-Sets gibt es für den mittleren Wertebereich zwei weitere trapezförmige Zugehörigkeitsfunktionen (Bild 2.3). Die linguistischen Werte bekommen hier folgende (feste) Bezeichnungen:

- „sehr klein“ / *vsmall*,
- „klein“ / *small*,
- „mittel“ / *medium*,

¹ Unter dem Begriff „Methode“ ohne weitere Zusätze werden die Methoden für Inferenz (Inferenzkern und Akkumulation) und Defuzzifizierung zusammengefasst. Die Regelbasisfehler-Methode ist eine spezielle Methode, die 1994 am Lehrstuhl für Automatisierungstechnik der Universität Kaiserslautern entwickelt wurde [1].

- „groß“ / big,
- „sehr groß“ / vbig.

Zur Anpassung an eine individuelle Regelungsaufgabe können Minimalwert und Maximalwert der jeweiligen linguistischen Variablen eingegeben werden sowie die Knickstellen der Zugehörigkeitsfunktionen. Die einzige Einschränkung bei der Parametrierung ist die, dass die fallenden bzw. steigenden Geradenstücke nicht unendlich steil sein dürfen. Hinsichtlich der Überlappung der einzelnen Zugehörigkeitsfunktionen gibt es keine Einschränkungen. Auch können die Trapez-Funktionen so parametrierung werden, dass sie zu Dreiecken werden. Singletons lassen sich approximieren durch sehr schmale Trapeze oder Rampen.

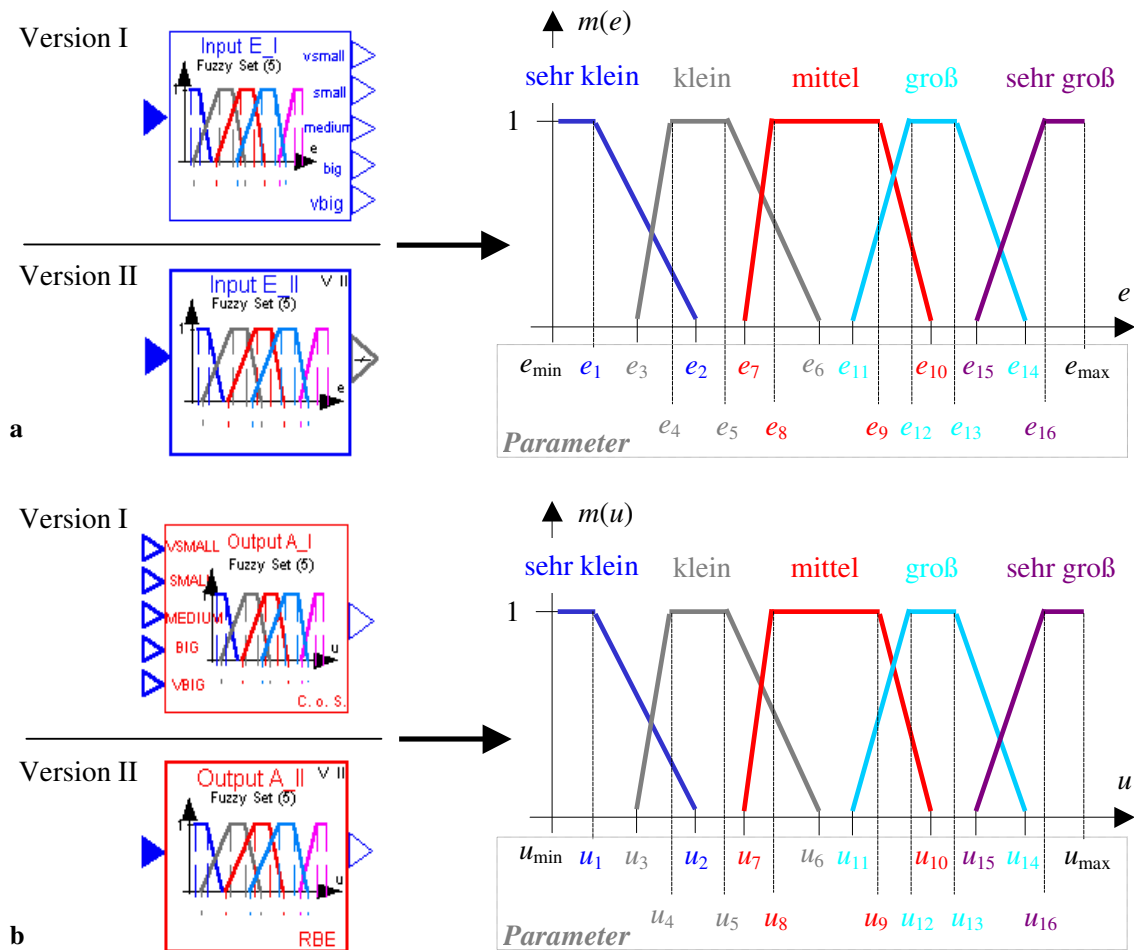


Bild 2.3: Zur Parametrierung der Fuzzy-Sets (**a** Eingangsgröße, **b** Ausgangsgröße)

Die Parametrierung der Fuzzy-Sets in den Eingangsblöcken und in den Ausgangsblöcken erfolgt völlig gleich. Eine Ausnahme stellen Ausgänge dar, in denen die Defuzzifizierung nach der Höhenmethode erfolgt. In diesem Fall sind nur Singletons vorgesehen, da die Höhenmethode (im Englischen „centre of singletons“) den genauen Verlauf der Zugehörigkeitsfunktionen nicht berücksichtigt.

2.5 Eingabe des Regelwerkes

Bei der Konzeption der Fuzzy-Control-Bibliothek mit den in *DYMOLA/Modelica* verfügbaren Mitteln stellt die Bedienungslogik der Regel-Eingabe den entscheidenden Schritt dar. Die Auswertung einer Regel benötigt Informationen über die in ihr verknüpften Eingangs- und Ausgangsgrößen. Dies erfordert geeignete Verknüpfungen zwi-

schen Eingangs- und Ausgangsblöcken, die der Anwender auf der graphischen Benutzeroberfläche von *DYMOLA* herzustellen hat. Entscheidend ist dabei, dass diese Verknüpfungen einem einfachen Prinzip folgen und auch ein Anwender, der nicht genau weiß, welche Signale dadurch ausgetauscht werden, sie intuitiv in Zusammenhang mit dem Formulieren der Regel bringt.

Es wurden zwei unterschiedliche Konzepte zur Regel-Eingabe entwickelt:

A Version I: Verknüpfungsbasierte Regel-Eingabe

Zur Auswertung einer Regel genügt es, nur die Eingänge und Ausgänge miteinander zu verknüpfen, zwischen denen durch die Regel eine Beziehung hergestellt wird. Die Grundidee der ersten Version ist, dass bereits in dieser graphisch erkennbaren Verbindung die Eingabe der Regel besteht. Ein Eingangsblock besitzt zu jedem seiner linguistischen Werte (lw) einen Ausgangs-Cut und ein Ausgangsblock zu jedem linguistischen Wert (LW) einen Eingangs-Cut. Zusätzlich wird für jede Regel ein Regelblock benötigt, der in der Verknüpfung zwischen den Eingängen und Ausgängen liegt.

Um also die Regel

„WENN Eingangsgröße $e_\alpha = lw\ n$ UND Eingangsgröße $e_\beta = lw\ m$,
DANN Ausgangsgröße $u_\mu = LW\ j$ UND Ausgangsgröße $u_\nu = LW\ k$ “

eingzugeben, verbindet man den Cut $lw\ n$ des Eingangsblocks α sowie den Cut $lw\ m$ des Eingangsblocks β jeweils mit einem eigenen Eingangs-Cut eines Regelblockes und je

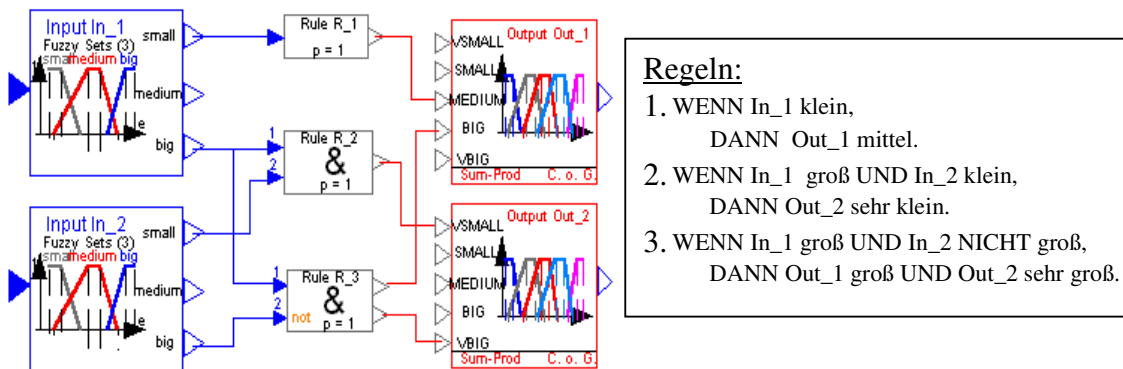


Bild 2.4: Modelica-Fuzzy-Controller nach Version I

einen Ausgangs-Cut des Regelblockes mit dem Cut $LW\ j$ des Ausgangsblocks μ sowie dem Cut $LW\ j$ des Ausgangsblocks ν . In der Bibliothek stehen hierfür Regelblöcke mit unterschiedlich vielen Eingängen und Ausgängen zur Verfügung. Weiterhin kann zwischen UND- sowie ODER-Verknüpfungen der Eingangsgrößen gewählt werden. Die Ausgangsgrößen erfahren hingegen immer eine UND-Verknüpfung, da eine ODER-Verknüpfung keine eindeutige Auswertung der Regel erlauben würde. Ein übersichtliches Beispiel zu Version I zeigt Bild 2.4.

Zur besseren Übersicht kann bei einer größeren Anzahl von Regeln für jeden Ausgang ein getrennter Fuzzy-Controller aufgebaut werden. Zwischen den einzelnen Controllern wird keine Verknüpfung benötigt, da die Regelauswertung für jede Ausgangsgröße getrennt im jeweiligen Ausgangsblock und in den relevanten Regelblöcken erfolgt. Darüber hinaus können den Verbindungslinien verschiedene Farben gegeben werden.

B Version II: Textbasierte Regel-Eingabe

Die Regel-Eingabe nach der ersten Version verlangt vom Benutzer für größere Anzahlen von Regeln eine gewisse Disziplin bei der Anordnung der Regelblöcke und der Verbindungslinien zu den Eingangs- und Ausgangsblöcken. Andernfalls lassen sich die Regeln zwar leicht eingeben, im Nachhinein aber nur noch mühsam lesen.

Die zweite Version sorgt hier für mehr Übersichtlichkeit. Sie benötigt nur einen Regelblock für bis zu 50 Regeln, zehn Eingangsgrößen und fünf Ausgangsgrößen. Auch die Zahl der Verknüpfungslinien reduziert sich deutlich, da in dieser Version jeder Eingangs- und jeder Ausgangsblock nur noch mit je einer Verknüpfung an einen seiner Nummer zugeordneten Cut des zentralen Regelblocks anzuschließen ist. Dass bei weniger als zehn Eingangsblöcken einige Eingangs-Cuts des Regelblockes frei bleiben, stellt für die Simulation kein Problem dar, weil die Variablen dieser Cuts als Flussvariablen deklariert sind. Wenn ein solcher Cut an keinen anderen angeschlossen ist, erhalten die Flussvariablen den Wert 0, der im vorliegenden Fall vom Regelblock richtig weiterverarbeitet wird. Bild 2.5 zeigt den zum obigen Beispiel aus Bild 2.4 äquivalenten FC-Aufbau in Version II.

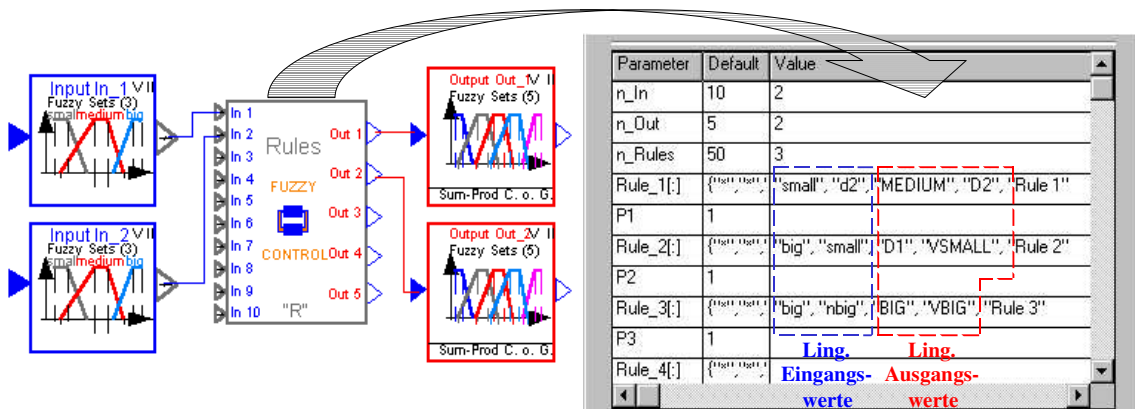


Bild 2.5: Modelica-Fuzzy-Controller nach Version II (gleiche Regeln wie in Bild 2.4)

3 Implementierung des FC-Algorithmus in Modelica

3.1 Die Modellierungssprache Modelica

Die Sprache *Modelica* ist in besonderer Weise geeignet für die Modellierung dynamischer Systeme physikalischer Natur. Teilsysteme werden durch Vertreter unterschiedlicher Modellklassen dargestellt, die über ihre Schnittstellen (Cuts) miteinander verknüpft sind. Die FC-Bibliothek macht überwiegend Gebrauch von kausalen, signalflossorientierten Verknüpfungen, die man z.B. von *Matlab/Simulink*-Modellen kennt. Es gibt jedoch auch Schnittstellen mit sog. Flussvariablen, die es erlauben, an einen (sinngemäßen) Eingangs-Cut beliebig viele (auch keine) Verknüpfungslinien anzuschließen. Eine analoge Übertragung der FC-Komponenten in *Simulink*-Blockschaltbilder ist daher nicht möglich.

3.2 Der FC-Algorithmus

Auf eine allgemeine Einführung in die Theorie von Fuzzy-Control wird hier verzichtet. Ausführliche Darstellungen sind in entsprechenden Lehrbüchern zu finden (z.B. [2], [3]). Auch eine detaillierte Erläuterung der implementierten *Modelica*-Modelle würde den Rahmen dieses Beitrages übersteigen. An dieser Stelle kann lediglich die

grundsätzliche Struktur der *Modelica*-Implementierung skizziert werden. Im Folgenden wird stets von einem *Mamdani*-FC ausgegangen.

Die Verarbeitung der Eingangsgrößen e_1, \dots, e_N des FC zu den Ausgangsgrößen u_1, \dots, u_M , die alle einem kontinuierlichen Wertebereich entstammen, geschieht gewöhnlich in fünf Stufen (vgl. Bild 3.1). Bild 3.1 zeigt auch deren grundsätzlich Zuordnung zu den drei Typen der Funktionsblöcke (Eingangs-, Ausgangs- und Regelblöcke), die in der *Modelica*-Implementierung benötigt werden.

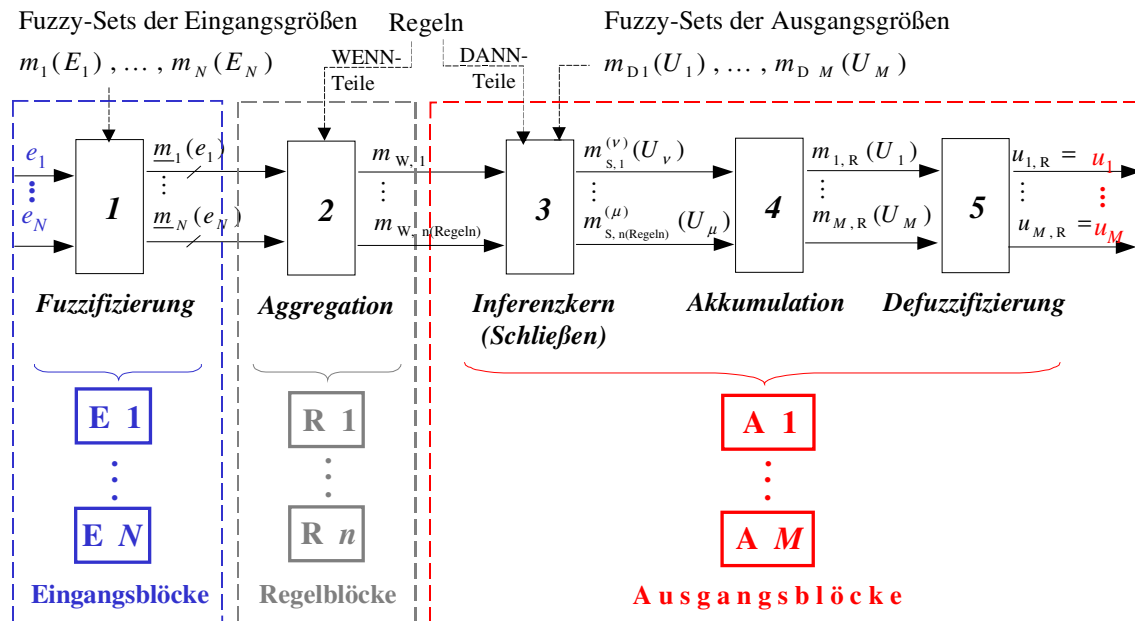


Bild 3.1: Die Verarbeitungsschritte eines FC.

N : Anzahl der Eingänge, M : Anzahl der Ausgänge, $n(\text{Regeln})$: Anzahl der Regeln.

1. Durch **Fuzzifizierung** werden die Zugehörigkeitsgrade des Momentanwertes jeder Eingangsgröße zu den Fuzzy-Mengen (Fuzzy-Sets) der ihr zugeordneten linguistischen Variablen ermittelt. Für jede Eingangsgröße e_i ergibt sich damit ein Vektor $\underline{m}_{E_i}(e_i)$ von Zugehörigkeitsgraden, dessen Komponentenzahl gleich der Zahl der linguistischen Werte der Eingangsgröße ist.

Die Fuzzifizierung erfolgt in den Eingangsblöcken durch Einsetzen der momentanen Eingangswerte in die durch die Knickstellen parametrisierten Zugehörigkeitsfunktionen.

2. Die **Aggregation** liefert zu jeder Regel i den Erfüllungsgrad m_{W_i} ihrer Prämisse. Die Prämisse wird hier auch als WENN-Teil der Regel bezeichnet – daher der Index „W“.

Die Aggregation leisten die Regel-Verknüpfungsblöcke (Version I) bzw. der zentrale Regelübersetzer (Version II).

3. Aus dem **Inferenzkern**, dem unscharfen Schließen, geht zu jeder Ausgangsgröße u_v , die in der Konklusion (dem DANN-Teil) einer Regel i erscheint, eine Zugehörigkeitsfunktion $m_{S_i}(u_v)$ hervor. Hierzu wird die Zugehörigkeitsfunktion $m_D(u_v)$ des in der Regel i für u_v geforderten linguistischen Wertes benötigt.

In Version I wird der Inferenzkern bereits in den Ausgangsblöcken ausgeführt, in Version II gemeinsam vom Regelübersetzer und den Ausgangsblöcken.

- Die **Akkumulation**, die zu jeder Ausgangsgröße u_v alle aus dem Schließen der Regeln hervorgegangenen Funktionen $m_{S_i}(u_v)$ zusammenfasst, liefert die resultierenden Zugehörigkeitsfunktionen $m_{v,R}(u_v)$.

Die Akkumulation wurde durch geeignete mathematische Umformungen eng mit dem Inferenzkern verwoben und erfolgt in den Ausgangsblöcken (Version I) bzw. im Regelübersetzer und in den Ausgangsblöcken (Version II).

- Durch die **Defuzzifizierung** entstehen schließlich die resultierenden Werte u_{1R}, \dots, u_{MR} . Diese sind die (scharfen) Ausgangswerte u_1, \dots, u_M des FC.

Die Defuzzifizierung findet in den Ausgangsblöcken statt.

4 Demonstrationsbeispiel: Invertiertes Pendel

4.1 Anlagenstruktur und Regelwerk

Ein Paradebeispiel für die Anwendung von Fuzzy-Control ist die Stabilisierung eines invertierten Pendels. Neben dem Balancieren in der labilen Gleichgewichtslage soll hier auch das kontrollierte Aufschwingen des Pendels simuliert werden. Die Regelstrecke wird modelliert durch einen Schlitten, der mit geringer Reibung auf geraden, horizontalen Schienen gleitet, sowie durch ein auf dem Schlitten montiertes Pendel. Auf den Schlitten wirkt eine vom Regler vorgegebene Kraft, die diesen und das Pendel beschleunigt; andere Einflussmöglichkeiten auf das Pendel gibt es nicht. Ein entsprechendes Modell wurde in *Modelica* implementiert (Bild 4.1).



Bild 4.1: DYMOLA-Animation des invertierten Pendels

Die Aufgabe in der vorliegenden Versuchsanordnung hat zwei Teile: das Aufschwingen und das Balancieren des Pendels. Die Kontrolle über das Aufschwingen hat dabei den Charakter einer Steuerung, das Balancieren entspricht eher der Aufgabe einer Regelung. Ein Fuzzy-Controller ist nicht eindeutig in eine der Kategorien Regelung und Steuerung (im klassischen Sinn) festzulegen und kann daher beide Aufgaben übernehmen. Um die Möglichkeiten der neuen Fuzzy-Control-Bibliothek umfassend auszunutzen, wird in diesem Zusammenhang eine geschlossene Lösung gezeigt in Gestalt eines MIMO-Fuzzy-Controllers mit sechs Eingängen und zwei Ausgängen.

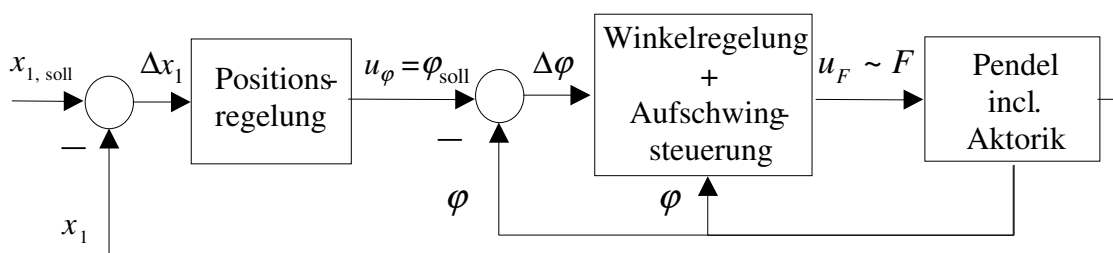


Bild 4.2: Struktur der Regelung/Steuerung zum invertierten Pendel

Die Steuerung des Aufschwings kann als spezielle Betriebsart der Winkelkontrolle gesehen werden. Das scheinbare Umschalten zwischen Winkelregelung und Aufschwungsteuerung erfolgt hier ohne eine übergeordnete Steuerung durch ein entsprechend parametrisiertes Fuzzy-Set der linguistischen Eingangsvariablen ϕ . Insgesamt ergibt sich eine kaskadierte Struktur entsprechend Bild 4.2. Der äußere Teil ist für die Positionsregelung des Schlittens zuständig und gibt dem inneren Teil den einzuregulierenden Soll-Winkel vor. Dieser liegt, je nachdem, ob sich der Schlitten rechts oder links von der Soll-Position befindet, etwas unter oder etwas über dem Winkel der labilen Ruhelage von $\pm\pi$. Dadurch wird der Schlitten veranlasst, in Richtung der Soll-Position zu fahren.

Der hier entworfene Fuzzy-Controller arbeitet mit den sechs linguistischen Eingangsgrößen

- x_1 : Schlittenposition, gemessen von der Mitte der Schienen,
- der_x_1 : Schlittengeschwindigkeit,
- x_d : Regelabweichung der Schlittenposition von der Soll-Position ($x_d = x_{1, \text{soll}} - x_1$),
- ϕ : Pendelwinkel abzüglich voller Umdrehungen,
- der_phi : Winkelgeschwindigkeit des Pendels,
- D_phi : Regelabweichung des Pendelwinkels ($\Delta\phi = \phi_{\text{soll}} - \phi$)

und den beiden linguistischen Ausgangsgrößen

- ϕ_{soll} : Soll-Winkel des Pendels, vorgegeben durch die Regler-Ausgangsgröße u_ϕ ,
- F : Antriebskraft auf den Schlitten.

Das verwendete Regelwerk besteht aus 22 Regeln. Für den Aufschwungvorgang werden vier Regeln formuliert, wobei die dritte und vierte verhindern sollen, dass der Schlitten über die Schienenenden hinaus fährt:

- R 1: WENN ϕ mittel UND x_1 mittel UND der_phi klein, DANN F klein.
 R 2: WENN ϕ mittel UND x_1 mittel UND der_phi groß, DANN F groß.
 R 3: WENN ϕ mittel UND x_1 klein, DANN F groß.
 R 4: WENN ϕ mittel UND x_1 groß, DANN F klein.

Für das Balancieren des invertierten Pendels gibt es erstens neun Regeln zur Korrektur der Schlittenposition x_1 durch Vorgabe des Soll-Winkels ϕ_{soll} :

- R 5: WENN ϕ nicht mittel UND x_d klein UND der_x_1 klein, DANN ϕ_{soll} mittel.
 R 6: WENN ϕ nicht mittel UND x_d klein UND der_x_1 mittel, DANN ϕ_{soll} klein.
 R 7: WENN ϕ nicht mittel UND x_d klein UND der_x_1 groß, DANN ϕ_{soll} klein.
 R 8: WENN ϕ nicht mittel UND x_d mittel UND der_x_1 klein, DANN ϕ_{soll} groß.
 R 9: WENN ϕ nicht mittel UND x_d mittel UND der_x_1 mittel, DANN ϕ_{soll} mittel.
 R 10: WENN ϕ nicht mittel UND x_d mittel UND der_x_1 groß, DANN ϕ_{soll} klein.
 R 11: WENN ϕ nicht mittel UND x_d groß UND der_x_1 klein, DANN ϕ_{soll} groß.
 R 12: WENN ϕ nicht mittel UND x_d groß UND der_x_1 mittel, DANN ϕ_{soll} groß.
 R 13: WENN ϕ nicht mittel UND x_d groß UND der_x_1 groß, DANN ϕ_{soll} mittel.

Zweitens gibt es neun Regeln zum Einregeln des Soll-Winkels ϕ_{soll} :

- R 14: WENN phi nicht mittel UND D_phi klein UND der_phi klein, DANN F mittel.
- R 15: WENN phi nicht mittel UND D_phi klein UND der_phi mittel, DANN F groß.
- R 16: WENN phi nicht mittel UND D_phi klein UND der_phi groß, DANN F groß.
- R 17: WENN phi nicht mittel UND D_phi mittel UND der_phi klein, DANN F klein.
- R 18: WENN phi nicht mittel UND D_phi mittel UND der_phi mittel, DANN F mittel.
- R 19: WENN phi nicht mittel UND D_phi mittel UND der_phi groß, DANN F groß.
- R 20: WENN phi nicht mittel UND D_phi groß UND der_phi klein, DANN F klein.
- R 21: WENN phi nicht mittel UND D_phi groß UND der_phi mittel, DANN F klein.
- R 22: WENN phi nicht mittel UND D_phi groß UND der_phi groß, DANN F mittel.

Die linguistische Größe phi wird hier nur gebraucht, um zu erkennen, ob sich das Pendel noch im Bereich des Aufschwingens befindet („mittel“) oder ob es schon in der Nähe der labilen Ruhelage angekommen ist („nicht mittel“). Für phi wird also nur eine Zugehörigkeitsfunktion benötigt; die Parametrierung der anderen Zugehörigkeitsfunktionen ist gleichgültig, da andere linguistische Werte von phi in keiner Regel erscheinen („nicht“ ist als Operator zu verstehen).

4.2 Aufbau des Fuzzy-Controllers

Der Modelica-Fuzzy-Controller für die Versuchsanordnung des invertierten Pendels kann nun aus den Komponenten der in Kapitel 2 vorgestellten Bibliothek aufgebaut werden. In diesem Beispiel wird von der Version I Gebrauch gemacht (Bild 4.3). Für beide Ausgänge (φ_{soll} und F) wurde die Höhenmethode benutzt.

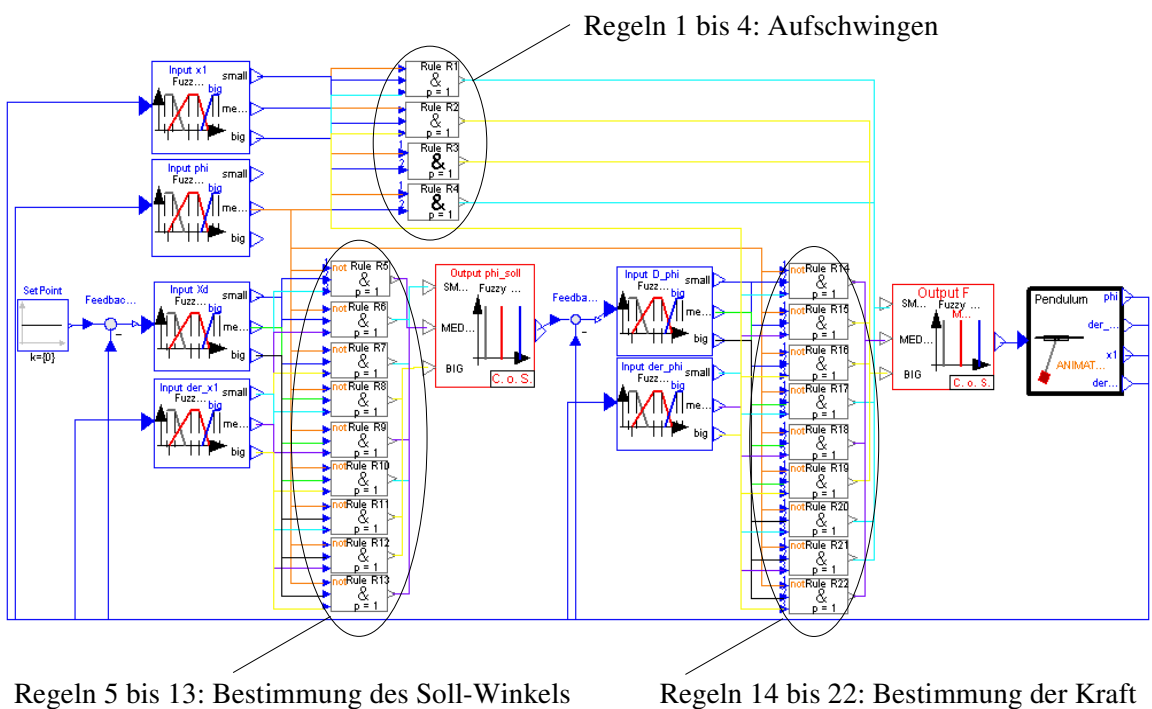


Bild 4.3: Modelica-Modell des invertierten Pendels mit Fuzzy-Controller nach Version I

4.3 Simulationsergebnisse

Die Simulation des Modells zum invertierten Pendel aus dem vorigen Kapitel (Bild 4.4) liefert mit den Anfangswerten

- $\varphi(0) = 0,1 \text{ rad}$,
- $x_1(0) = 0$,
- $\dot{\varphi}(0) = 0$,
- $\dot{x}_1(0) = 0$

die Ergebnisse laut Diagramm 4.1.

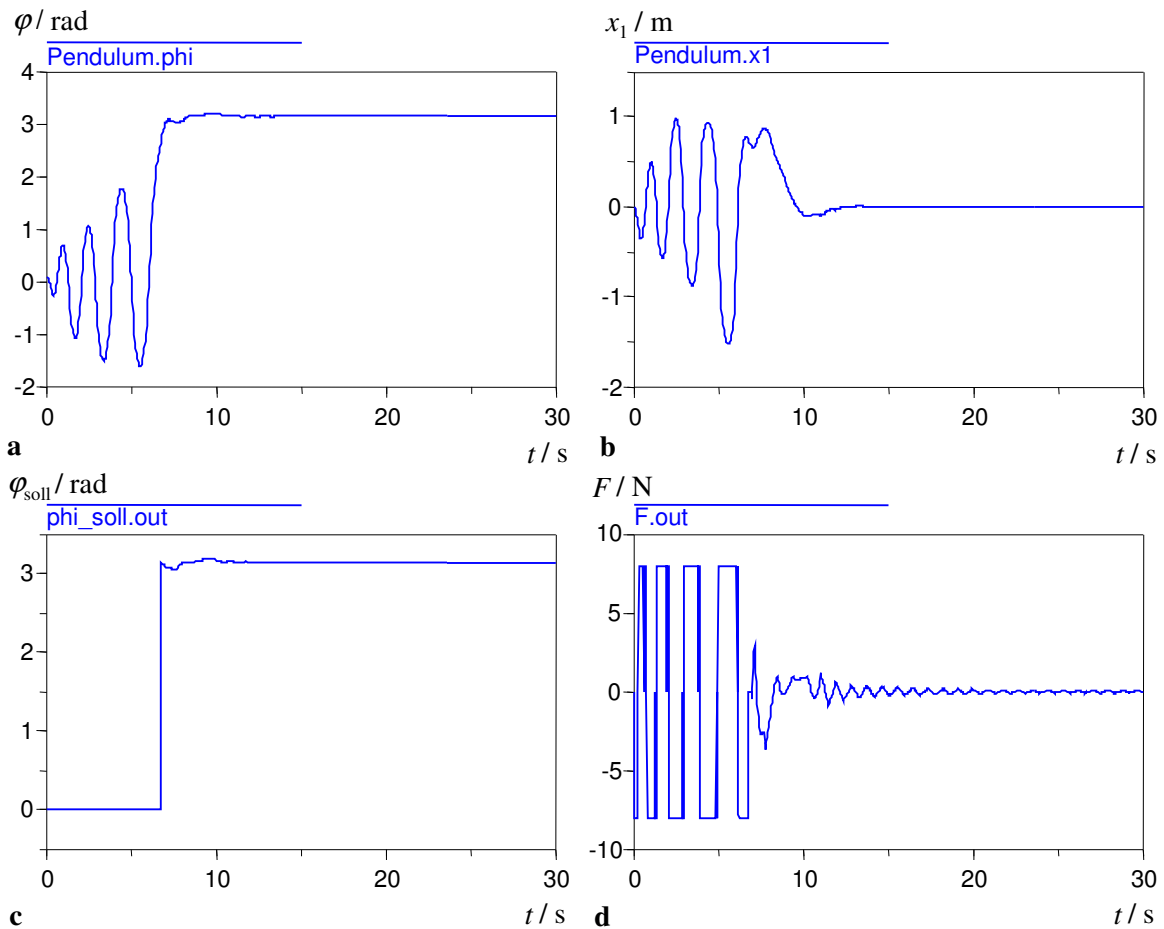


Diagramm 4.1: Simulationsergebnisse zum invertierten Pendel

a Pendelwinkel, **b** Schlittenposition, **c** Soll-Winkel, **d** Kraft

5 Fazit

Mit Hilfe der neuen FC-Bibliothek eignet sich die Entwicklungsumgebung *DYMOLA/Modelica* als Entwurfswerkzeug für Fuzzy-Controller. Dies gilt in besonderem Maße, wenn der Reglerentwurf eng verzahnt mit der Entwicklung eines neuen technischen Systems erfolgt. Denn die Bibliothek ermöglicht es nicht nur, Fuzzy-Controller leicht aufzubauen und zu modifizieren; da *Modelica*-Modelle an keine Berechnungskausalität gebunden sind, können parallel dazu auch strukturelle Änderungen an komplizierten Prozessmodellen sicher und effizient vorgenommen werden.

Für die Modellbausteine der Bibliothek wurden Icons entworfen, durch die sich die Funktion der einzelnen Komponenten dem Anwender intuitiv erschließt. Die Integration in die vorgegebene Benutzeroberfläche von *DYMOLA* verlangt gewisse Abstriche bei den graphischen Darstellungsmöglichkeiten. Andere Einschränkungen, die z.B. hinsichtlich der Anzahl der linguistischen Werte einer Variablen oder der Länge des Regelwerkes in der aktuellen Fassung bestehen, haben hingegen keine prinzipiellen Ursachen. Die innere Logik der Implementierung ist so gehalten, dass Erweiterungen der Bibliothek in den genannten Punkten analog zu den vorhandenen Algorithmen möglich sind.

6 Literatur

- [1] König, H.: *Der Regelbasisfehler – ein Zugang zum transparenten Entwurf von Fuzzy Controllern*. Fortschritt-Berichte VDI, Reihe 8, Nr. 554. VDI-Verlag, Düsseldorf; 1996.
- [2] Kahlert, J.: *Fuzzy-Logik und Fuzzy-Control*. Vieweg-Verlag, Braunschweig, Wiesbaden; 1993.
- [3] Kahlert, J.: *Fuzzy Control für Ingenieure*. Vieweg-Verlag, Braunschweig, Wiesbaden; 1995.

Benchmark: Qualitätskontrolle bei KFZ-Aggregaten

L. Haendel

Lehrstuhl für Elektrische Steuerung und Regelung

Universität Dortmund, 44221 Dortmund

Tel.: 0231 755 2760

haendel@esr.e-technik.uni-dortmund.de *oder* lars@haendels.de

Zusammenfassung

KFZ-Aggregate werden nach ihrer Fertigstellung – anhand der während eines Prüflaufes entwickelten Geräusche – von ausgebildeten Prüfern auf ihre einwandfreie Funktion hin getestet. Ein Aggregat kann entweder den Zustand einwandfrei (*io*) oder defekt (*nio*) haben. Um den Prüfer zu unterstützen soll ein System entwickelt werden, welches vollautomatisch den Zustand des Aggregates ermittelt. Dazu wird die Geräuschentwicklung der Aggregate während des Prüflaufes gemessen und daraus werden 304 akustische Merkmalswerte abgeleitet. Ziel ist es nun, ein Modul zu lernen, das in der Lage ist, den Zustand eines Aggregates bei Kenntnis der akustischen Merkmalswerte vorherzusagen. Nach Möglichkeit sollte dabei nur eine kleine Auswahl der Merkmale verwendet werden.

Im laufenden Betrieb wurden für 3 verschiedene Stichproben mit Aggregaten die akustischen Merkmale und die dazugehörige Entscheidung des Prüfers aufgezeichnet. Aufgrund der Aussagen eines Experten wird eine Auswahl von 40 relevanten Merkmalen getroffen und mittels des Fuzzy-ROSA-Verfahrens werden für 2 der Stichproben Fuzzy-Regeln generiert und an den jeweils anderen Stichproben getestet.

1 Aufgabenstellung

KFZ-Aggregate werden nach ihrer Fertigstellung auf ihre einwandfreie Funktion hin überprüft. Dazu wird jedes Aggregat auf einen Prüfstand gefahren und einem Prüflauf unterzogen der verschiedene typische Betriebssituationen simuliert. Ein ausgebildeter Prüfer beurteilt die dabei entwickelten Geräusche und beurteilt den Zustand eines Aggregates entweder als einwandfrei (*io*) oder als defekt (*nio*). Um ein System zu entwickeln, welches parallel dazu vollautomatisch den Zustand des Aggregates ermittelt, wird die Geräuschentwicklung der Aggregate während des Prüflaufes mittels Körperschallsensoren gemessen und daraus werden 304 akustische Merkmalswerte abgeleitet. Viele dieser Merkmale sind stark korreliert, da sie teilweise einmal dem mittleren und einmal dem maximalen Meßwert mehrerer Sensoren entsprechen. Die Ursachen eines Defektes sind vielfältiger Natur und äußern sich u.a. in der abnormen Erhöhung einiger weniger Merkmalswerte.

Ziel ist es, ein Modul zu lernen, mit dem es möglich ist, den Zustand eines Aggregates bei Kenntnis der akustischen Merkmalswerte vorherzusagen. Nach Möglichkeit sollte dabei nur eine kleine Auswahl der Merkmale verwendet werden, da so der Aufwand für die Aufzeichnung und Archivierung der Daten reduziert werden könnte. Aus Sicht der Produktion soll die Prüfung Aufschluß bezüglich der folgenden Gesichtspunkte geben:

Einerseits interessiert die Anzahl der defekten Aggregate, die in der Prüfung nicht erkannt werden und daher das Werk unbeanstandet verlassen. Andererseits interessiert die Anzahl der Aggregate, die tatsächlich einwandfrei sind, jedoch vom Prüfsystem als defekt beanstandet werden. Im ersten Fall entstehen Kosten für einen nachträglichen Austausch des Aggregats, im zweiten Fall werden Kosten durch unnötige Nacharbeiten verursacht. Basierend auf diesen beiden Gesichtspunkten sind die beiden Gütemaße *Auffindequote* und *Glaubwürdigkeit* wie folgt definiert.

- Die *Auffindequote* η gibt an, welcher Prozentsatz aller defekten Aggregate vom betrachteten Prognosemodul gefunden wird. Es handelt sich um den Quotienten aus der Anzahl der vom Modul korrekt als defekt eingestuften Aggregate zu der Gesamtzahl aller defekten Aggregate. Optimal ist ein Wert von 100%.
- Die *Glaubwürdigkeit* κ gibt an, mit welcher Wahrscheinlichkeit ein vom Modul als defekt eingestuftes Aggregat auch tatsächlich defekt ist. Es handelt sich um den Quotienten aus der Anzahl der vom Modul korrekt als defekt eingestuften Aggregate zu der Gesamtzahl aller vom Modul als defekt eingestufte Aggregate. Optimal ist ein Wert von 100%.

Das Prognosemodul soll einen menschlichen Bediener unterstützen. Eine höhere Auffindequote auf Kosten der Glaubwürdigkeit ist akzeptabel: Aus betrieblicher Sicht wird eine Auffindequote von mehr als 70% bei einer Glaubwürdigkeit von mehr als 30% angestrebt.

Im laufenden Betrieb wurden für 3 verschiedene Stichproben A , B bzw. C mit 2612, 4373 bzw. 10613 Aggregaten die akustischen Merkmale und die dazugehörigen Entscheidungen des Prüfers aufgezeichnet. Die Stichproben wurden chronologisch nacheinander während eines Zeitraums von etwa 4 Monaten aufgenommen. Der Anteil der defekten Aggregate beträgt jeweils ungefähr 1%.¹

Zur Anonymisierung wurden die Merkmalswerte innerhalb jeder Stichprobe auf den Bereich $[0..1]$ normiert. Da sich die Sensorcharakteristik des Prüfstandes über die Zeit verändert, empfiehlt es sich, die Stichproben mit dem jeweiligen Mittelwert \bar{x} und der Standardabweichung σ zu re-normieren nach $\tilde{x} = \frac{x-\bar{x}}{\sigma}$. Dies stellt nicht zwangsläufig eine Verletzung der Validierungsbedingung² dar, da beim realen Einsatz im laufenden Betrieb vor Ort problemlos die benötigten Werte online aktualisiert werden können.

2 Das Fuzzy-ROSA-Verfahren

Mittels des Fuzzy-ROSA³-Verfahrens [KK94, Kie97, Kro99, SKK98] ist es möglich, aus gemessenen Prozeßdaten computergestützt Fuzzy-Regeln der Form

WENN *Prämisse* DANN *Konklusion*

zu generieren. Dieser Prozeß ist in vier Hauptschritte unterteilt. Für jeden sind alternative Strategien verfügbar, so daß das Verfahren an die unterschiedlichen Anwendungsanforderungen angepaßt werden kann.

¹Diese Daten wurden von uns im Rahmen des GMA-FA 5.22 als Benchmark zur Verfügung gestellt.

²Bei Validierung eines gelernten Moduls dürfen keinerlei Informationen aus der zur Validierung verwendeten Stichprobe genutzt werden.

³Rule Oriented Statistical Analysis

- **Projektdefinition:** Vor der eigentlichen Regelgenerierung müssen die verwendeten Eingangsgrößen festgelegt und für diese, sowie für die Ausgangsgröße, Zugehörigkeitsfunktionen für die linguistischen Variablen festgelegt werden. Dies kann wissensbasiert, datenbasiert oder heuristisch erfolgen. Außerdem kann mit der maximalen Kombinationstiefe c_{max} die Anzahl der linguistischen Terme, die in einer Regelprämisse verknüpft werden dürfen, beschränkt werden.
- **Regelgenerierung:** Die Regelgenerierung erfolgt in zwei Schritten: Zunächst werden potentielle Kandidaten für eine Regel, sogenannte Hypothesen, aufgestellt. Anschließend werden diese Hypothesen mit Hilfe eines Relevanztests daraufhin überprüft, ob sie eine signifikante Regelmäßigkeit in der Datenstichprobe beschreiben. Besteht die Hypothese diesen Test, dann wird sie zu einer Regel erhoben und mit einem Bewertungsindex bewertet. Dieser gibt an, wie stark die Signifikanz der Regel durch die Daten abgesichert ist. Regeltest und Bewertungsindex werden im folgenden auch zusammenfassend als Relevanzmaß bezeichnet.
- **Regelreduktion :** Die Anzahl der Regeln kann anschließend durch Reduktionsverfahren verringert werden. Dabei können verschiedene Gesichtspunkte wie z.B. die komplette Überdeckung aller Eingangssituationen, die gleichmäßige Ausnutzung der Lerndaten oder der Bewertungsindex der Regeln betrachtet werden.
- **Regelsatzanalyse :** Durch die abschließende Analyse des Regelsatzes kann der Modellierungsprozeß und das Modellierungsergebnis bewertet und ggf. Rückschluß über die Problemstellung erhalten werden.

3 Vorgehensweise

Verwendete Eingangsgrößen Die Regelgenerierung mit dem Fuzzy-ROSA-Verfahren ist umso aufwendiger, je mehr Eingangsgrößen betrachtet und je mehr mögliche Kombinationen linguistischer Werte zugelassen werden und daher überprüft werden müssen. Auch war hier eine mit dem Fuzzy-ROSA-Verfahren durchgeführte Modellierung unter Verwendung aller 304 Merkmale auf Anhieb nicht erfolgreich. Es ergaben sich zwar durchaus zufriedenstellende Auffindequoten, jedoch sank die Glaubwürdigkeit teilweise auf $\kappa < 5\%$ ab. Dies wies darauf hin, daß die vorliegenden Stichproben wesentlich schwieriger zu modellieren sind, als eine früher für den selben Prozeß erhobene andere Stichprobe, bei der sich mit dem Fuzzy-ROSA-Verfahren zufriedenstellende Ergebnisse ergaben [Sla01]⁴. Um die Schwierigkeit der hier betrachteten Stichproben einschätzen zu können, wurde die Anzahl der Merkmale unter Nutzung von Expertenwissen reduziert und nur 40 ausgewählte Merkmale zur Regelgenerierung verwendet⁵.

Die Nummern der ausgewählten Merkmale sind: 154, 155, 156, 157, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 187, 188, 189, 192, 198, 216, 217, 220, 226, 233, 234, 243, 247, 248, 249, 250, 264, 274, 275, 276, 277, 280, 286, 298

⁴Diese dort verwendete Stichprobe haben wir inzwischen zusätzlich im Rahmen des GMA-FA 5.22 als Benchmark zur Verfügung gestellt.

⁵Bei Untersuchung des Meßrauschens der Merkmale in [Hae99] hatte sich nämlich gezeigt, daß einige der Merkmale aufgrund zu geringer Meßsicherheit für die Beurteilung des Aggregatzustandes weniger gut geeignet sind.

Zugehörigkeitsfunktionen Es werden für jedes Merkmal jeweils 5 eingangsseitige trapez- bzw. dreiecksförmige Zugehörigkeitsfunktionen (ZGF) mit den Namen *very low*, *low*, *normal*, *high* und *very high* definiert. Die Zugehörigkeitsfunktionen sind durch jeweils vier Punkte, die den Ecken eines Trapezes entsprechen, beschrieben. Die exakte Position ist, wie in folgender Tabelle ersichtlich, an den Mittelwert \bar{x} und die Standardabweichung σ des jeweiligen Merkmals j in der jeweiligen Stichprobe – und zwar sowohl beim Lernen als auch beim Validieren⁶ – gekoppelt.

Name	links		rechts	
	unten	oben	oben	unten
<i>very low</i>	$-\infty$	$-\infty$	$\bar{x} - \sigma$	$\bar{x} - 2\sigma$
<i>low</i>	$\bar{x} - 2\sigma$	$\bar{x} - \sigma$	\bar{x}	$\bar{x} - \sigma$
<i>normal</i>	$\bar{x} - 3\sigma$	\bar{x}	\bar{x}	$\bar{x} + 3\sigma$
<i>high</i>	\bar{x}	$\bar{x} + 3\sigma$	∞	∞
<i>very high</i>	$\bar{x} + 3\sigma$	$\bar{x} + 7\sigma$	∞	∞

Tabelle 1: eingangsseitige Zugehörigkeitsfunktionen

Der Zustand des Aggregats stellt die einzige Ausgangsgröße dar und kann die beiden Werte *io* und *nio* annehmen⁷. Als ausgangsseitige Zugehörigkeitsfunktionen werden, wie in Abbildung 1 skizziert, zwei Singletons $\mu_{io}(\mathbf{x})$ und $\mu_{nio}(\mathbf{x})$ verwendet.

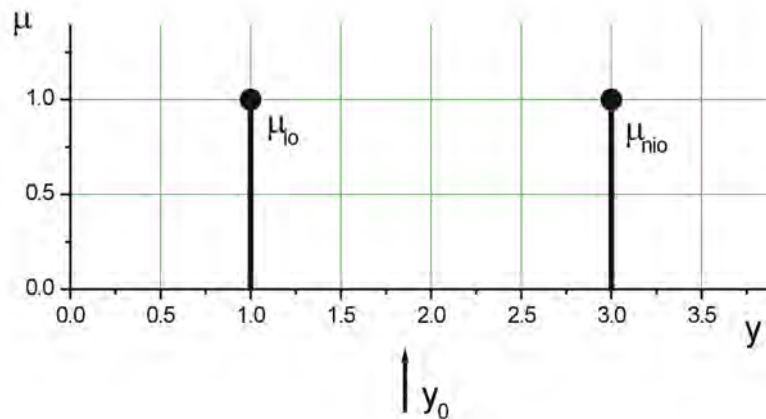


Abbildung 1: ausgangsseitige Zugehörigkeitsfunktionen

Der defuzzifizierte Ausgangsgrößenwert y gibt Aufschluß über den Zustand des Aggregates. Für Ausgangsgrößenwerte, die einen Schwellwert y_0 erreichen oder überschreiten, wird das dazugehörige Aggregat als *nio* klassifiziert. Siehe hierzu auch die genaueren Erläuterungen in Kapitel 4.

Kombinationstiefe Die Kombinationstiefe legt fest, wieviele linguistische Terme maximal in einer Regelprämisse miteinander verknüpft werden dürfen. Je weniger

⁶Diese Vorgehensweise stellt eine Alternative zu der in der Aufgabenstellung empfohlenen Re-Normierung der Daten dar und verletzt – wie dort beschrieben – nicht die Validierungsbedingung.

⁷Die Zustände sind als 1 für einwandfreie und als 3 für defekte Aggregate kodiert.

Terme verknüpft werden, umso kleiner wird der Suchraum. Bei den Regelgenerierungen in [PSKL98] ergaben sich bei Verwendung einer evolutionären Suchstrategie im wesentlichen Regeln deren Prämissen jeweils nicht mehr als 3 Terme enthielten, obwohl bis zu 6 Terme zulässig waren. Daher wird die Kombinationstiefe hier zu 2 oder 3 gewählt. Der sich dann ergebende Suchraum ist klein genug, um mit dem Fuzzy-ROSA-Verfahren auf einem derzeit üblichen PC vollständig durchsucht werden zu können.

Relevanzmaß Das hier für die Regelgenerierung verwendete Relevanzmaß ist die *relevante Trefferquote* J_{RHR} , die sich gemäß Gl. (1) berechnet. Die Ungleichheitsbedingung beschreibt den Regeltest.

$$J_{RHR} = \hat{P}(c | p) \quad \text{für } V_l(c | p) > V_u(c) \quad (1)$$

Darin ist $P(c | p)$ die bedingte Wahrscheinlichkeit dafür, daß die Konklusion c erfüllt ist, wenn die Prämisse p erfüllt ist und $P(c)$ bezeichnet die Wahrscheinlichkeit dafür, daß die Konklusion c generell erfüllt ist. $\hat{P}(c | p)$ und $\hat{P}(c)$ bezeichnen die für die unbekanntenen exakten Wahrscheinlichkeitswerte, basierend auf den relativen Häufigkeiten, berechneten Schätzwerte. Ferner sind V_l bzw. V_u die untere bzw. obere Grenze des Konfidenzintervalls, in dem die exakten Wahrscheinlichkeitswerte $P(c | p)$ bzw. $P(c)$ mit einer vorzugebenden Irrtumswahrscheinlichkeit α liegen.

Irrtumswahrscheinlichkeit Die Irrtumswahrscheinlichkeit α bestimmt die Größe der Konfidenzintervalle. Diese werden umso größer, je geringer die Irrtumswahrscheinlichkeit ist. Je größer die Konfidenzintervalle sind, umso mehr Datensätze müssen eine Regel stützen und umso schwieriger ist es, überhaupt Regeln zu finden. Hier wird mit $\alpha = 0.01$ ein Standard-Wert gewählt, der relativ große Konfidenzintervalle bedingt.

Regelreduktion Es besteht die Möglichkeit, einen unteren Grenzwert für die Bewertung einer Regel anzugeben, unterhalb derer diese Regel wieder gelöscht wird. Dieser Grenzwert wird zu eins gewählt. Es werden somit nur Regeln akzeptiert, für die innerhalb der betrachteten Lerndaten kein einziges Gegenbeispiel existiert.

Fuzzy-Operatoren und Defuzzifizierungsmethode Die Wahl der Fuzzy-Operatoren und der Defuzzifizierungsmethode hat Einfluß auf die Auswertung der generierten Regeln. Gestützt auf Expertenwissen werden Fuzzy-Parameter und Regelgenerierungsparameter vorab auf die Problemstellung abgestimmt: Als Fuzzy-UND-Operator wird das algebraische Produkt $\mu = \mu_1 \mu_2$ und als Fuzzy-ODER-Operator die algebraische Summe $\mu = \mu_1 + \mu_2 - \mu_1 \mu_2$ verwendet. Für die Defuzzifizierung wird die Schwerpunktmethode (COG) eingesetzt.

4 Interpretation des defuzzifizierten Ausgangsgrößenwertes

Der defuzzifizierte Ausgangsgrößenwert y liegt auf jeden Fall in dem Intervall $[1, 3]$. Die Interpretation eines Wertes an den Grenzen des Intervalls ist eindeutig: Ein Wert

von 1 bedeutet, ausschließlich Regeln, die einen Zustand von *io* empfehlen sind aktiviert. Ein Wert von 3 bedeutet analog dazu, daß ausschließlich Regeln aktiviert sind, die einen Zustand von *nio* empfehlen. In diesen Fällen ist der Klassifikationsentscheid eindeutig *io* bzw. *nio*. Für Ausgangsgrößenwerte innerhalb des Intervalls sind offenbar sowohl Regeln, die *io* und solche die *nio* empfehlen aktiviert. Zur Auflösung dieser Konfliktsituationen wird ein Aggregat als *nio* bewertet, wenn der Wert der defuzzifizierten Ausgangsgröße eine Schwelle y_0 erreicht oder überschreitet, d.h. wenn

$$y_{COG} \geq y_0 \quad (2)$$

gilt. Als Standardwert für die Auswertung aller Regelgenerierungen wird zunächst $y_0 = 2$ gewählt. Dadurch wird der am meisten empfohlene Zustand prognostiziert. Zusätzlich werden auch die Ergebnisse für andere Werte von y_0 ermittelt. In nicht abgedeckten Eingangssituationen, d.h. wenn keine einzige Regel aktiviert ist, wird ein Aggregat als *io* gewertet. Dieses ist vor allem deshalb sinnvoll, weil damit zu rechnen ist, daß größere Teile der Stichproben von einem generierten Regelsatz nicht klassifiziert werden. Daher ist es sinnvoll, als Defaultwert den am häufigsten vorkommenden Zustand zu prognostizieren.

Wichtige Folgerung: Zusammen mit obiger Vorgabe, nicht klassifizierte Datensätze als *io* zu werten, bedeutet ein Wert von $y_0 = 1$, daß de facto ausschließlich Regeln verwendet werden, die in ihrer Konklusion die Ausgangsgröße *nio* empfehlen, d.h. die Regeln, die *io* als Ausgangsgrößenwert empfehlen, haben keinen Einfluß auf das Klassifikationsergebnis.

5 Ergebnisse

Für die Stichproben *B* bzw. *C* werden mit dem Fuzzy-ROSA-Verfahren Fuzzy-Module generiert und auf den jeweils anderen Stichproben getestet. Vorher ist der Anteil der Datensätze von defekten Getrieben durch ihre Replikation auf etwa 10% erhöht worden, da ansonsten Probleme mit dem verwendeten Relevanzmaß aufgetreten wären. Tabelle 2 listet die erzielten Ergebnisse auf. Dabei bezeichnet c_{max} die gewählte maximale Kombinationstiefe und N die Anzahl der generierten Regeln.

Die Verwendung der höheren Kombinationstiefe $c_{max} = 3$ führt zu wesentlich größeren Regelsätzen. Die resultierende Auffindequote ist teilweise wesentlich besser, die Glaubwürdigkeit dabei allerdings dann etwas schlechter. Bei Berücksichtigung der Zielvorgabe einer minimalen Auffindequote von 70% und einer minimalen Glaubwürdigkeit von 30% scheint die Wahl der höheren Kombinationstiefe sinnvoll zu sein.

Mittels des Schwellwertes y_0 kann die Empfindlichkeit des generierten Fuzzy-Moduls bezüglich der Regeln, die *nio* als Zustand des Aggregates empfehlen, eingestellt werden. Ein niedrigerer Wert führt einerseits zu einer besseren Auffindequote und andererseits zu einer schlechteren Glaubwürdigkeit. Eine empirisch ermittelte Wahl von $y_0 = 1.1$ stellt einen sinnvollen Kompromiß dar.

		Lerndaten		c_{max}	N	
		<i>B</i>		2	206	
y_0	Validierung A		Lerndaten B		Validierung C	
	η	κ	η	κ	η	κ
2	11.5%	27.3%	60.0%	100%	46.9%	49.3%
1.1	23.1%	28.6%	60.0%	100%	49.4%	44.9%
1.0	26.9%	30.4%	60.0%	100%	49.4%	44.0%

		Lerndaten		c_{max}	N	
		<i>B</i>		3	1723	
y_0	Validierung A		Lerndaten B		Validierung C	
	η	κ	η	κ	η	κ
2	26.9%	21.2%	73.3%	100%	53.1%	32.3%
1.1	65.4%	21.8%	73.3%	100%	63.0%	19.8%
1.0	73.1%	16.7%	73.0%	100%	64.2%	13.8%

		Lerndaten		c_{max}	N	
		<i>C</i>		2	294	
y_0	Validierung A		Validierung B		Lerndaten C	
	η	κ	η	κ	η	κ
2	11.5%	25.0%	24.4%	84.6%	33.3%	100%
1.1	11.5%	21.4%	26.7%	52.2%	33.3%	100%
1.0	11.5%	16.7%	28.9%	44.8%	33.3%	100%

		Lerndaten		c_{max}	N	
		<i>C</i>		3	5200	
y_0	Validierung A		Validierung B		Lerndaten C	
	η	κ	η	κ	η	κ
2	19.2%	23.8%	42.2%	63.3%	66.7%	100%
1.1	38.5%	19.6%	60.0%	31.8%	66.7%	100%
1.0	50.0%	12.9%	60.0%	21.8%	66.7%	100%

Tabelle 2: Ergebnisse der Regelgenerierungen

6 Fazit und Vergleich zu früheren Untersuchungen

Hier wurden mit dem Fuzzy-ROSA-Verfahren unter Verwendung einer experten-basierenden Auswahl von 40 Merkmalen für den Benchmark KFZ-Aggregate Regelsätze generiert und getestet. Für $c_{max} = 3$ und $y_0 = 1.1$ ergeben sich im Mittel eine Auffindequote von $\eta \approx 57\%$ bei einer Glaubwürdigkeit von $\kappa \approx 23\%$. Damit werden bei Verwendung der ausgewählten 40 Merkmale bessere und robustere Ergebnisse erzielt, als in [Hae99], wo für eine Auswahl von 94 Merkmalen mit vergleichbarer Vorgehens-

weise mit dem Fuzzy-ROSA-Verfahren Regelsätze generiert und getestet wurden.

Insgesamt werden jedoch nicht für alle Stichproben Ergebnisse in der gewünschten Qualität erzielt. Für denselben zugrunde liegenden Prozeß wurden in einer Voruntersuchung in [Sla01] unter Verwendung einer anderen Datenstichprobe Ergebnisse in der gewünschten Qualität erzielt⁸. Die hier verwendeten Daten wurden – im Gegensatz zu den Daten der Voruntersuchung – über einen wesentlich längeren Zeitraum aufgenommen, um die Einsatzfähigkeit der datenbasiert generierten Fuzzy-Module unter praktischen Bedingungen – insbesondere bei Vorliegen von Parametervariationen des Prüfstandes und der Sensorik sowie bei Veränderungen der Fehlerarten und deren Ursachen – zu untersuchen.

In [Pri01] wurden mit dem Lernverfahren FUCS⁹ unter Verwendung aller Merkmale Regelsätze generiert und getestet. Damit konnten keine akzeptablen Ergebnisse erzielt werden. Daß hier bessere – wenn auch noch nicht befriedigende – Ergebnisse erzielt wurden, läßt keine Rückschlüsse auf die eingesetzten Lernverfahren zu, da hier Expertenwissen für die Merkmalsauswahl verwendet wurde. Ob und wie es es möglich ist, rein datenbasiert eine sinnvolle Merkmalsauswahl zu treffen, wird zur Zeit untersucht.

Literatur

- [Hae99] HAENDEL, L.: *Entwicklung und Implementierung eines adaptiven Fuzzy-Klassifikators zur akustischen Qualitätssicherung*, Universität Dortmund, Lehrstuhl für Elektrische Steuerung und Regelung, Diplomarbeit, 1999
- [Kie97] KIENDL, H.: *Fuzzy-Control methodenorientiert*. München : Oldenbourg Verlag, 1997
- [KK94] KRABS, M. ; KIENDL, H.: Automatische Generierung von Fuzzy-Regeln mit dem ROSA-Verfahren. In: *VDI-Berichte* 1113 (1994), S. 29–39
- [Kro99] KRONE, A.: *Datenbasierte Generierung von relevanten Fuzzy-Regeln zur Modellierung von Prozesszusammenhängen und Bedienstrategien*. Düsseldorf : VDI Verlag, 1999 (Fortschritt-Berichte VDI, Reihe 10, Nr. 615)
- [Pri01] PRIBER, Ulrich: Ergebnisse der Untersuchung von Benchmark-Daten mittels FUCS – Fuzzy Classification System. In: *Proc. 11. Workshop Fuzzy-Control des GMA-FA 5.22* (2001)
- [PSKL98] PRACZYK, J. ; SLAWINSKI, T. ; KIENDL, H. ; LIESKE, D.: Generierung eines Fuzzy-Klassifikators mit WINROSA zur Qualitätskontrolle in der Automobilindustrie. In: *Anwendersymposium zu Fuzzy Technologien und Neuronalen Netzen* (1998)
- [SKK98] SLAWINSKI, T. ; KRONE, A. ; KIENDL, H.: Automatisierung durch datenbasierte Fuzzy-Modellierung von Prozess bedienern. In: *Computational Intelligence: neuronale Netze, evolutionäre Algorithmen, Fuzzy-Control im industriellen Einsatz, Tagung Berlin*. Düsseldorf : VDI-Verlag, 1998 (VDI-Berichte, Nr. 1381), S. 203–219
- [Sla01] SLAWINSKI, T.: *Analyse und effiziente Generierung von relevanten Fuzzy-Regeln in hochdimensionalen Suchräumen*, Universität Dortmund, Diss., 2001. – VDI Verlag Düsseldorf, Fortschrittberichte VDI, Reihe 10, Nr. 686

⁸Es ergaben sich eine Auffindequote von $\eta = 67.5\%$ bei einer Glaubwürdigkeit von $\kappa = 40\%$ auf Validierungsdaten.

⁹Fuzzy-Classification-Systems

Fuzzy-Modellbildung für den Benchmarkdatensatz Kfz-Aggregate

Ralf Mikut

Forschungszentrum Karlsruhe GmbH, Institut für Angewandte Informatik,
D-76021 Karlsruhe, Postfach 3640,
Telefon: (07247) 82-5731, Fax: (07247) 82-5702, E-Mail: mikut@iai.fzk.de

1 Motivation

In diesem Beitrag wird die Leistungsfähigkeit automatischer Verfahren zur Fuzzy-Modellbildung, die am Forschungszentrum Karlsruhe in den letzten Jahren entwickelt wurden [1, 2, 3], mit alternativen Verfahren verglichen. Dazu wird für den Benchmarkdatensatz "Kfz-Aggregate" der Universität Dortmund und des GMA-Fachausschusses 5.22 ein Fuzzy-Modell entworfen und somit ein Methodenvergleich zu anderen Verfahren (z. B. FUCS in [4], ROSA in [5, 6]) ermöglicht.

Dazu wird

- der Datensatz vorgestellt (Abschnitt 2) und
- die Datenvorverarbeitung, Merkmalsbildung sowie das entworfene Fuzzy-Modell diskutiert (Abschnitt 3).

2 Daten

Die Aufgabenstellung besteht darin, aus Messdaten den Zustand eines Kfz-Aggregates zu beurteilen [5, 6]. Der Datensatz enthält eine Datenmatrix mit 307 Spalten. In den ersten drei Spalten stehen die Ausgangsklasse, die Identifikationsnummer und die Nummer der Prüfung. Somit resultieren 304 Merkmale x_i mit der Bezeichnung x_4, \dots, x_{307} für die i -te Spalte im Datensatz (Messdaten). Nähere Angaben zu den Merkmalen sind nicht bekannt, allerdings wird auf deren Korreliertheit und eine Parameterdrift hingewiesen. Die Ausgangsgröße y enthält die Klassen B_1 (fehlerfreies Aggregat, ca. 99% der Datensätze) und B_2 erkannter Fehler (ca. 1% der Datensätze). Aufgrund der subjektiven Erfassung ist diese Einstufung fehlerbehaftet. Die Zeilen der Matrix stehen für zeitlich fortlaufende Messungen. Die Stichprobe 3 mit 10613 Datensätzen wurde im Folgenden als Lerndatensatz verwendet, die Stichproben 1 (2612 Datensätze) und 2 (4373 Datensätze) als Testdatensätze.

3 Ergebnisse

Die Zielstellung besteht darin, mindestens 70 % der vom Prüfer als fehlerhaft eingestuften Aggregate mit einem Klassifikator als fehlerhaft zu erkennen (Auffindequote), wobei höchstens 30 % der vom Klassifikator als fehlerhaft erkannten Aggregate vom Prüfer als fehlerfrei eingestuft sein sollen (Glaubwürdigkeitsquote).

Zunächst wurden einige erfolglose Versuche unternommen, mit den gegebenen Merkmalen einen Fuzzy-Klassifikator zu entwerfen. Beteiligte Ursachen waren

- die teilweise erhebliche Parameterdrift (z. B. geschätzte Mittelwerte $\hat{\mu}$ und Streuungen $\hat{\sigma}^2$ von x_{14} für fehlerfreie Aggregate $\hat{\mu}_{14} = 0.224, \hat{\sigma}_{14} = 0.077$ und für als fehlerhaft eingestuftem Aggregate $\hat{\mu}_{14} = 0.306, \hat{\sigma}_{14} = 0.269$ in der Stichprobe 1, sowie für fehlerfreie $\hat{\mu}_{14} = 0.104, \hat{\sigma}_{14} = 0.030$ und für als fehlerhaft eingestufte Aggregate $\hat{\mu}_{14} = 0.216, \hat{\sigma}_{14} = 0.195$ in der Stichprobe 3 – deswegen geben auf einem Lerndatensatz entworfene Klassifikatoren auf einem Testdatensatz schlechte Ergebnisse) und
- diverse Verfahrensprobleme durch die extrem unterschiedliche Anzahl von Beispielen für die beiden Ausgangsklassen.

Aufgrund der Parameterdrift sind zunächst geeignete transformierte Merkmale zu suchen, um eine gute Klassifikation überhaupt erst zu ermöglichen. Somit handelt es sich eher um ein Merkmalsuche-Problem als um ein reines Klassifikationsproblem.

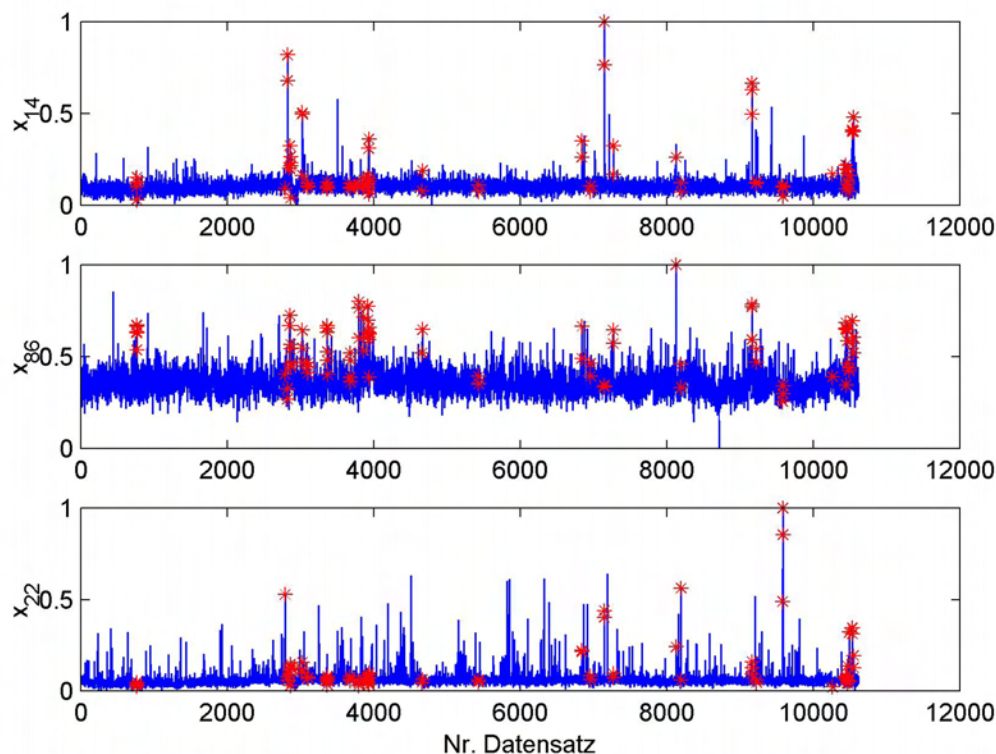


Bild 1: Zeitlicher Verlauf der drei besten Einzelmerkmale nach MANOVA im Lerndatensatz (Stichprobe 3), vom Untersucher erkannte Fehler wurden mit * gekennzeichnet

Zur Lösung wurde mit dem Lerndatensatz (Stichprobe 3) folgender Algorithmus angewendet:

- In einem ersten Schritt wurde mit dem MANOVA (Multivariate Analysis of Variance)-Verfahren [7] nach relevanten Merkmalen gesucht. Dabei

wurden bei einer schrittweisen Ergänzung des am besten passenden Merkmals zum bisher gefundenen Merkmalsatz in der Reihenfolge 15 Merkmale ($x_{14}, x_{86}, x_{22}, x_{103}, x_{87}, x_{67}, x_{220}, x_{16}, x_{187}, x_{35}, x_{218}, x_{70}, x_{236}, x_{219}, x_{81}$) ausgewählt. Den Verlauf der drei besten Merkmale verdeutlicht Bild 1. Tendenziell sind bei deutlich erhöhten Werten dieser Merkmale erkannte Fehler (*) häufiger, allerdings scheint sich kein einfacher Zusammenhang zu ergeben.

- Zur Bekämpfung der Parameterdrift wird für alle 15 ausgewählten Merkmale eine Vorfilterung durchgeführt. Ziel dieser Vorfilterung ist es, den aktuellen Mittelwert $\hat{\mu}_i[k]$ und die Standardabweichung $\hat{\sigma}_i[k]$ fehlerfreier Aggregate für das i -te Merkmal rekursiv zu schätzen. Werte, die außerhalb eines Korridors von Mittelwert und 3-facher Standardabweichung liegen, werden nicht in die Schätzung einbezogen. Die zugrundeliegenden Rekursionsvorschriften (zeitdiskrete IIR-Filter 1. Ordnung mit den Filterkonstanten $a_\mu = 0.99, a_\sigma = 0.995$) lauten für die Mittelwertschätzung

$$\hat{\mu}_i[k] = \begin{cases} \hat{\mu}_i[k-1] & \text{wenn } |x_i[k] - \hat{\mu}_i[k-1]| > 3 \cdot \hat{\sigma}_i[k-1] \\ a_\mu \cdot \hat{\mu}_i[k-1] + (1 - a_\mu) \cdot x_i[k] & \text{sonst} \end{cases} \quad (1)$$

und für die Standardabweichung

$$\hat{\sigma}_i[k]^2 = \begin{cases} \hat{\sigma}_i[k-1]^2 & \text{wenn } |x_i[k] - \hat{\mu}_i[k-1]| > 3 \cdot \hat{\sigma}_i[k-1] \\ \text{oder } |x_i[k-1] - \hat{\mu}_i[k-1]| > 3 \cdot \hat{\sigma}_i[k-1] \\ a_\sigma \cdot \hat{\sigma}_i[k-1]^2 + \frac{1}{2} \cdot (1 - a_\sigma) \cdot (x_i[k] - x_i[k-1])^2 & \text{sonst.} \end{cases} \quad (2)$$

Als Startwerte wurden der erste Messwert $x_i[1]$ im Datensatz (Mittelwertschätzer) bzw. ein konstanter Wert von $\hat{\sigma}_i[1] = 0.003$ eingesetzt. Ähnliche Schätzvorschriften wurden in [8] zur Erkennung von Muskelkontraktionen aus einer gestörten Zeitreihe mit gemessenen Muskelaktivierungen eingesetzt.

- Eine visuelle Analyse ergab, dass fehlerhafte Aggregate häufig durch einen Ausbruch aus dem aktuellen Korridor von Mittelwert und n -facher Standardabweichung gekennzeichnet sind. Zur Formalisierung dieser Analyse wird ein modifiziertes Merkmal eingesetzt, das die Abweichung des aktuellen Wertes vom Mittelwert als Vielfaches der Standardabweichungen angibt:

$$x_{Norm,i}[k] = \frac{x_i[k] - \hat{\mu}_i[k]}{\hat{\sigma}_i[k]}. \quad (3)$$

Diese modifizierten Merkmale wurden nun auf Basis der 15 ausgewählten Merkmale berechnet. Ähnliche Modifikationen wurden als sogenannte Normabweichungen in [9] zur Auswertung von Ganganalysedaten verwendet.

- Aus den neuen Merkmalen ($x_{Norm,14}, x_{Norm,86}, x_{Norm,22}, x_{Norm,103}, x_{Norm,87}, x_{Norm,67}, x_{Norm,220}, x_{Norm,16}, x_{Norm,187}, x_{Norm,35}, x_{Norm,218}, x_{Norm,70}, x_{Norm,236}, x_{Norm,219}, x_{Norm,81}$) wurde ein reduzierter Lerndatensatz mit 15 neuen Merkmalen zusammengestellt, der die 81 Fehler der Stichprobe 3 und nur noch 352 fehlerfreie Datensätze enthielt. Damit wird das Verhältnis fehlerhafter Aggregate zu fehlerfreien Aggregaten erhöht, wodurch der Algorithmus zu besseren Ergebnissen gelangt. Zur Auswahl der fehlerfreien Datensätze wurden diese nach der Summe der neuen Merkmale sortiert und jeder 30. verwendet, um möglichst unterschiedliche fehlerfreie Datensätze einzubeziehen.

- Eine automatische Fuzzy-Modellbildung [1, 2, 3] wurde mit den Schritten
 - Entwurf der Zugehörigkeitsfunktionen (Standardparameter: 5 Terme, etwa gleiche Beispiellanzahl pro Term unter Beachtung der Interpretierbarkeit),
 - Generieren von 10 Entscheidungsbäumen (Merkmale im Wurzelknoten: beste 10 Einzelmerkmale),
 - Regelextraktion,
 - Regelpruning mit Standardparametern und
 - Suche nach einer Regelbasis mit erzwungener Defaultregel *SONST fehlerfreies Aggregat* ($y = B_1$)

durchgeführt.

- Im Ergebnis entstand eine Regelbasis mit 4 Regeln, die 7 Merkmale enthalten, sowie einer Defaultregel und Zugehörigkeitsfunktionen entsprechend Tabelle 1:

$$\begin{aligned}
R_1: & \text{ WENN } (x_{Norm,103}=PM \cup PG) \cap (x_{Norm,87}=PK \cup PG) \cap x_{Norm,218}=PK \cap (x_{Norm,70}=PK \cup PG) \text{ DANN } y=B_2 \\
R_2: & \text{ WENN } x_{Norm,86}=PG \text{ DANN } y=B_2 \\
R_3: & \text{ WENN } x_{Norm,14}=PG \text{ DANN } y=B_2 \\
R_4: & \text{ WENN } x_{Norm,22}=PG \text{ DANN } y=B_2 \\
R_5: & \text{ WENN SONST DANN } y=B_1
\end{aligned}$$

Bezeichnung	$a_{l,1}$	$a_{l,2}$	$a_{l,3}$	$a_{l,4}$	$a_{l,5}$
$x_{Norm,14}$	-4.00	-1.00	0.00	1.00	40.00 (PG)
$x_{Norm,86}$	-3.00	-1.00	0.00	1.00	10.00 (PG)
$x_{Norm,22}$	-3.00	-1.00	0.00	1.00	60.00 (PG)
$x_{Norm,103}$	-10.00	0.00	1.00	4.00 (PM)	50.00 (PG)
$x_{Norm,87}$	-3.00	-1.00	0.00	1.00 (PK)	10.00 (PG)
$x_{Norm,218}$	-5.00	-1.00	0.00	1.00 (PK)	10.00
$x_{Norm,70}$	-3.00	-1.00	0.00	2.00 (PK)	10.00 (PG)

Tabelle 1: Stützpunkte $a_{l,i}$ der i -ten Zugehörigkeitsfunktion für die ausgewählten Merkmale x_l und Termbezeichnungen der in der Regelbasis verwendeten Terme PK: Positiv Klein, PM: Positiv Mittel, PG: Positiv Groß

Anschließend wurden für alle Datensätze im vollständigen Lerndatensatz und den beiden Testdatensätzen die sieben in der Regelbasis verwendeten transformierten Merkmale berechnet.

Auszüge aus der Regelbewertung für den reduzierten und den vollständigen Lerndatensatz zeigt Tabelle 2. Im reduzierten Lerndatensatz erreicht die Regelbasis eine Auffindefquote von 72.8% (59/81 Fehlern) und eine Glaubwürdigkeit von 87% (9 fehlerfreie Aggregate von insgesamt 68 als fehlerhaft erkannten Aggregaten). Über dem ursprünglichen Lerndatensatz (Stichprobe 3) erreichte die Regelbasis eine Auffindefquote von ebenfalls 72.8% und eine Glaubwürdigkeit von 21.8% mit einer Konfusionsmatrix

$$C_3 = \begin{pmatrix} 10321 & 211 \\ 22 & 59 \end{pmatrix}. \quad (4)$$

Regel-Nr.	B_k	Güte Q	Prognosegüte Q_P	Klarheit Q_K	erkannte Fehler	als fehlerhaft erkannte fehlerfreie Aggregate
R_1	B_2	0.24	0.24	1.00	41 (41)	4 (145)
R_2	B_2	0.22	0.22	1.00	27 (27)	1 (40)
R_3	B_2	0.07	0.07	1.00	6 (6)	0 (0)
R_4	B_2	0.05	0.05	1.00	4 (4)	5 (11)
R_5	B_1	0.00	0.00	0.81	-	-

Tabelle 2: Bewertung der Regeln [3] über dem reduzierten Lerndatensatz und Klassifikationsergebnisse über dem vollständigen Lerndatensatz (Zahlen in Klammern)

Dabei geben die Zeilen die vom Prüfer als fehlerfrei (1. Zeile) bzw. fehlerhaft (2. Zeile) und die Spalten die Fuzzy-Klassifikationsergebnisse fehlerfrei (1. Spalte) bzw. fehlerhaft (2. Spalte) an.

Die Regeln deuten auf eine Subgruppenbildung hin, bei der unterschiedliche Fehlerursachen eine Rolle spielen. Der in Regel R_1 modellierte komplexe Zusammenhang mit vier Merkmalen erfasst den größten Teil der Fehler und somit eine Subgruppe (41 erkannte Fehler). Die Redundanz zwischen den Aktivierungen der Regeln R_1 und R_2 ist sehr hoch, R_2 deckt lediglich 3 weitere Fehler ab und liefert bei 24 Fehlern identische Aussagen zu Regel R_1 . Relativ unabhängig voneinander bilden die Regeln R_3 (4 weitere Fehler) und R_4 (3 weitere Fehler) weitere Subgruppen. 8 weitere Fehler werden durch das Zusammenwirken aller Regeln gefunden, so dass die Regelbasis im Lerndatensatz 59/86 Fehlern findet. Interessant ist ebenso, dass einige Fehler nur extrem niedrige Aktivierungen der Regeln aufweisen (z. B. 14 fehlerhafte Aggregate mit einer Summe der Aktivierungen der Regeln R_1 bis R_4 kleiner 0.1) und somit ein grundlegend anderes Verhalten zeigen.

Eine grafische Veranschaulichung der Subgruppen über die beiden relevantesten Merkmale $x_{Norm,14}$ und $x_{Norm,86}$ ist in Bild 2 dargestellt. Die sechs am weitesten rechts liegenden fehlerhaften Aggregate \circ werden von Regel R_4 und die sich in der oberen Bildhälfte häufenden Fehler von Regel R_2 erkannt. Dabei ist deutlich zu erkennen, dass bei beiden Fehlergruppen das andere Merkmal normale Werte nahe Null aufweisen kann, aber auch Fehler mit deutlich erhöhten Werten für beide Merkmale existieren.

Mit der gefundenen Regelbasis resultiert über den Testdaten für die Stichprobe 1 eine Auffindequote von 53.8%, eine Glaubwürdigkeit von 26.9% und die Konfusionsmatrix

$$C_1 = \begin{pmatrix} 2548 & 38 \\ 12 & 14 \end{pmatrix} \quad (5)$$

bzw. für die Stichprobe 2 eine Auffindequote=60.0%, eine Glaubwürdigkeit=20.6% und die Konfusionsmatrix

$$C_1 = \begin{pmatrix} 4224 & 104 \\ 18 & 27 \end{pmatrix}, \quad (6)$$

die zwar noch nicht die Zielstellungen erfüllen, aber zumindest die prinzipielle Detektion wesentlicher Zusammenhänge belegen.

Der zum Erzielen dieser Ergebnisse notwendige Arbeitseinsatz betrug ca. zwei Arbeitstage. Weitere manuelle und automatische Ansätze zur Optimierung der Ergebnisse, wie

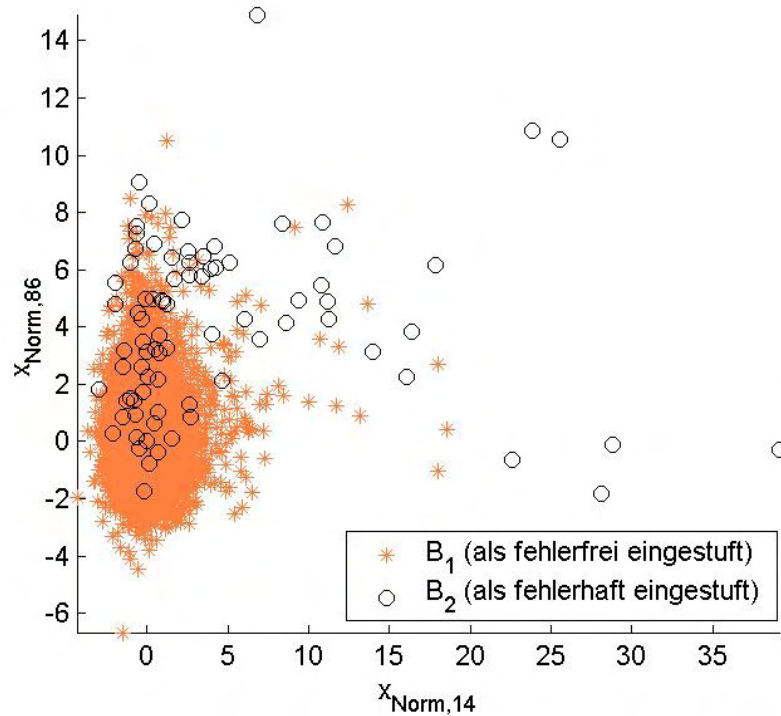


Bild 2: Als fehlerfrei und als fehlerhaft erkannte Aggregate über den zwei relevantesten Merkmalen $x_{Norm,14}$ und $x_{Norm,86}$ im vollständigen Lerndatensatz

- Optimierung der Stützpunkte der Zugehörigkeitsfunktionen mit dem Lerndatensatz in Stichprobe 3,
- Erhöhung der Anzahl falsch klassifizierter Datensätze im Lerndatensatz und erneutes Anlernen,
- Ergänzung weiterer Merkmale sowie
- Optimierung von Parametern im Suchverfahren und in der Datenvorverarbeitung

wurden aus Aufwandsgründen nicht durchgeführt, lassen aber Potenziale zur Ergebnisverbesserung vermuten. Allerdings dürfen die Testdatensätze dabei nur zur Validierung des Ergebnisses, aber keinesfalls zur Strategiewahl über dem Lerndatensatz verwendet werden, weil aufgrund der kleinen Beispielzahlen für fehlerhafte Aggregate und deren heterogenen Beschreibungen starke stochastische Streuungen der Ergebnisse auch über den Testdatensätzen zu erwarten sind.

Weitere Untersuchungen (Anlernen über die kleineren Stichproben 1 bzw. 2) wurden ebenfalls nicht vorgenommen. Hierbei sind wegen der kleineren Anzahl fehlerhafter Aggregate im Lerndatensatz schlechtere Klassifikationsergebnisse zu erwarten.

Die Ergebnisse liegen etwas schlechter als die Resultate in [5] mit einer Auffindequote von 67.5 % und einer Glaubwürdigkeitsquote von 40% über Testdaten, und ähnlich zu den Werten in [6] über Testdaten (Auffindequote von 57%, Glaubwürdigkeit 23%).

Auffällig ist, dass die dort erzielten Ergebnisse mit deutlich größeren Regelbasen und unter Einbeziehung vieler Merkmale (z. B. 40) erreicht werden konnten, was den Mess- und Vorverarbeitungsaufwand für eine Online-Realisierung im Vergleich zum hier entwickelten Fuzzy-Modell erhöht. Zudem werden durchgängig abweichende Merkmale herangezogen.

4 Zusammenfassung

In diesem Beitrag wird mit den baumbasierten Verfahren aus [1, 2, 3] für den Benchmarkdatensatz "Kfz-Aggregate" der Universität Dortmund und des GMA-Fachausschusses 5.22 ein Fuzzy-Modell entworfen. Nach der Einführung modifizierter Merkmale gelingt es mit einer auffällig kleinen Regelbasis von nur 5 Regeln, befriedigende Ergebnisse zu erzielen (Auffindequoten über Testdaten 53-60%, Glaubwürdigkeit 20-29%). Aus Aufwandsgründen wurde das Optimierungspotenzial der Algorithmen nicht ausgeschöpft.

Literatur

- [1] Mikut, R.; Jäkel, J.; Malberg, H.; Bretthauer, G.: Datenbasierter Entwurf von Fuzzy-Systemen für medizinische Diagnoseaufgaben. *Automatisierungstechnik* 48 (2000) 7, S. 317–326.
- [2] Mikut, R.; Peter, N.; Bretthauer, G.; Rupp, R.; Abel, R.; Siebel, A.; Döderlein, A.; Gerner, H.-J.: Fuzzy-Regelgenerierung und multivariate statistische Verfahren zur Schritt-Phasenerkennung in der Instrumentellen Ganganalyse. In: *Berichtsband 10. Workshop Fuzzy Control des GMA-FA 5.22*, S. 112–125. Karlsruhe: Forschungszentrum Karlsruhe, FZKA 6509. 2000.
- [3] Mikut, R.; Malberg, H.; Peter, N.; Jäkel, J.; Gröll, L.; Bretthauer, G.; Abel, R.; Döderlein, L.; Rupp, R.; Schablowski, M.; Siebel, A.; Gerner, H.: *Diagnoseunterstützung für die Instrumentelle Ganganalyse (Projekt GANDI)*. Forschungszentrum Karlsruhe, FZKA-6613, <http://bibliothek.fzk.de/zb/berichte/FZKA6613.pdf>. 2001.
- [4] Priber, U.: Ergebnisse der Untersuchung von Benchmarkdaten mittels FUCS – Fuzzy Classification System. In: *Proc. 11. Workshop Fuzzy Control des GMA-FA 5.22* (Mikut, R.; Reischl, M., Hg.), S. 100–105. Forschungszentrum Karlsruhe, FZKA 6660. 2001.
- [5] Slawinski, T.: *Analyse und effiziente Generierung von relevanten Fuzzy-Regeln in hochdimensionalen Suchräumen*, Bd. 10 (686) von *VDI-Fortschritt-Berichte, Reihe Informatik/Kommunikation*. Düsseldorf: VDI-Verlag. 2001.
- [6] Haendel, L.: Benchmark: Qualitätskontrolle von Kfz-Aggregaten. In: *Proc. 12. Workshop Fuzzy-Systeme* (Mikut, R.; Reischl, M., Hg.). Forschungszentrum Karlsruhe, FZKA-6767. 2002.

- [7] Ahrens, H.; Läuter, J.: *Mehrdimensionale Varianzanalyse: Hypothesenprüfung, Dimensionserniedrigung, Diskrimination bei multivariaten Beobachtungen*. Berlin: Akademie-Verlag. 1974.
- [8] Reischl, M.; Mikut, R.; Pylatiuk, C.; Schulz, S.; Beck, S.; Bretthauer, G.: Steuerungs- und Signalverarbeitungskonzepte für eine multifunktionale Handprothese. *Automatisierungstechnik (at)* 50 (6) (2002), S. 279–286.
- [9] Loose, T.; Malberg, H.; Mikut, R.; Dieterle, J.; Schabrowski, M.; Wolf, S.; Abel, R.; Döderlein, L.; Rupp, R.: Ein modulares Verfahren zur automatisierten Auswertung von Ganganalysedaten. *Biomedizinische Technik* 47 (2002) Suppl. 1(2), S. 700–703.

Entwicklungsstand und WWW-Präsenz des Tutorials "Erfolgreiche Anwendungen von Fuzzy Control" des GMA FA 5.22

B-M. Pfeiffer
Siemens AG
A&D ATS 32 (Vorfeldentwicklung Automatisierungsfunktionen)
D-76181 Karlsruhe
Tel.: +49-(0)721-595-5973 - Fax: +49-(0)721-595-6728
E-mail: Bernd-Markus.Pfeiffer@siemens.com

M. Reischl
Forschungszentrum Karlsruhe GmbH
Institut für Angewandte Informatik (IAI)
D-76021 Karlsruhe
Postfach 3640
Tel.: ++49-(0)7247-82-5749 - Fax: ++49-(0)7247-82-5786
E-Mail: Markus.Reischl@iai.fzk.de

Kurzfassung: Das im Jahre 2001 im Rahmen des GMA-Fachausschusses 5.22 vorgestellte Fuzzy-Tutorial ist erweitert und für verschiedene Zielgruppen und Medien aufbereitet worden: als Internetpräsenz und als Übersichtsaufsatz in der Zeitschrift *Automatisierungstechnik* (at). Dieser Artikel beschreibt die aktuelle Version des Tutorials, sowie die Struktur der erstellten Internetseite. Dabei wird u. a. der Informationsserver des Sonderforschungsbereichs (SFB) 531 der Uni Dortmund vorgestellt, in dessen Datenbank die einzelnen Steckbriefe des Tutorials integriert wurden. Abschließend folgt eine Diskussion über Erweiterung der gegenwärtigen Internetpräsenz und Anpassungsmöglichkeiten an die SFB-Datenbank.

1 Einführung

Das Tutorial „Erfolgreiche Anwendungen von Fuzzy Control“ wurde in einer Arbeitsgruppe des GMA Fachausschusses 5.22, bestehend aus B-M. Pfeiffer, J. Jäkel, A. Kroll, C. Kuhn, H.-B. Kuntze, U. Lehmann, T. Slawinski und V. Tews erarbeitet, wobei Steckbriefe zu einzelnen Anwendungen auch von weiteren Ausschussmitgliedern beigesteuert wurden.

Diese Sammlung, Aufbereitung und Klassifikation industrieller Applikationen soll die Erfahrungen bei der Anwendung von Fuzzy-Logik in der Prozess-Führung, -Regelung und -Steuerung kondensieren, die in 11 Jahren Arbeit des GMA-Fachausschusses 5.22 gesammelt worden sind. Die Sammlung ist nach *Grundaufgaben der eingesetzten Fuzzy-Module* in folgende Abschnitte gegliedert:

- Direkte Fuzzy-Regelung und Steuerung (Kürzel DFC),
- Fuzzy-Logik zur Parameter-Steuerung oder Adaption (ADAP),
- Prädiktivregler mit Fuzzy-Prozessmodellen (FPC),

- Diagnose (DIAG),
- Klassifikation/Mustererkennung (KLAS),
- Datenbasierte Modellierung/Identifikation (IDEN),
- Fuzzy-basierte Prozessführung und Optimierung (SUP),
- Experten- bzw. Entscheidungsunterstützungs-Systeme (DSS, „decision support systems“).

In jedem Abschnitt werden zunächst Literaturstellen angeführt, die sich auf Grund ihrer anschaulichen Darstellung als einführende Lektüre zu dieser Themengruppe eignen. Anschließend werden ausgewählte industrielle Applikationen an Hand von Steckbriefen vorgestellt. Diese umfassen Problemstellung, Lösungsmethode, Realisierungsform sowie Aufwand und Ergebnis (erzielter Nutzen). Abschließend werden aus der Zusammenfassung und Bewertung der Erfahrungen Kriterien zum erfolgreichen Einsatz von Fuzzy Control abgeleitet.

Die Darstellung des Tutorials lässt verschiedene Formen der Nutzung zu:

- Anwender, die mit den Grundbegriffen der Fuzzy-Logik vertraut sind, können sich einen Überblick über verschiedene methodische Nutzungsmöglichkeiten der Fuzzy-Logik innerhalb einer Automatisierungslösung verschaffen.
- Anwender und Lösungsanbieter, die einen bestimmten methodischen Ansatz verfolgen, können sich über vorhandene ähnliche Applikationen informieren, und von diesen Erfahrungen profitieren.
- Anwender, die einer bestimmten Branche angehören, können sich über Einsatzmöglichkeiten (Referenzen) in ihrer Branche informieren.
- Anbieter von Fuzzy-Produkten und Fuzzy-Lösungen können die Informationen vertrieblich nutzen, um Kunden die vielfältigen Einsatzmöglichkeiten und Vorteile zu demonstrieren.
- Lehrende an Hochschulen, Fachhochschulen und industriellen Schulungseinrichtungen können die Beispiele zur Illustration der methodischen Grundlagen heranziehen.
- Alle Leser, auch aus dem wissenschaftlichen Umfeld, können das Tutorial als Einstieg zum Aufsuchen weiterführender Literatur nutzen, oder sogar direkt passende Ansprechpartner finden.

2 Historie des Tutorials und neuere Entwicklungen

Eine erste Textfassung [1.] wurde auf dem Workshop „Fuzzy Control“ des GMA 5.22 im Jahr 2001 vorgestellt. Eine deutlich erweiterte Langfassung [2.] wurde im Juli 2002 bei der Zeitschrift Automatisierungstechnik (at) eingereicht. Sie ist positiv begutachtet und wird voraussichtlich in zwei aufeinander folgenden Heften 10 und 11/2002 als Übersichtsaufsatz erscheinen. Im Rahmen der Erstellung der Langfassung in der Arbeitsgruppe sind weitere Steckbriefe entstanden:

- Regelung eines Trocknungsprozesses in der Zuckerindustrie
- Regelung und Prozessüberwachung von Kläranlagen
- Fuzzy-basierte Prozessführung einer Neutralisationsanlage

Die für die Zeitschrift Automatisierungstechnik (at) ausgewählten Steckbriefe wurden fast durchgängig mit Bildern illustriert. Außerdem wurde das Literaturverzeichnis deutlich erweitert.

Aufgrund der positiven Resonanz wurde im November 2001 im Ausschuss beschlossen, das Tutorial auch im Internet zur Verfügung zu stellen. Der Einstieg erfolgt über die Internet-Seite des Fachausschusses [4.]. Die einzelnen Steckbriefe liegen als „Methodeneinsätze“ in der Datenbank [5.] des SFB 531 „Computational Intelligence“ der Uni-Dortmund. Die Internet-Umsetzung wurde von H. Reischl und H. Dr. Mikut, Forschungszentrum Karlsruhe, realisiert.

3 Internet-Umsetzung des Tutorials

3.1 Der Informationsserver „Computational Intelligence“ des SFB 531 der Uni-Dortmund

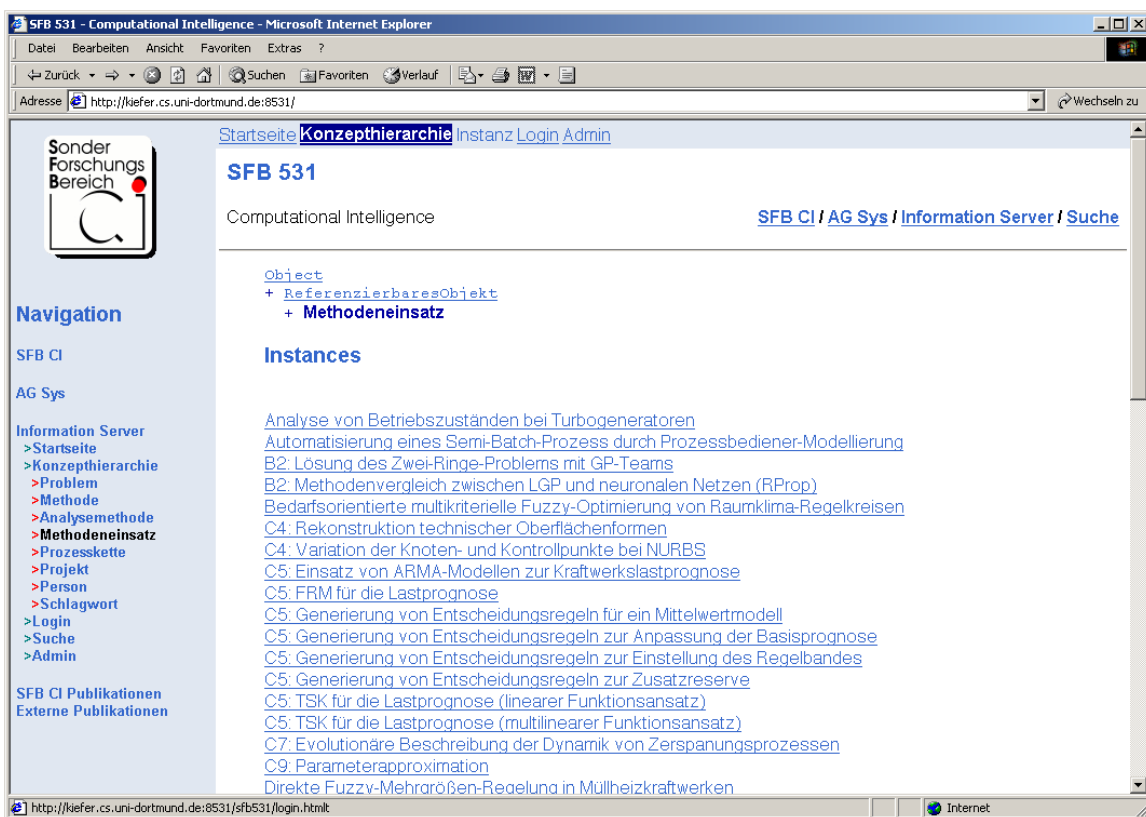


Bild 1: SFB 531 – Informationsserver

Auf der Suche nach der Lösung eines Problems besteht aus Sicht des Anwenders eine große Schwierigkeit darin, eine geeignete Methode für die jeweilige Problemstellung zu finden. Um diesen Arbeitsablauf zu vereinfachen, wurde vom SFB 531 ein Informationsserver implementiert, mit dem Ziel, verschiedene Bereiche der Computational Intelligence systematisch zu erfassen und Verweise auf artverwandte Themengebiete zu generieren. Die Einordnung verschiedener Themengebiete in das Kategoriensystem (s. Navigationsleiste links in Bild 1) orientiert sich dabei an

Problemen und *Methoden* (auch Analysemethoden), sowie *Methodeneinsätzen* bzw. *Prozessketten*, d. h. sukzessiver Anwendung verschiedener *Methodenklassen*.

Dem Server liegt folgende Hierarchie von Konzepten zu Grunde, wobei zu manchen Objekten exemplarische Instanzen einblendet werden:

Object

+ Anwendungsgebiet

- C4:Dynamische Triangulation
- C4:Flächenrekonstruktion
- C7: Modellbildung von Zerspanungsprozessen
- C7: Spanflussmodellbildung

+ Anwendungsgebietsart

- Hauptanwendungsgebiet
- Randanwendungsgebiet

- + Datenraumwert
- + Domaenenwissen
- + Domaenenwissenseigenschaft
- + Einsatzart
- + Einzelergebnis
- + Link
- + Literaturangabe
- + Parameterbeschreibung
- + Parametrisierung
- + Person
- + Problemloesungsmethode

- ARMA-Modelle
- Conjugate Gradient Strategy
- DIOGENES Diffusionsansatz nach J. Sprave
- FEM (Finite Elemente Methode)
- Neuronales Netz (-> Teilprojekt C6)
- Simulation (z.B. Atomverhalten ->C7)
- klassische Sortieralgorithmen

+ Projekt

+ ReferenzierbaresObjekt

- | + Analysemethode

- | + Guetemass

- | + Methode

- | + Methodenbereich

- Evolutionäre Algorithmen (EA)
- Fuzzy Logik (FL)
- Genetische Programmierung (GP)

- Maschinelles Lernen (ML)

- Neuronale Netze (NN)

- Statistik

- | + Methodeneinsatz

- Analyse von Betriebszuständen bei Turbogeneratoren
- Automatisierung eines Semi-Batch-Prozess durch Prozessbediener-Modellierung
- B2: Lösung des Zwei-Ringe-Problems mit GP-Teams
- B2: Methodenvergleich zwischen LGP und neuronalen Netzen (RProp)

....

- | + Problem

Bestimmung der Herkunft von Weinen (WINE)
C4: ES/GP-CSG-Optimierung
C4: Evolutionäre Flächenrekonstruktion
C4: Evolutionäre Optimierung dynamischer Triangulationen
C4: Evolutionäre diskrete NURBS-Flächenapproximation
C4: Hybridisierung von NURBS und CSG -Ansätzen
C4: Parallelisierung von Evolutionären Algorithmen

.....

| + Problemart

Approximation
Interpolation
Klassifikation
Mustererkennung
Optimierung
Prognose
Prozeßmodellierung
Regelungsproblem
Regression
theoretisches Analyseproblem

| + Prozesskette

| + Validierungsansatz

+ Schlagwort

+ Versuchsinstanz

Die vorhandene Infrastruktur des Informationsservers eignet sich gut zur Aufnahme und Verwaltung der Steckbriefe des Fuzzy-Tutorials. Jeder Steckbrief wird als ein *Methodeneinsatz* definiert. Die einzelnen Steckbriefe werden an die Datenbankmasken angepasst, wobei die Gliederung in drei Abschnitte gemäß Tabelle 1 erhalten bleibt, sowie durch bis zu drei Literaturstellen ergänzt wird.

3.2 Internetdarstellung des Tutorials

Um einen komfortablen und übersichtlichen Zugang zu den Steckbriefen auf dem Informationsserver zu gewährleisten, wurde eine Einstiegsseite erstellt, die dem Anwender die Steckbriefe geordnet präsentiert und die Verlinkung auf den Informationsserver übernimmt (Bild 2). Als Ordnungsschema kommt hierbei eine Ordnung nach Methodenklassen und eine Ordnung nach Branchen zum Einsatz. Zusätzlich ist eine zusammenhängende Textfassung des Tutorials verfügbar (vgl. [1.]), sowie eine Kontaktadresse und verschiedene weiterführende Links. Die Navigation auf der Einstiegsseite geschieht durch Buttons, die sich auf der linken Seite der Homepage befinden.

3.2.1 Ordnung nach Methodenklassen

Um eine Ordnung nach *Methodenklassen* durchzuführen, wurde die in der Einleitung erwähnte Klassifizierung nach Grundaufgaben der Fuzzy-Module verwendet und anhand von Blockschaltbildern verdeutlicht (Bild 3).

Jeder *Methodenklasse* sind dabei die zugehörigen Steckbriefe untergeordnet, deren Inhalt per Hyperlink angezeigt werden kann.



Bild 2: Darstellung eines Steckbriefes gemäß des Formates des SFB 531

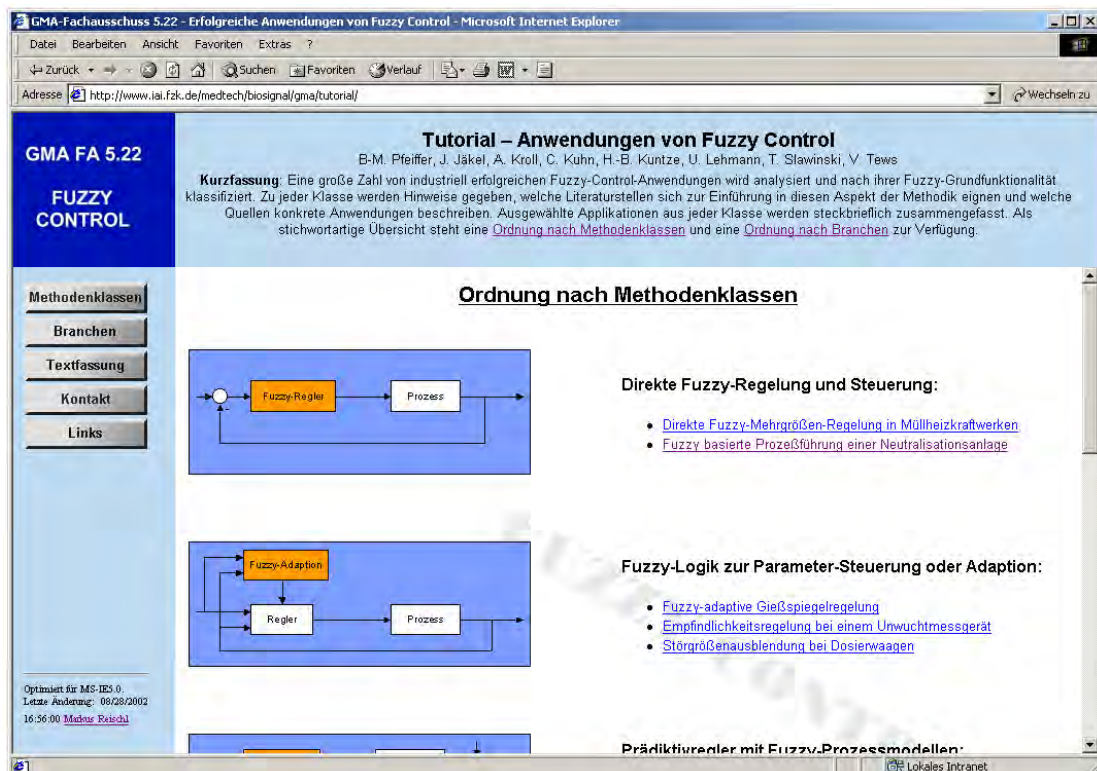


Bild 3: WWW-Präsenz des Fuzzy Tutorials, Einstiegsseite, Ordnung nach Methodenklassen

3.2.2 Ordnung nach Branchen

Die Ordnung nach *Branchen* gibt einen Überblick über den Einsatz verschiedener Methoden in den Branchen Maschinenbau, Automobilproduktion, Chemie, elektrische Energietechnik, Stahl, Glas-Steine-Erde, Heizung-Lüftung-Klima (Bild 4).



Bild 4: Ordnung nach Branchen

3.3 Zusammenhang zwischen der Begriffswelt des Tutorials und des Informationservers

Im augenblicklichen Zustand sind alle Steckbriefe des Fuzzy-Tutorials als *Methodeneinsatz* in die Datenbank des Informationservers integriert. Problematisch ist hierbei, dass einerseits

- eine konsistente Einordnung der Steckbriefe auf Grund einer zu groben Hierarchie noch nicht möglich ist und andererseits
- Begrifflichkeiten zwischen Tutorial und Informationsserver differieren.

Tabelle 1 zeigt eine Übersicht über verwendete Begriffe und deren Definitionen in den beiden Systemen.

Die eingangs vorgestellten *Grundaufgaben der eingesetzten Fuzzy-Module* sind als Unterklassen zum *Methodenbereich* „Fuzzy Logik“ des SFBs einzuordnen. Dabei befassen sich die ersten drei *Methodenklassen* mit der Lösung von Aufgaben der *Problemart* "Regelungsproblem", die vierte und fünfte Klasse dient zur Lösung von Aufgaben der *Problemart* "Klassifikation" oder "Mustererkennung". Methoden der sechsten Klasse können zur "Prozessmodellierung", "Prognose", "Klassifikation" oder

"Regelung" eingesetzt werden, Methoden der siebten Klasse dienen primär der "Optimierung".

	SFB	GMA	Beispiel
Hierarchie	Problemart		Regelungsproblem, Klassifikation, Mustererkennung, Prozess-Modellierung, Prognose
	Methodenbereich		Fuzzy-Logik, Neuronale Netze
	(fehlt)	Methodenklasse	Direkte Fuzzy-Regelung, Fuzzy-Identifikation
	Methode		Fuzzy-Rosa-Verfahren
	Methodeneinsatz	Steckbrief (Anwendung)	Modellierung und Regelung einer Roheisen-Entschwefelungsanlage
Steckbrief	Problembeschreibung	Problemstellung	(Die drei Abschnitte eines Steckbriefs lassen sich 1:1 zuordnen)
	Ansatz	Lösungsmethode, Realisierungsform	(Die drei Abschnitte eines Steckbriefs lassen sich 1:1 zuordnen)
	Zusammenfassende Bewertung	Aufwand und Ergebnis (erzielter Nutzen)	(Die drei Abschnitte eines Steckbriefs lassen sich 1:1 zuordnen)
Sonstiges	(fehlt)	Branche	
	Anwendungsgebiet	Steckbrief (Anwendung)	Modellbildung von Zerspanungsprozessen

Tabelle 1: Vokabular von SFB und GMA Tutorial

Falls bei *Methodeneinsätzen*, d.h. konkreten Anwendungen aus einer *Methodenklasse* (z. B. Identifikation) einzelne *Methoden* (z. B. Fuzzy-Rosa-Verfahren) zum Einsatz kommen, die bereits in der Liste der *Methoden* verfügbar sind, wird eine entsprechende zweiseitige Relation gesetzt.

Die zur Charakterisierung einzelner *Steckbriefe* notwendige Hierarchiestufe *Methodenklasse* ist im Konzept des Informationsservers bislang nicht implementiert.

Weiterhin ist ein referenzierbares Objekt "Branche" erforderlich. Die bisherigen sog. "Anwendungsgebiete" sind konkrete Anwendungen, und lassen sich den entsprechenden Branchen zuordnen. Auch in dieses Schema lassen sich alle unsere Steckbriefe einordnen, z.B. Anwendung "Gießspiegelregelung" der Branche "Stahl".

Die Zusammenhänge zwischen den verschiedenen Konzepten und Begriffen zeigt Bild 5.

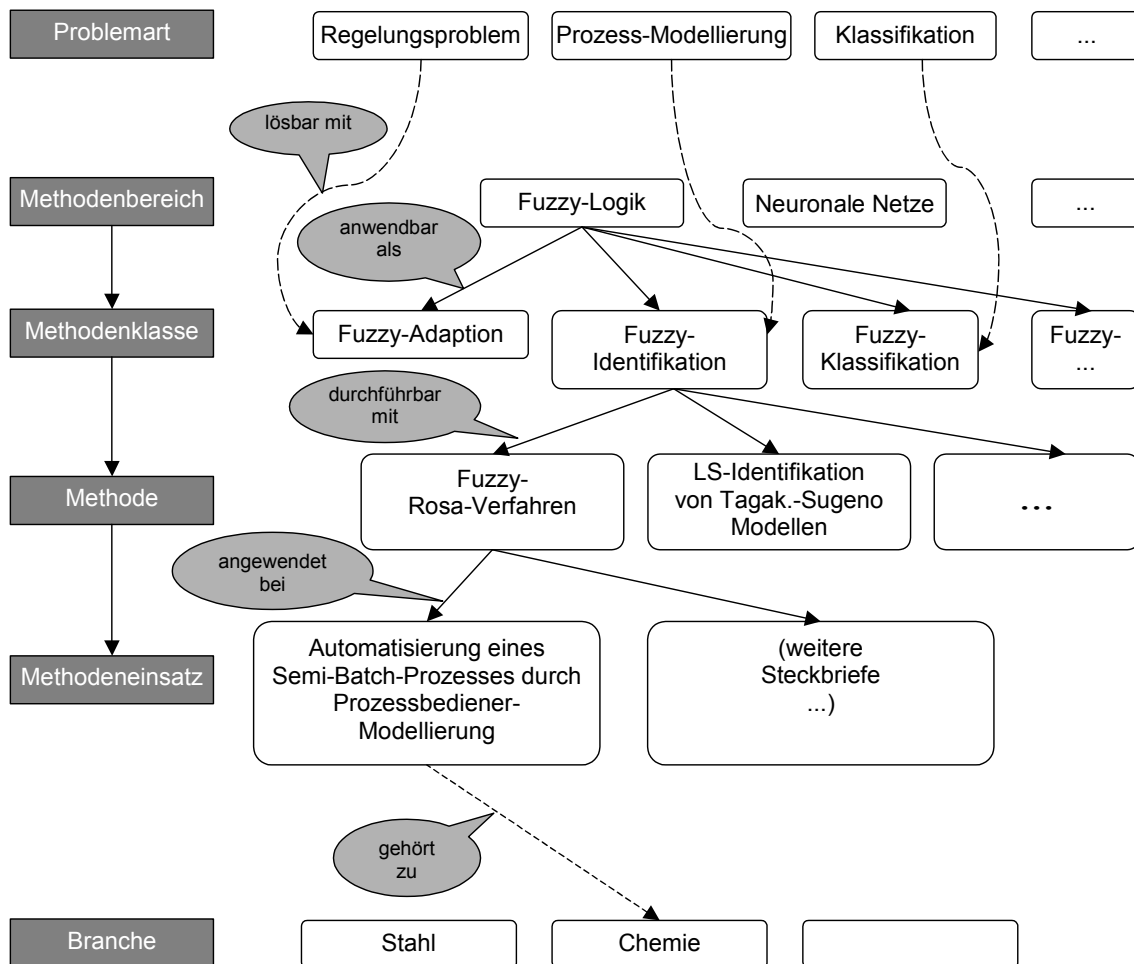


Bild 5: Gewünschte Hierarchie von Konzepten auf dem Informationsserver

4 Erstellung von Steckbriefen

Neben der weltweiten, einfachen Zugänglichkeit liegt ein besonderer Vorteil der Internet-Darstellung darin, dass sie laufend aktualisiert werden kann. Es ist also ausdrücklich erwünscht, dass erfolgreiche Applikationen, die von Mitgliedern des Fachausschusses selbst entwickelt worden oder ihnen zumindest direkt bekannt sind, in die Internet-Darstellung aufgenommen werden.

Ein Steckbrief umfasst ca. eine Seite ASCII-Text, der sich in drei Abschnitte gliedert:

- Problemstellung
- Lösungsmethode, Realisierungsform
- Aufwand und Ergebnis (erzielter Nutzen)

Eine Zuordnung zu den 8 Methodenklassen kann vom Autor des Steckbriefs selbst vorgeschlagen oder in der Tutorial-Arbeitsgruppe vorgenommen werden. Die Illustration mit Bildern ist zumindest in der jetzigen Version des Internet-Auftritts noch nicht vorgesehen (→ Diskussion !). Zu jedem Steckbrief sollte mindestens eine recherchierbare (!) Literaturstelle (Buch, Zeitschrift, Konferenzband mit ISSN-Nummer o.ä.) angegeben werden, in der sich der Leser vertieft informieren kann.

Eingereichte Beiträge werden in der Arbeitsgruppe begutachtet (→ Diskussion !) und dann vom FZK (Forschungszentrum Karlsruhe) ins Internet gestellt.

5 Ausblick und Diskusssion

Auf dem 12. Workshop soll der Internet-Auftritt im Kreis des Fachausschusses diskutiert werden. Im Fokus steht dabei die genaue Einordnung in die Taxonomie und Begriffswelt des SFB, sowie die laufende Aktualisierung der Internet-Seiten durch Ergänzung neuer Steckbriefe.

Literatur

- [1.] B-M. Pfeiffer, J. Jäkel, A. Kroll, C. Kuhn, H. - B. Kuntze, B. Lehmann, T. Slawinsky, V. Tews: Tutorial: erfolgreiche Anwendungen von Fuzzy Control. 11. Workshop "Fuzzy Control" des GMA-Fachausschusses 5.22, Dortmund, Nov. 2001.
- [2.] M. Pfeiffer, J. Jäkel, A. Kroll, C. Kuhn, H. - B. Kuntze, B. Lehmann, T. Slawinsky, V. Tews: Übersicht: Erfolgreiche Anwendungen von Fuzzy Logik und Fuzzy Control. (voraussichtlich) Automatisierungstechnik (at), 10 und 11/2002.
- [3.] R. Klinkenberg, T. Slawinski: Systematisierung der Beschreibung von Problemen, Methoden und Methodeneinsätzen. 11. Workshop "Fuzzy Control" des GMA-Fachausschusses 5.22, Dortmund, Nov. 2001
- [4.] <http://www.iai.fzk.de/medtech/biosignal/gma/tutorial>
- [5.] <http://kiefer.cs.uni-dortmund.de:8531/sfb531>

Evolved ant colonies optimization in the scheduling of logistic processes

C. A. Silva^{1,2} T.A. Runkler¹ J.M. Sousa²

¹Siemens AG, Corporate Technology
Information and Communications, Neural Computation Department
81730 Munich, Germany

² Technical University of Lisbon
Instituto Superior Técnico, DEM, GCAR
1049-001 Lisbon, Portugal
E-Mail: carlos.silva.external@mchp.siemens.de

Abstract

This paper addresses the problem of optimizing the schedule of logistic processes using ant colonies and fuzzy decision making. We consider here the problem of dynamic assignment of components to orders and choose the solution that is able to deliver more orders at the correct delivery date, while respecting the priority degree of the orders. A compromise between these conflicting goals is achieved using the ant colonies algorithm to optimize a fuzzy weighted objective function. The simulation results show that the proposed evolved ant colonies with fuzzy optimization present good results for this type of problems.

1 Introduction

Recently, the analysis and control of logistic chains have been playing an increasing role in industrial and distribution systems. Logistics can be defined as a process that deals with the planning, handling, and control of the storage of goods between the manufacturing point and the consumption point. In the past, goods were produced, stored and then delivered on demand. Nowadays, many companies do not work with stocks, using instead *cross-docking centers* [1]: the goods are transported from the suppliers to these cross-docking centers (e.g. airports), stored, and then shipped to the customers. The lack of storage may increase the delivery time, but it considerably reduces the volume of invested capital and increases the flexibility of the supply chain. The key issue is to deliver the goods in time by minimizing the stocks. The goods should be delivered at the correct date (not earlier or later) in order to ensure the customers' satisfaction.

A common scheduling process *pre-assigns* the components to the orders. In this strategy, when the components from an order are purchased, they are assigned to that specific order and thus can not be used by any other order. This scheduling strategy revealed to be poor: it is not robust to disturbances, such as delays or changes in the orders made by the clients, since it does not allow a redistribution of components among the orders. Therefore, a *dynamic assignment* strategy, allowing to exchange components between orders and optimizing the process, is necessary.

A dynamic scheduling method can be achieved using an ordering (sorting) of the orders. The components are then distributed according to the sorting until all the components are assigned or all the orders are fulfilled. The easiest way to sort the orders is to use

the *First Come First Served* principle, where the first incoming orders are delivered first. This method however does not account for important aspects of the orders, e.g. the actual and desired delivery times, and the priority of the orders. However, choosing a sorting using only one of these factors, will also be poor. The combination of criteria can be highly advantageous since each ordering principle fulfills only one particular aspect of the logistic process. Here, we use the combination of the ordering principles by means of fuzzy decision making in a fuzzy optimization framework, as presented in [2].

Nevertheless, a sorting based method will not make use of all optimization potential in the components exchange between the orders, as demonstrated in [3, 4], because it is still a centralized optimization method. McKay *et al* in [5], have already pointed out the need to an increasing replacement of centralized static scheduling strategies by dynamic distributed methods. A distributed approach explores better the optimization space. One way is to assign individual agents to the orders and let the population of agents interactively find an optimal scheduling solution [6]. The interaction between the agents is realized by exchanging information about quantity, desired date and arriving date, but in logistic problems, the quantity of information that has to be exchanged is very large. Multi-agent algorithms based on social insects can avoid this complexity. In [3], a new ant algorithm for logistic processes was proposed, using as objective function the delay minimization. Here, we extend the application of this algorithm combining it with the fuzzy weighted aggregation, where both the desired delivery dates and the priority of the orders are criteria to optimize. In this way, we present in this paper a distributed scheduling method where the decision making is based on a multiple criteria objective, by means of evolving ant colony optimization algorithms with fuzzy weighted aggregation.

The paper is organized as follows. Section 2 presents first the modeling of the logistic process as a birth and death process, followed by a global description of logistic processes. Section 3 describes the control strategies applied to logistic processes, such as a fuzzy decision making strategy and an ant colonies optimization algorithm. Section 4 presents a simulation example and the analysis of the obtained results. Finally, Section 5 concludes the paper and gives the main lines for future work.

2 The logistic process

Figure 1 presents a schematic representation of a logistic process, which can be described in probabilistic terms. In fact, the birth process of the system (arrival of new orders in a certain period of time) and the death process (delivery of orders per unit of time, or the time it took them to be processed), can be described by the classical theory of queuing processes [7]. For the process being studied, this theory asserts the Poisson distribution for the model of the birth process,

$$p(x, \lambda T) = \frac{(\lambda T)^x}{x!} e^{-\lambda T} \quad (1)$$

where x is the random variable *number of orders* and λT is the parameter indicating the probability of this event *occur on a certain time* T . The death process is modeled by the exponential distribution

$$p(T, \mu) = \mu e^{-\mu T} \quad (2)$$

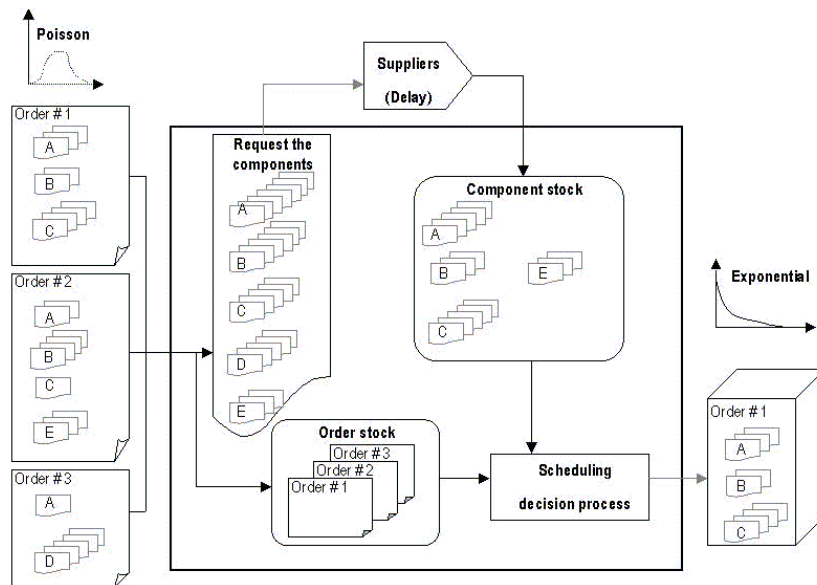


Figure 1: General representation of the logistic process

where the time T is the random variable and μ is the death rate. The process can be divided into five sequential steps:

- **Order arrival.** The client buys a product, called *order*, which is a set of one or more different items, the *components*. An order must contain a *desired delivery date*, which is the date required by the client for the order to be delivered. Each order has a *priority* index, which may reflect the importance of the client, or the profit of the order.
- **Component request.** The different components must be requested from the external suppliers. Each component is characterized by a certain *quantity*.
- **Component arrival.** Each component takes some time to be delivered to the logistic system. This time is called the *supplier delay*. After this time, the component is delivered to the so-called cross-docking places. A *component stock* list is built at these places, which contains the available components and their quantity.
- **Component assignment.** Usually the components are not all available at the same time. For this reason, the orders have to wait for all the required components to be available. This waiting list is called the *order stock*. Each order has its own desired delivery date. The decision process has to decide which orders are going to be delivered, taking into account the availability of their components. This is normally performed once per day. The focus of this paper is to optimize this component assignment process. If the delivery of two orders with different priority indices is mutually exclusive, then the order with the highest priority has to be delivered.
- **Order delivery** The order is delivered to the client, with a certain tardiness T , where *tardiness* stands for the difference between the delivery date and the desired date. The tardiness should always be close to zero.

3 Scheduling policies

The *component assignment* step is the key issue in logistic processes. The company can not influence the arrival rates of the orders (birth process), or the suppliers delay. The service rates (death process) of the orders are the only control variable by influencing the assignment of components to the orders. The control goal is to generate particular death rates for each order by some scheduling process using external information like desired times or internal information, such as the stock levels [8]. The next sections describe possible scheduling possibilities that can be used in a logistic system.

3.1 Pre-assignment (PA)

When the components arrive at the cross-docking center from the external suppliers, they are already assigned to specific orders. The components are stored there until all the missing components arrive and then the order is completed and can be delivered. This strategy can not deal efficiently with disturbances, e.g. a delay in the component arrival. This assignment method is called a static scheduling method.

3.2 Dispatching rules

These methods use some sorting between the orders based on a performance index. They are centralized dynamic strategies, since they allow a dynamic change of components between orders, but always using a centralized list. In our problem, the most interesting dispatching rules that can be used are:

- *First Desired First Delivered (FDFS)* - This method uses the desired delivery times for each order and sorts them in ascending order. The order with the earliest desired delivery time is the first in the list. The components in the stock are subsequently assigned to the orders in this list, starting from the first list entry. It is equivalent to the so called *Earlier Due Date* dispatching scheduling rule used for manufacturing systems [9].
- *Ordering by priority* - This methods uses a list sorted by priority indices. Some orders can have priority over others. An example of an important order is one with a large number of components required by a very important client. This order should have priority over others, because not only the order is big and the profit associated with it is large, but also the client should be very satisfied in order to assure possible future orders. Note that the company may have to delay less important orders slightly, in order to satisfy more important ones.

3.3 Fuzzy Weighted Aggregation (FWA)

This method can be seen also as a dispatching rule, since it is based on a list where the index is defined following a fuzzy optimization method. Fuzzy optimization is the name given to techniques that formulate optimization problems with flexible, approximate or uncertain criteria by using fuzzy sets. Fuzzy sets can be used to represent flexibility in the

goals and in the constraints. Hence, the criteria normally assumed to be crisp, are now assumed to have some flexibility, which can be exploited for improving the optimization objective. The general formulation for *fuzzy optimization* in the presence of flexible goals and constraints is given by

$$\begin{aligned} & \text{fuzzy maximize}_{\mathbf{x} \in X} [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_p(\mathbf{x})] \\ & \text{subject to} \quad g_i(\mathbf{x}) \lesssim 0, \quad i \in \{1, 2, \dots, q\}, \end{aligned} \quad (3)$$

where the \lesssim sign denotes a fuzzy satisfaction of the constraints and \lesssim denotes that $g_i(\mathbf{x}) \leq 0$ can be satisfied to a degree smaller than 1. The fuzzy maximization corresponds to achieving the highest possible aspiration level for the goals $f_1(\mathbf{x})$ to $f_p(\mathbf{x})$, given the fuzzy constraints of the problem. This optimization problem can be solved by using the approach of Bellman and Zadeh to fuzzy decision making [10].

Consider a decision making problem, like the logistic problem in this paper, where the decision alternatives are $\mathbf{x} \in X$. In the logistic process, \mathbf{x} corresponds to a certain assignment of components to the orders. A fuzzy goal $F_i, i = 1, \dots, p$, is a fuzzy subset of X . Similarly, fuzzy constraints $G_i, i = 1, \dots, q$, can be defined as fuzzy subsets of X . Goals and constraints can be treated as criteria C_i , with $i = 1, \dots, m$, and $m = p + q$ is the total number of criteria. The membership function of a criterion C_i is defined as $u_i(\mathbf{x})$, with $u_i : X \rightarrow [0, 1]$, and it indicates the degree of satisfaction of the decision criterion (goal or constraint) by the decision alternative $\mathbf{x} \in X$. The fuzzy decision D is defined as the confluence of the criteria, i.e.

$$D(\mathbf{x}) = u_1(\mathbf{x}) \circ \dots \circ u_p(\mathbf{x}) \circ u_{p+1}(\mathbf{x}) \circ \dots \circ u_m(\mathbf{x}), \quad (4)$$

where \circ denotes an aggregation operator for fuzzy sets. The optimal decision alternative \mathbf{x}^* is then the argument that maximizes the fuzzy decision, i.e.

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in X} D(\mathbf{x}). \quad (5)$$

When some trade off amongst the goals is desirable, the aggregation may be modeled by an averaging operation. The aggregation of goals allowing for some compensation between goals can be done using averaging operators. When the criteria are not equally important they can be combined using weighted fuzzy aggregation.

Weighted aggregation has been used quite extensively especially in fuzzy decision making, where the weights are used to represent the relative importance that the decision maker attaches to different decision criteria. Almost always an averaging operator has been used for the weighted aggregation, such as the generalized means, which can be naturally extended to weighted equivalents. The weighted generalized mean operator has been used in many fields, and it has been studied in the context of fuzzy set aggregation. This paper applies the generalized means to aggregate the several criteria in the logistic process. The confluence of the m criteria using the weighted generalized mean is defined as:

$$D_\gamma^w(\mathbf{x}) = \left(\sum_{i=1}^m w_i \cdot u_i(\mathbf{x})^\gamma \right)^{1/\gamma}. \quad (6)$$

for any $\gamma \in \mathbb{R} \setminus \{0\}$. The weight vector is defined as $\mathbf{w} = [w_1, w_2, \dots, w_m]$ and satisfies $\sum_{i=1}^m w_i = 1$. The sum of the weights w_i is one in order to define a relative importance between the weights.

The logistic process can be seen as a fuzzy optimization problem when it is formulated as follows. At each time period the optimization algorithm computes a fuzzy index (membership grade) for each order. This index combining the several criteria is computed for each order using the weighted generalized mean given in (6). This paper uses $\gamma = 1$, which is the arithmetic mean, in order to allow the compensation between criteria. For the logistic problem we defined two criteria: C_1 is the desired delivery date and C_2 is the priority. The weighted aggregation of C_1 and C_2 is given by

$$D(\mathbf{x}) = \sum_{i=1}^2 w_i \cdot u_i(\mathbf{x}). \quad (7)$$

The result of this weighted aggregation gives a membership grade for each order. The membership grades are then used to sort the orders as in the dispatching rules optimization algorithms.

3.4 Ant colonies in scheduling (ACO)

In [11] an optimization algorithm called the Ant System (AS), was presented to solve different classes of NP-hard problems. This algorithm tried to mimic the foraging behavior of real ants, i.e. the behavior of wandering in the search for food. After this introductory work, the artificial ants have been successfully used to solve first the Traveling Salesman Problem (TSP) [12] and then other NP hard optimization problems, including applications in quadratic assignment [13], vehicle routing [14], sequential ordering [15], routing in Internet networks [16] and so on. Dorigo and Di Caro proposed in [17] the general meta-heuristic Ant Colony Optimization (ACO), as a common framework for the existing applications and algorithms variants. Thus, every algorithm that follows the ACO meta-heuristic is called now an ACO algorithm. In [18], a global overview of the ACO meta-heuristic and its most relevant extensions can be found, as well as an overview of the most important applications where the algorithm has been used.

The optimization of the scheduling process described in Section 2 is also a NP-hard problem, for which the optimization heuristics need to process a large amount of information, such as the arrival and desired delivery dates, which can be extremely time-consuming. The ant colonies algorithm provides an optimization method where all the problem's information can be translated into an indirect form, the *pheromones*, and used by all the interacting agents in order to achieve a good global solution. In this way, a new framework of the ant colonies algorithm for logistic processes was introduced in [3]. In this paper, we introduce some fuzzy features in the algorithm, in order to extend the range of possible applications.

The idea to mimic the biological behavior of ants in order to solve optimization problems arose from the efficiency demonstrated by the ants to find always the shortest path to food sources. When an ant is searching for the nearest food source and comes across with several possible trails, it tends to choose the trail with the largest concentration of pheromones τ , with a certain probability p . After choosing the trail, it deposits another pheromone, increasing the concentration of pheromones in this trail. The ants return to the nest using always the same path, depositing in the way back another portion of

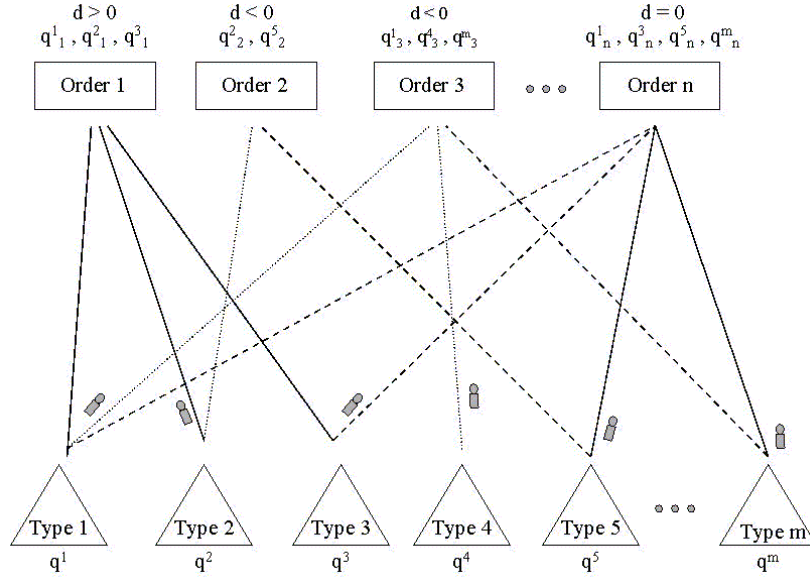


Figure 2: Example of an ant colony applied to the logistic process with pheromone concentration level on the trails: High (—), Medium (- -) and Low (· ·)

pheromone. Imagine then, that two ants at the same location choose two different trails at the same time. The pheromone concentration on the shortest way will increase faster than the other: the ant that chooses this way, will deposit more pheromones in a smaller period of time, because it returned earlier. If a whole colony of thousands of ants follows this behavior, soon the concentration of pheromone in the shortest path will be much higher than the concentration in other paths. Then the probability of choosing any other way will be very small, and only very few ants among the colony will fail to follow the shortest path. There is another phenomenon related with the pheromone concentration. Since it is a chemical substance, it tends to evaporate in the air, so the concentration of pheromones vanishes along the time. In this way, the concentration of the less used paths will be much lower than that on the most used ones, not only because the concentration increases in the other paths, but also because its own concentration decreases.

In the logistic process, two different sets of entities can be identified: the orders and the components. Each type of component stands for a type of food source and each order stands for a nest. The nests may require different types of food in different amounts, as orders require different types of components in different quantities. Different nests may require the same type of food, as different orders may require common types of components. Conceptually, the job of the ants from this colony is very different from the foraging job described in [11]. Here, there are m ants, one per food source. The job of these ants is to distribute the quantity of food in the food sources to the n nests. Once again, in every iteration t of the algorithm, the ants have to choose with some probability p which is the nest to visit first. Finally, they deposit a pheromone τ in the path from the food source to the nest. One tour is complete when all the m ants from each of the food source have visited all the n nests.

Each ant delivers an amount q_j^i from the total amount q^i of component $i \in \{1, \dots, m\}$ to an order $j \in \{1, \dots, n\}$. Since there are several nests to visit, the ant k chooses the path

to a particular nest with a probability

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha \cdot \eta_{ij}^\beta}{\sum_{r \notin \Gamma^k} \tau_{ir}^\alpha \cdot \eta_{ir}^\beta} & \text{if } j \notin \Gamma^k \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where τ_{ij} is the amount of pheromone connecting component type i to order j , η_{ij} is a visibility function and the Γ^k is the tabu list of the k^{th} ant. The tabu list is the list of the orders that the ant k has already visited plus the orders that do not need component type i (thus the visit can be avoided). The visibility function η_{ij} is defined as:

$$\eta_{ij} = e^{T_j} \quad (9)$$

where T_j is the *tardiness* of order j . This function reflects the objective of providing the ants with information regarding the delay of a certain order, since the objective of the problem is not only to fulfill the highest number of orders per day, but also to deliver them on the desired day (not before and not after). Orders that should be delivered on the actual date have visibility $\eta_{ij} = 1$. Orders that can still wait ($T < 0$) will have a lower visibility, and orders that are already delayed ($T > 0$) will have a higher one, so the ants will preferably feed delayed orders. The pheromone trails indicate the prior attempts from other ants. The parameters α and β express the relative importance of trail pheromone (experience) with visibility (knowledge), respectively.

After choosing the order to visit, the ant will deposit a pheromone on the trail connecting the type of component i to the order j . At this point, the ant can find two different situations:

1. When the total amount of components q^i of component type i is not enough to fulfill the amount of components q_j^i of same type needed by order j , the ant should not reinforce this path, so the pheromone value of this connection remains the same;
2. When sufficient components q^i exist to deliver to order j , the ant should reinforce this path, adding a pheromone value to the existing pheromone concentration on this path.

The local update of the pheromone concentration is then given by

$$\delta_{ij}^k(t) = \begin{cases} \tau_c & \text{if } q_j^i \leq q^i \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

where τ_c is some small constant. In this way, the pheromones are locally updated with the information considering the amount of components i to deliver to the order j . After the ant visits order j , we decrement q^i by q_{ij} before the next ant from resource i visits order $j + 1$. Note also that the ant always delivers all the q_j^i components that order j needs, to avoid the deadlock situation where all orders have part of the needed components, but none has the complete amount. In this way, the algorithm only finds feasible solutions. At the end of a complete tour the change of pheromones in all paths is given by

$$\Delta\tau_{ij} = \sum_{k=1}^m \delta_{ij}^k. \quad (11)$$

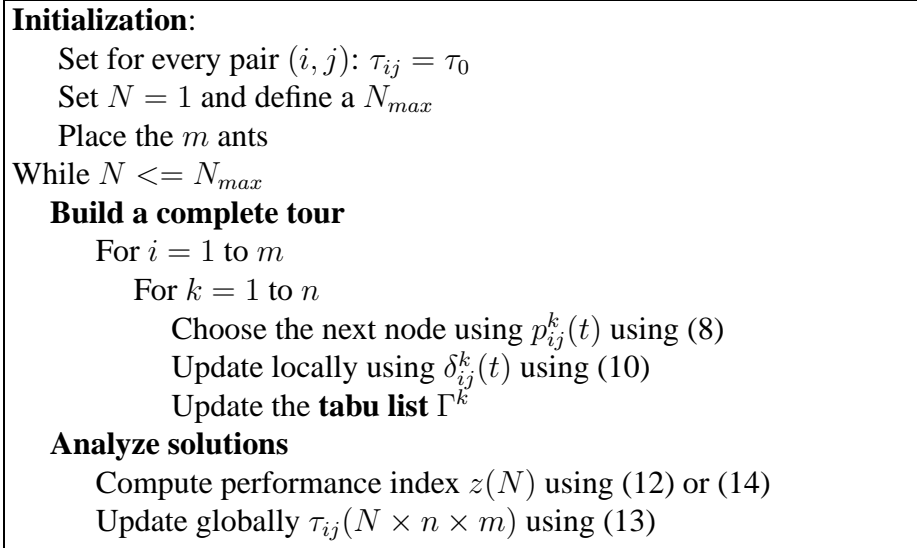


Figure 3: Ant colonies optimization algorithm for logistic processes

At this point, when all ants have delivered all possible components, the solution can be evaluated using a performance index

$$z = \sum_{j=1}^n a_j, \text{ where } \begin{cases} a_j = 1 & \text{if } T_j = 0 \\ a_j = 0 & \text{otherwise} \end{cases} \quad (12)$$

where n is the number of orders and T_j is the tardiness of order j . This index is used to update globally the pheromones.

At each tour N of the algorithm (where each tour has $n \times m$ iterations t), a z is computed and stored in the set $Z = \{z(1), \dots, z(N)\}$. If $z(N)$ is higher than the previous $z \in Z$, then the actual solution has improved and the used pheromones should be increased. If it is worse, they should be reduced. This is done once again by the global pheromone update, where the evaporation coefficient $(1 - \rho)$, $\rho \in [0, 1]$ avoids the solution to stagnate. In this way, the global pheromone update done after each complete tour N is the following:

$$\tau_{ij}(N \times n \times m) = \begin{cases} \tau_{ij}(n \times m) \times \rho + \Delta\tau_{ij} & \text{if } z(N) \geq \max Z \\ \tau_{ij}(n \times m) \times \rho & \text{otherwise} \end{cases} \quad (13)$$

Figure 2 schematically represents this framework. This particular framework of the ant colonies algorithm can be implemented using the general algorithm described in Fig. 3. Notice that the algorithm is initialized with a τ_0 value of pheromone concentration in every path, and that the algorithm runs a maximum number of tours N_{max} .

3.5 Evolved Ant Colony Optimization with Fuzzy Weighted aggregation (ACO & FWA)

The only difference to the previous optimization method, is that the objective function to optimize presented in (12) is replaced by the fuzzy decision function

$$z = \sum_{j=1}^n D_j(x) \quad (14)$$

where n is the number of orders and $D_j(x)$ is the fuzzy index of order j as defined in (7). In this way, instead of considering the optimization only according to tardiness, the artificial ants will optimize also according to priorities.

4 Experiments

We consider here a very simple logistic system with the following parameters: the number of orders arriving at each day follows a Poisson distribution as described in (1), with $\lambda T = 10$. The example considers that each order can have at the most 5 different types of components and the quantity for each component varies between 1 and 20. Each type of component has a constant supplier delay that varies between 1 and 7. For each order, a desired date is generated using an exponential distribution as in (2) with $\mu = 7$. This means that in average, the components will arrive within the desired delivery date. The orders have also a priority index, which can be $\{0.25, 0.5, 0.75, 1\}$ where 0.25 means normal order and 1 means important order. The majority of the orders have normal priority (0.25), but there are some with the "important" label (1). There exist also very few with intermediate levels (0.5 and 0.75). These indices are defined using a normal distribution with average 0.25 and variance 0.75 [19]. The simulation is done for an interval of 1 month. At each day the system adds the components arriving at that day to the components stock list (except for the PA strategy). These components are then distributed to the other scheduling rules defined in Section 3.

To measure the performance of the different methods, we use tables representing the number of orders classified by type *tardiness* T : if $T < 0$, it means the orders were delivered before the desired date; if $T = 0$, it means they were delivered at the correct date; if $T > 0$, it means they were delivered after the correct date. The tables indicate also the minimum and maximum tardiness as an indicator of the variance. The orders are divided into 4 groups, referring to priority indices. The objective is not only to deliver the maximum number of orders with $T = 0$, but also to deliver as much as possible orders with priority 1.

In the next sections, we present the results of the simulations divided by scheduling policies: pre-assignment (PA), dispatching rules (FDFS and FWA) and distributed approach (ACO and ACO & WFA).

4.1 Pre-assignment (PA)

As explained in Section 2, in this method, the components are pre-assigned to the orders, so the orders will be delivered as soon as the last component arrives to the docking center. While it waits for the missing components to arrive, all the other components have to stay there. Even if an order B is missing a component that exists in the stock assigned to an order A , and the order B could be delivered on time if it received this component, the component is not assigned to order B . The result, as it can be seen in Table 1, is that very few orders are delivered at the correct date. In terms of priorities, this method is also inefficient: there are more orders delivered at the correct date with low priority than with high priority. In this way we conclude that this method is very poor to achieve the $T = 0$ objective, as well as to respect the priorities.

Table 1: Pre-assignment (PA)

Method	Priority	$T < 0$	$T = 0$	$T > 0$	$\min(T)$	$\max(T)$
PA	0.25	26	20	56	-15	4
	0.5	8	7	14	-18	4
	0.75	8	5	9	-4	4
	1	13	8	24	-11	4

Table 2: FDFS strategy

Method	Priority	$T < 0$	$T = 0$	$T > 0$	$\min(T)$	$\max(T)$
FDFS	0.25	56	38	18	-12	4
	0.5	16	9	7	-13	3
	0.75	12	9	2	-12	4
	1	23	8	7	-7	3

4.2 Dispatching rules (FDFS)

It is expectable that any dynamic scheduling, where the components can be exchanged between the orders, performs better than the PA method. Moreover, if the method follows a desired delivery date sorting, it is expectable that the number of orders delivered at the correct day increases. In Table 2 we confirm this expectation: the number of orders delivered at the correct date increases for all types of priorities, except for priority of type 1. In fact, when priority is 1, this method delivers less orders than the PA method. This shows that, although in terms of tardiness the performance is better, in terms of priorities the results are worse.

4.3 Fuzzy optimization (FWA)

To apply the generic framework defined in Section 3.3, the weighted aggregation in (7) was defined as follows: since the sum of the weights w_i is one, the weights are defined using a parameter $\epsilon = w_1$ and consequently $(1 - \epsilon) = w_2$. Thus, the aggregation of the criteria is given by

$$D(\mathbf{x}) = \epsilon u_1 + (1 - \epsilon) u_2. \quad (15)$$

The criterion C_1 is the tardiness, so the proposed membership function u_1 is the one presented in Fig. 4. The membership function u_1 is very small before the desired date (following an exponential function), reaching the value 0.1 when the actual day is the desired day. Then, between 0 and 10, the function grows linearly to one, and it remains constant then. This means that an order that is already 10 days delayed should be delivered immediately. The criterion C_2 , the priority, is already a fuzzy number. The used value of $\epsilon = 0.8$, derived from the study presented in [2].

With this strategy, where both tardiness and priority are taken into account in a fuzzy index, the results improve in terms of priorities, as shown in Table 3. With this method, it was possible to deliver more orders with the highest priority in the correct date. However, the result implies that for the other indices of priority, the performance is worse in terms of tardiness.

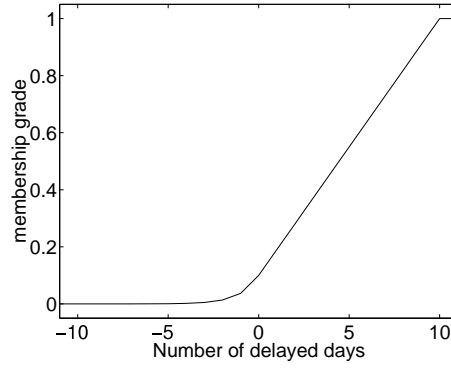


Figure 4: membership function u_1 of the tardiness (delayed days)

Table 3: FWA strategy

Method	Priority	$T < 0$	$T = 0$	$T > 0$	$\min(T)$	$\max(T)$
FWA	0.25	28	21	39	-13	4
	0.5	14	8	11	-14	3
	0.75	14	5	2	-8	4
	1	45	11	6	-13	4

4.4 Ant colony optimization (ACO)

The distributed strategies are expected to perform even better than the dispatching rules. The ACO method is used as presented in [3], where the $T = 0$ objective is the only criteria considered in the optimization. The algorithm uses the following parameters: $\alpha = 1$, $\beta = 10$, $\rho = 0.5$, $\tau_0 = 1$, $\tau_c = 1$ and $N_{max} = 20$.

Table 4 confirms that in general, for every type of priorities, more orders were delivered at the correct date ($T = 0$), and even when there is tardiness, negative or positive, the spread is not so wide. We conclude that this method, without taking into account the priority index, achieves very good results in terms of orders delivered at the correct date ($T = 0$). However, we can see that the number of orders delivered with higher priorities is smaller for this method than for the FWA approach. It is true that this method is able to deliver many orders at the correct date, but most of them are the ones with the lowest priority. In this sense, the results are not so impressive.

4.5 Evolved Ant colony Optimization (ACO & FWA)

Our objective is to prove that using ants to optimize a fuzzy optimization function, taken into account priority indices, will outperform all the previous strategies and achieves the

Table 4: Ant Colony Optimization (ACO)

Method	Priority	$T < 0$	$T = 0$	$T > 0$	$\min(T)$	$\max(T)$
ACO	0.25	41	57	10	-9	5
	0.5	9	11	5	-5	4
	0.75	10	5	5	-3	4
	1	18	17	8	-5	5

Table 5: ACO & WFA strategy

Method	Priority	$T < 0$	$T = 0$	$T > 0$	$\min(T)$	$\max(T)$
ACO	0.25	46	26	16	-8	4
&	0.5	16	14	6	-4	3
FWA	0.75	13	6	3	-3	2
	1	23	20	14	-6	4

complex objective of $T = 0$ for the most important orders. Both the ACO and the FWA were applied using the parameters defined previously. The results are presented in Table 5. For every degree of priority except the lowest one, there are more orders delivered at the correct day ($T = 0$), without changing the variance around $T = 0$. It is visible that the trade-off done between number of orders delivered with $T = 0$ and the priority indices, is done at the expenses of the lowest priority indices.

5 Conclusions

This paper applies an ant colony optimization methodology using the fuzzy decision making framework to optimize the scheduling of a logistic process. This type of process can be modeled probabilistically as birth and death processes. The dynamic assignment of components to orders is the key issue in the optimization of logistic processes, and the ant colony optimization algorithm is able to do it in a distributed approach. The use of multiple optimization criteria is solved using a weighted fuzzy aggregation in a fuzzy decision making environment. A simple but illustrative example shows that the proposed evolved method can be applied with good results to this type of processes. The example considered is a simplified example of a real world logistic process at Fujitsu-Siemens Computers [20]. As the next step we want to apply our methods to the real full scale logistic process.

References

- [1] Stephen F. Smith Jayashankar M. Swaminathan and Norman M. Sadeh. Modeling supply chain dynamics: A multiagent approach. *Decision Sciences Journal*, 29(3):607–632, 1998.
- [2] J.M. Sousa, Rainer Palm, C.A. Silva, and Thomas Runkler. Fuzzy optimization of logistic processes. In *Proceedings of World Congress on Computational Intelligence WCCI-2002*, Honolulu, USA, May 2002.
- [3] C.A. Silva, Thomas Runkler, J.M. Sousa, and Rainer Palm. Optimization of logistic processes using ant colonies. In *Proceedings of Agent-Based Simulation 3*, pages 143–148, Passau, Germany, April 2002.
- [4] C.A. Silva, Thomas Runkler, J.M. Sousa, and Rainer Palm. Ant colonies as logistic processes optimizers. In *Proceedings of Ants2002, Brussels, Belgium*, Brussels, Belgium, September 2002.
- [5] K. McKay, M. Pinedo, and S. Webster. A practice-focused agenda for production scheduling research. *Production and Operations Management*, 10, 2001.

- [6] Rainer Palm and Thomas Runkler. Multi-agent control of queuing processes. In *Proceedings of the 15th IFAC Congress on Automatic Control*, Barcelona, Spain, 2002.
- [7] R.W. Wolff. *Stochastic Modeling and the Theory of Queues*. Prentice-Hall, London, 1989.
- [8] Rainer Palm and Thomas Runkler. Decentralized control of hybrid systems. In *Proceedings of the Fifth International symposium on autonomous decentralized systems (ISADS)*, Dallas, USA, 2001.
- [9] Michael Pinedo. *Scheduling: Theory, Algorithms, and Systems*. Prentice Hall, second edition, 2002.
- [10] R. E. Bellman and Lotfi A. Zadeh. Decision-making in a fuzzy environment. *Management Science*, 17(4):141–164, December 1970.
- [11] Marco Dorigo, Vittorio Maniezzo, and Alberto Coloni. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 26(1):29–41, 1996.
- [12] Luca Maria Gambardella and Marco Dorigo. Solving symmetric and asymmetric tsp by ant colonies. In *Proceedings of the 1996 IEEE Conference on Evolutionary Computation (ICEC'96)*, pages 622–627, Piscataey,NJ, 1996.
- [13] L. Gambardella, E. Taillard, and M. Dorigo. Ant colonies for the quadratic assignment problem. *Journal of the Operational Research Society*, 50(2):167–176, 1999.
- [14] Luca Maria Gambardella, Éric Taillard, and Giovanni Agazzi. MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. In David Corne, Marco Dorigo, and Fred Glover, editors, *New Ideas in Optimization*, pages 63–76. McGraw-Hill, 1999.
- [15] L. M. Gambardella and M. Dorigo. HAS-SOP: Hybrid ant system for the sequential ordering problem. Technical Report IDSIA-11-97, 1, 1997.
- [16] Gianni Di Caro and Marco Dorigo. Antnet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317–365, 1998.
- [17] Marco Dorigo and Gianni Di Caro. The ant colony optimization meta-heuristic. In David Corne, Marco Dorigo, and Fred Glover, editors, *New Ideas in Optimization*, pages 11–32. McGraw-Hill, London, 1999.
- [18] M. Dorigo and T. Stützle. The ant colony optimization metaheuristic: Algorithms, applications, and advances. *Handbook of Metaheuristics*, F. Glover and G. Kochenberger.
- [19] Sheldon M. Ross. *Introduction to Probability and Statistics for Engineers and Scientists*. John Wiley & Sons, Inc, 1987.
- [20] Carlos A. Silva. Systemanalyse - Fujitsu Siemens Computers. Technical Report NIVELLI-S1, Siemens AG, Corporate Technology, Neural Computation Department, October 2001.

Ein Zwei-Welten-Modell für die Prozeßanalyse und Klassifikation

Christian Kuhn

Technische Universität Ilmenau,
Fakultät für Informatik und Automatisierung
P.F.: 100565, D-98684 Ilmenau
Tel.: (03677) 691467
Fax: (03677) 691434
E-Mail: christian.kuhn@tu-ilmenau.de

Zusammenfassung

In diesem Beitrag wird ausgehend von der Weltanschauung des Realismus ein Zwei-Welten-Modell entwickelt, das sich zur Lösung von Aufgaben der Prozeßanalyse und Klassifikation eignet. Im Gegensatz zum klassischen Objektmodell erlaubt ein alternatives Objektmodell die Modellierung von gegensätzlichen Objekten durch den Gebrauch semantischer Dipole.

Schlüsselwörter: Menge, Dipol, Realismus, Zwei-Welten-Modell, Noumene, Phänomene, Zufall, Wahrscheinlichkeit, Bestimmtheit, Gewißheit

1 Einführung

1.1 Ziel des Modells

Täglich wird der Mensch mit der Situation konfrontiert, sich in dieser Welt zurechtzufinden. Er trifft dabei auf bekannte Situationen, für die er schon Strategien entwickelt hat, um sie zu meistern. Er wird aber auch — und das besonders in jungen Jahren — mit Situationen konfrontiert, für die solche Strategien noch fehlen. Gerade diese Situationen fordern ihn. Mitunter ist er hier auf Hilfe angewiesen, oder schafft es mit Hilfe seines Verstandes, solche Strategien zu entwickeln. An den Reaktionen der Umwelt kann er die Wirksamkeit einer solchen Strategie ermessen und sie gegebenenfalls verifizieren. Tritt eine ähnliche Situation später wieder auf, kann er auf Erfahrungen zurückgreifen, d. h. eine erprobte Strategie anwenden, die bei einer ähnlichen früheren Situation zum Erfolg geführt hat. Derartige Strategien stellen einen Teil seines Wissens dar. Der Mensch ist bestrebt, Erkenntnisse über die Welt zu gewinnen, um sich in ihr besser zurechtfinden zu können und eigene, immer effizientere Strategien zu entwickeln, die immer weniger Aufwand erfordern und die sich schließlich automatisieren lassen. Die Selbstreflexion seines Handelns unterstützt das Verständnis intuitiver Prozesse. Das ordnende Prinzip des menschlichen Bewußtseins, welches für die Verarbeitung neuer Eindrücke notwendig ist, läßt sich auch auf Problemstellungen der Prozeßanalyse und Klassifikation übertragen.

Ein Modell, das einen Teil des menschlichen Erkenntnisprozesses widerspiegelt, wird in diesem Beitrag vorgestellt und kann zur Lösung von Problemen der Prozeßanalyse und Klassifikation herangezogen werden. Zwei unterschiedliche Welten – die Prozeßwelt und die Analysewelt – bilden die Grundlage dieses Modells. Diese Welten sind

gleichbedeutend mit der Realität und der Welt des menschlichen Bewußtseins. Ziel ist die Abbildung von Dingen der Realität bzw. des Prozesses eine zweite, künstliche Analysewelt, in der nun die Verfahren der Prozeßanalyse und Klassifikation für die weitere Verarbeitung der empirischen Prozeßbeobachtungen sorgen.

1.2 Realismus als erkenntnistheoretische Grundlage

Ausgangspunkt dieser Betrachtungen stellt die realistische Weltanschauung dar [1]. Nach ihr existiert die Wirklichkeit unabhängig von der menschlichen Erfahrung und dem menschlichen Bewußtsein. Mit den sinnlichen Erfahrungen (Wahrnehmungen über die Sinnesorgane) erfolgt der Zugang zu einer bewußtseinsunabhängigen Wirklichkeit. Je nachdem, wie der Zugang zur Wirklichkeit erfolgt, lassen sich verschiedene Formen des Realismus unterscheiden. Die wichtigsten Strömungen umfassen den *empirischen Realismus*, der eine Erfahrungswelt anerkennt, die zwar unabhängig von den einzelnen erkennenden Wesen ist, aber von der Möglichkeit abhängt, daß überhaupt erkennende Wesen existieren. Eine tiefere Bedeutung für die Verfahren der automatischen Klassifikation hat der *repräsentative Realismus*. Nach ihm wird die Welt des menschlichen Bewußtseins zwar von der physischen Wirklichkeit verursacht, diese erlaubt jedoch niemals sichere Erkenntnisse, sondern nur Repräsentationen der Wirklichkeit, wie diese beispielsweise durch Sinneseindrücke gegeben sind. Ein Vergleich dieser Repräsentationen mit der Wirklichkeit bleibt dem Menschen verwehrt.

1.3 Bewußtsein

Die Orientierung des Menschen in der Realität wird durch seinen Intellekt bzw. sein Bewußtsein gesteuert, in welchem die Reflexion der Realität erfolgt. Die Welt der Realität (die Wirklichkeit) soll mit W_R , die des menschlichen Bewußtseins mit W_C bezeichnet werden. Geht man von einer realistischen Weltanschauung aus, erfolgt die Projektion der Realität in die Welt des Bewußtseins mit Hilfe von Sinnesorganen, die die acht bekannten Sinne des Menschen, nämlich *Riechen*, *Schmecken*, *Fühlen*, *Sehen*, *Hören* sowie den *Gleichgewichts-*, *Temperatur-* und *Schmerzsinn* repräsentieren [2], vgl. Abbildung 1. Da die Sinnesorgane nur über eine begrenzte

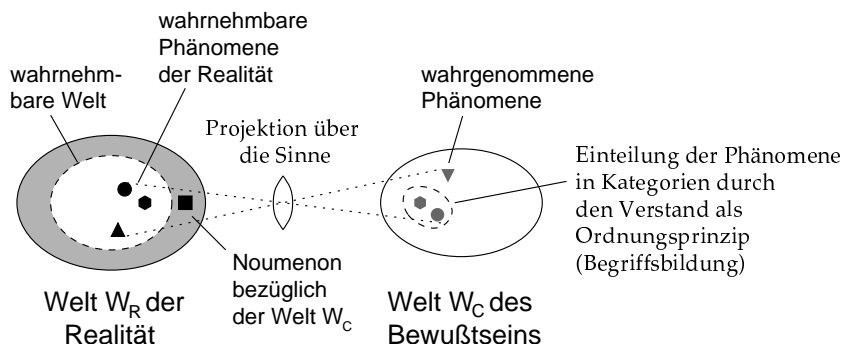


Abbildung 1: Wahrnehmung von Phänomenen der Realität im Bewußtsein

Wahrnehmungsfähigkeit verfügen und auch die Zahl der Neuronen im menschlichen Gehirn und damit der menschliche Verstand begrenzt ist, wird im Bewußtsein nur

ein Teil der Realität abgebildet. Die Projektion kann dabei nur so erfolgen, wie die Sinnesorgane es ermöglichen.

Nach Kant wird die Verarbeitung der Sinneswahrnehmungen durch den Verstand vorgenommen, der die Funktion eines zentralen Ordnungsprinzips innehat [3]. Dieses Ordnungsprinzip nimmt zur weiteren Verarbeitung der Sinneswahrnehmungen Abstraktionen vor, die u. a. zur Bildung neuer Begriffe führen. Begriffe werden durch den Verstand bestimmten Kategorien zugeordnet. Dem Verstand kann jedoch auch die Aufgabe zugeschrieben werden, aus Sinneswahrnehmungen eine Welt — die bewußte Welt — aktiv zu konstruieren. Zu dieser Konstruktion bedient er sich zahlreicher Gestaltgesetze, die Forschungsgegenstand der kognitiven Psychologie bilden [4]. Die wahrgenommene Welt des Bewußtseins W_C korreliert dabei um so besser mit der Welt der Realität W_R , je umfangreicher die und detailgetreuer die Beschreibung der Realität ist. Dies ist einerseits vom Wissen über die Realität und dem Umfang der verarbeiteten Sinneseindrücke abhängig. Begrenzt wird diese Korrelation aber auch durch die Endlichkeit des Verstandes, da das menschliche Gehirn zwar aus einer großen, aber dennoch endlichen Zahl von Neuronen besteht und damit dem Umfang der relationalen Beschreibung der Welt Grenzen gesetzt sind.

1.4 Dinge in einer Welt

Nach Platon werden die Dinge, die unabhängig vom menschlichen Geist existieren, als *Noumene* bezeichnet [5]. In Anlehnung an diese Bezeichnung sollen unerkannte Dinge in W_R , die (über die Sinne) keinen Zugang zum menschlichen Bewußtsein finden, als Noumene bezeichnet werden. Über Noumene liegen keine Kenntnisse vor, so daß auch Aussagen über Noumene bezüglich der Existenz oder ihren Eigenschaften rein spekulativ bleiben.

Im Gegensatz dazu werden hier die Dinge der Realität W_R , die sich den menschlichen Sinnen durch ihre Erscheinung darbieten, als *Phänomene* bezeichnet. Phänomene bilden das empirische Wissen einer menschlichen Intelligenz über die Dinge der Realität. Phänomene zeichnen sich durch innere Eigenschaften, die das Wesen von Phänomenen charakterisieren, und äußere Eigenschaften, wie Form, Gestalt usw. aus. Diese Auffassung von der Erkennbarkeit der Welt entspricht dem repräsentativen Realismus.

2 Zwei-Welten-Modell für Probleme der Prozeßanalyse und Klassifikation

2.1 Struktur des Zwei-Welten-Modells

Diese Anschauung spiegelt die Arbeitsweise der automatischen Klassifikation und Prozeßerkennung wider, so daß dieses Prinzip auch auf derartige Problemstellungen angewandt werden kann. Wir gehen von einer Prozeßwelt W_P aus, die mit der objektiven Realität vergleichbar ist. In dieser Prozeßwelt finden reale Prozesse statt. Aus dieser Prozeßwelt lassen sich bestimmte Phänomene in eine zweite, künstliche Welt — die hier als *Analysewelt* W_A bezeichnet werden soll — transformieren. Der

Zugang zur Prozeßwelt ausgehend von der Analysewelt erfolgt anstelle von Sinnesorganen über *Sensoren*. Im Anschluß an die Signalwandlung durch die Sensoren dienen Signaltransformationen zur Projektion der durch den Prozeß verursachten Phänomene in die Analysewelt. In der Analysewelt haben dann Prozeßanalyse- und Klassifikationsmethoden die Funktion des dem menschlichen Verstand entsprechenden Ordnungsprinzips. Da auch die Sensoren nur einen begrenzten Ausschnitt aus der Prozeßwelt wiedergeben können, ist das Wissen über den Prozeß in der Analysewelt keineswegs vollständig, so daß aus den verfügbaren Prozeßinformationen nicht notwendigerweise zuverlässig auf den Prozeß geschlossen werden kann. Das Prinzip der im Abschnitt 1.2 erwähnten repräsentativen Realität führt zur Konstruktion des Zwei-Welten-Modells nach Abbildung 2.

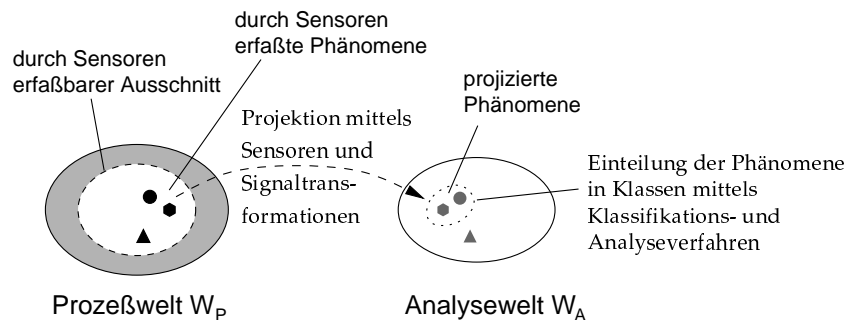


Abbildung 2: Zwei-Welten-Modell für die Prozeßanalyse und Klassifikation

2.2 Noumene und Phänomene

2.2.1 Noumene

Auch dieses Modell kennt Noumene und Phänomene. Dinge, die außerhalb der Analysewelt W_A existieren, sind *Noumene* bezüglich W_A . Da keine Beschreibung über Noumene vorliegt, ist nichts näheres über sie bekannt. Noumene sind somit Phänomene der Prozeßwelt W_P , die keinen Zugang zur Analysewelt W_A finden, also nicht durch geeignete Sensoren erfaßt werden. Es lassen sich weder Aussagen über ihre Existenz noch über bestimmte Eigenschaften treffen. Noumene können *offenbart werden*, indem sie über Sensoren in die Welt W_A projiziert werden. Offenbarte Noumene erscheinen in der Welt W_A als Phänomene.

2.2.2 Phänomene

Phänomene in W_A sind die Folge eines kausalen Zusammenhanges und werden durch ein offenbartes Noumenon (ein projiziertes Phänomen aus W_P) verursacht. Sie repräsentieren eine Erscheinung in der Welt W_A . Phänomene zeichnen sich durch eine Menge an Prädikaten aus.¹ Die Gesamtheit der Phänomene bildet das empirische Wissen über den Prozeß in W_P .

¹Als Prädikate sollen hier alle Dinge und Sachverhalte bezeichnet werden, mit denen sich ein Phänomen charakterisieren läßt, wie Merkmale, Relationen, Aktionen, etc.

Elementare Prädikate. Phänomene P besitzen zwei elementare Eigenschaften, die Existenz und die Identität. Diese Eigenschaften besitzen alle Phänomene unabhängig von ihren sonstigen Prädikaten. Die *Existenz* F_1 stellt eine notwendige Bedingung für Phänomene in einer Welt dar. Die Eigenschaft der Existenz von Phänomenen in W_A ist an die Offenbarung von Noumenen aus W_P (also die Projektion von erfaßten Phänomenen aus W_P) gekoppelt. Die folgende semantische Gleichung beschreibt die Existenz eines Phänomens P in einer Welt W

$$\frac{F_1(P)}{W} = \top. \quad (1)$$

Sie drückt aus, daß die Welt, in der sich das Phänomen befindet und die man durch den Ausdruck $F_1(P)$ erhält, der betrachteten Welt W entspricht. Im Vergleich zu einer anderen Welt $\neg W$ erhält man

$$\frac{F_1(P)}{\neg W} = \perp. \quad (2)$$

Die *Identität* F_2 ist die zweite elementare Eigenschaft der Phänomene. Jedes Phänomen besitzt eine es von anderen Phänomenen unterscheidende Identität. Die Ausprägung dieser Eigenschaft ist das Phänomen selbst. Die Identität eines Phänomens kann als *primäres Unterscheidungsmerkmal* gelten, da sich die Identität von zwei nicht identischen Phänomenen immer unterscheidet. Die Identität eines Phänomens in Bezug auf sich selbst ergibt

$$\frac{F_2(P)}{P} = \top. \quad (3)$$

In Bezug auf ein anderes Phänomen erhält man den Ausdruck

$$\frac{F_2(P)}{\neg P} = \perp. \quad (4)$$

Sekundäre Prädikate. Sekundäre Prädikate geben Aufschluß über die Erscheinung eines Phänomens. Sie können zum einen innere Eigenschaften eines Phänomens widerspiegeln. Dies können beispielsweise Zustände sein. Zu ihnen zählen aber auch die äußeren Eigenschaften eines Phänomens, wie Form oder die Lage im Raum, vgl. Abbildung 3. Die Menge aller sekundären Prädikate, die zur Beschreibung des Phänomens zur Verfügung stehen, wird als *Habitus* \mathcal{H} bezeichnet. Hingegen soll als *Phänotyp* $\mathcal{H}(P)$ das konkrete Erscheinungsbild eines bestimmten Phänomens P bezeichnet werden. Der Phänotyp entspricht dem Habitus eines Phänomens mit seinen für das Phänomen charakteristischen Ausprägungen der Prädikate. Für den Fall, daß durch einen gegebenen Phänotyp ein Phänomen eindeutig charakterisiert werden kann, läßt sich der Phänotyp $\mathcal{H}(P)$ eines Phänomens auch als sekundäres Unterscheidungsmerkmal benutzen.

Der Habitus eines Phänomens ist an dessen Welt geknüpft. Entsprechend der Weltanschauung des repräsentativen Realismus ist der vollständige Habitus \mathcal{H} eines Phänomens in W_P mit der Gesamtheit seiner verfügbaren Merkmale jedoch meist nicht bekannt, sondern über Sensoren ist nur eine Teilmenge aller Prädikate $\mathcal{H}_{P_v} \subset \mathcal{H}_P$, die mit der projizierten Merkmalmenge $\mathcal{H}_{P_v} \rightarrow \mathcal{H}_A$ korrespondiert, sichtbar. Analoges gilt für den Phänotyp. Dies bedeutet, daß sich aus der Welt W_A heraus keine Aussage über den vollständigen Habitus eines Phänomens der Welt W_P treffen läßt.

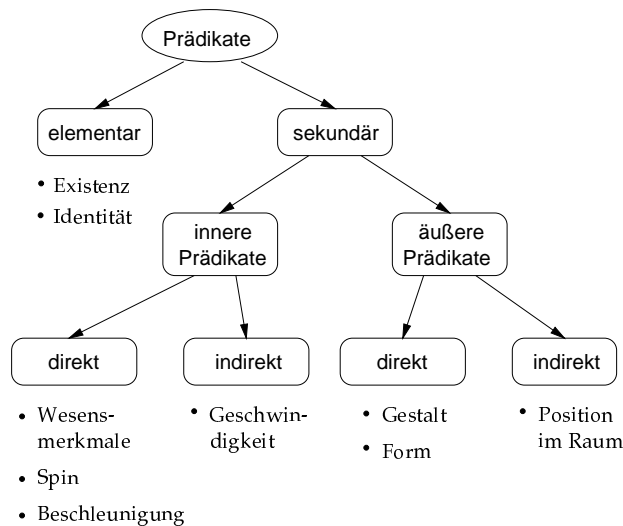


Abbildung 3: Prädikate zur Beschreibung von Phänomenen

Dies kann zu der fehlerhaften Annahme führen, daß zwei Phänomene der Welt W_A identisch sind, wenn die Phänotypen in dieser Welt übereinstimmen, sich jedoch in der Welt W_P unterscheiden. So ist es möglich, daß ein Betrachter zwei Sterne am Nachthimmel aufgrund der fehlenden Tiefeninformation als dicht beieinanderliegend wahrnimmt, diese aber in Wirklichkeit sehr weit voneinander entfernt sein können. Der Phänotyp der in die Analysewelt W_A projizierten Phänomene kann deshalb auch nicht als sicheres Unterscheidungsmerkmal herangezogen werden. Eine Klassifikation von Phänomenen in W_A ist deshalb immer mit einer Unsicherheit behaftet. Abbildung 4 verdeutlicht den Zusammenhang zwischen dem gesamten und erkennbaren Habitus.

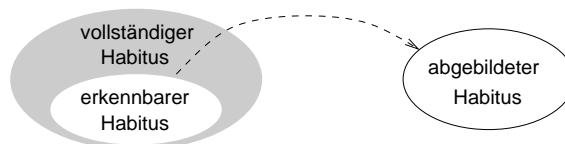


Abbildung 4: Gesamter und erkennbarer Habitus eines Phänomens aus W_P

Die Tatsache, daß nur eine Teilmenge aller Merkmale eines Phänomens aus W_P bei der Projektion in die Analysewelt berücksichtigt wird, beeinflusst die Merkmalselektion der automatischen Klassifikation maßgeblich. Eine ungünstige Merkmalswahl kann durch den Informationsverlust die Projektion beeinträchtigen, so daß projizierte Phänomene, die eigentlich verschiedene Prozeßzustände und -situationen in W_P repräsentieren, sich in der Analysewelt W_A überlagern können und fälschlicherweise eine Ähnlichkeit aufweisen. Abbildung 5 zeigt die Projektion von zweidimensionalen Objekten (Klassengebieten) der Prozeßwelt W_P auf die entsprechenden Merkmalsachsen, die den Analysewelten W_{A1} und W_{A2} entsprechen.

2.3 Die Analysewelt

Die meisten Untersuchungen verlangen eine Analysewelt, die aus Zeit und Raum gebildet wird. In diesem Fall soll der Raum durch einen orthogonalen Merkmalraum

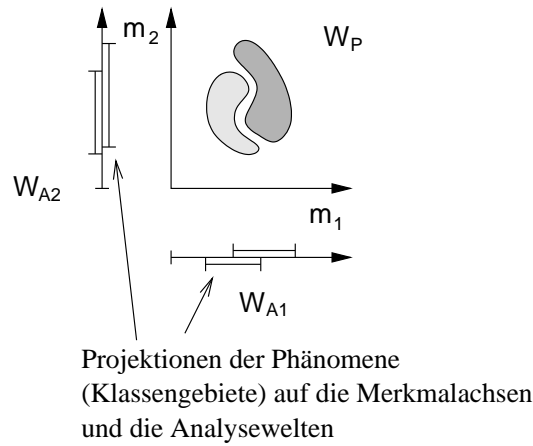


Abbildung 5: Informationsverlust durch Projektion von höherdimensionalen Phänomenen auf Unterräume

\mathcal{M} gebildet werden. Es ist nicht notwendig, den gesamten Merkmalraum für die Analyse auszuwerten. In den meisten Fällen ist es ausreichend, nur den Teil des Raumes, der die gesamte Objektmenge einschließt, für die weiteren Untersuchungen zu berücksichtigen, vgl. Abbildung 6. Um einen endlichen Raumausschnitt zu erhalten, setzen wir voraus, daß nur endliche Ausprägungen für die äußeren Merkmale der Objekte erlaubt sein dürfen. Der Analyserraum besitzt sein eigenes Koordinatensystem. Alle Objekte der Menge werden in dieses neue Koordinatensystem durch eine geeignete Skalierungstransformation transformiert

$$m'_i = f(m_i). \quad (5)$$

Die Realisierung des Analyserraumes als Ausschnitt des Merkmalraumes hat Vorteile, wobei bei der Wahl eines geeigneten Raumausschnittes verschiedene Randbedingungen (wie z. B. die Dynamik der Objektmenge) berücksichtigt werden müssen.

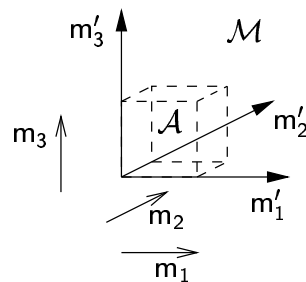


Abbildung 6: Analyserraum in einem dreidimensionalen Grundraum

Die Kenntnis der Größe eines Raumausschnittes – der Rauminhalt – ist notwendig für die Untersuchungen der Subräume. Unabhängig von der Dimension des Raumes wird der Rauminhalt hier als *Volumen* V bezeichnet [6]. Die Berechnung des Volumens von Teilräumen beliebiger Form ist schwierig zu realisieren, deshalb werden hier nur Volumina rechteckiger Raumausschnitte berücksichtigt, so daß bei einer Kantenlänge l das Volumen eines n -dimensionalen Raumausschnittes nach

$$V = \prod_{i=1}^n l_i \quad (6)$$

berechnet wird. Numerische Algorithmen verlangen meist nach einer Diskretisierung der Analysewelt. Das Ergebnis der Diskretisierung ist ein zelluläres Raum-Zeit-System, welches sich nun für Aufgaben der Klassifikation und Prozeßanalyse eignet. Die Struktur eines solchen zellulären Raumes wird in Abbildung 7 gezeigt. Der gesamte Analyse- und Raum ist hier in eine Menge orthogonaler Subräume aufgeteilt, im weiteren sollen diese Subräume als *Hyxel* bezeichnet werden [7]. Verschiedene Ebenen können Objekte mit verschiedenen Zustandsgrößen aufnehmen – beispielsweise pro Klasse eine Zustandsgröße.

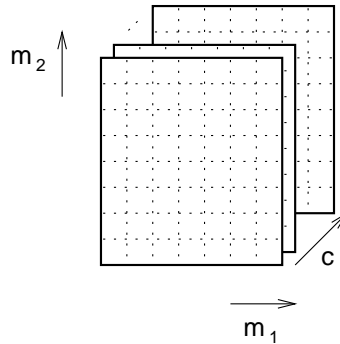


Abbildung 7: Zweidimensionaler zellulärer Raum für c Klassen

3 Ein erweitertes Objektmodell

3.1 Objekte nach dem klassischen Verständnis

Bevor mit der Auswertung von Prozeßdaten überhaupt begonnen werden kann, muß zunächst die Frage beantwortet werden: Was ist eigentlich ein Objekt? In [8] sind Objekte diskrete Projektionen von Prozessen oder Systemen, dies können die zu erkennenden Situationen oder Ereignisse der Umwelt sein [9]. Meßwerte, Sprachsequenzen oder Bildausschnitte können mit dem Terminus „Objekt“ bezeichnet werden. Im weiteren Sinne sind die Bemerkungen der zweiwertigen Aussagenlogik und der klassischen Mengenlehre gültig; Objekte repräsentieren auch einen begrifflichen Fakt und repräsentieren einen Ausschnitt unserer Anschauung und Sichtweise [10]. Allgemein wird ein Objekt als ein eingetretenes Ereignis interpretiert, wobei die Bedingungen für den Ereigniseintritt als Merkmale zur Beschreibung des Objektes dienen. Unter Berücksichtigung algorithmischer Gesichtspunkte ist ein Objekt ein Eintrag in einer Lernprobe bzw. ein Element einer Objektmenge. Ein Objekt, welches durch n Merkmale beschrieben wird, erscheint als n -dimensionaler Punkt im Merkmalraum, der nun das Ereignis repräsentiert. Eine Vielzahl von Auswertalgorithmen basiert auf statistischen Methoden und berechnet zum Beispiel Häufigkeiten oder statistische Momente mit Hilfe solcher Punkte.



Abbildung 8: Objekt als Schnittstelle zwischen dem realen Prozeß und den Methoden der Klassifikation und Prozeßanalyse

Ein Objekt hat die Aufgabe einer diskreten Systemabbildung, aber auch eine repräsentative Funktion im Merkmalraum. Als Teil des empirischen Prozeßwissens vermittelt es den numerischen Analysealgorithmen Informationen über den realen Prozeß. Während die Komponenten eines Klassifikationssystems lediglich dessen Struktur bestimmen, bekommt der Klassifikator über die Objekte einen Zugang zum realen Prozeß, vgl. Abbildung 8. Diese Objekte sind *scharf* – der Grund ist in der zugrundeliegenden zweiwertigen Aussagelogik zu suchen – und grenzen sich damit von dem sie umgebenden Merkmalraum klar ab. Mitunter sind jedoch scharfe Objekte nicht immer sinnvoll, zur Lösung derartiger Probleme eignen sich Fuzzymengen. Spezielle Klassifikatoren wie der Fuzzy-Klassifikator werden mit Hilfe dieses Konzeptes realisiert [8].

3.2 Objekte des Zwei-Welten-Modells

3.2.1 Allgemeines

Ein Objekt (ordinale Merkmale vorausgesetzt) wird als Punkt im n -dimensionalen Merkmalraum angenommen, falls ein Ereignis eintritt, dessen Bedingungen mit der entsprechenden Position im Merkmalraum korrespondieren. Tritt dieses Ereignis nicht ein, so wird kein Punkt im Merkmalraum verzeichnet. Dieser Darstellungsform sind jedoch Grenzen gesetzt, da aus der Angabe von Koordinaten allein noch nicht auf einen Ereigniseintritt geschlossen werden kann. Zusätzlich zu den Koordinaten ist noch die Information „dieser Punkt repräsentiert ein Ereignis“ notwendig, um ein diskretes Ereignis exakt zu beschreiben. Ein weiterer Mangel dieses Objektmodells ist die fehlende Möglichkeit der Bewertung des Ereignisses.²

3.2.2 Merkmale und Zustände eines Objektes

Das klassische Objektmodell läßt sich nun um die in Abbildung 3 vorgestellten Beschreibungsmöglichkeiten erweitern, die eine detailliertere Charakterisierung der äußeren Form, aber auch verschiedene Zustände zulassen.

Externe Merkmale. Wie schon bei Phänomenen in Abschnitt 2.2.2, lassen sich auch bei Objekten die externen Merkmale in direkte und indirekte externe Merkmale klassifizieren. Während externe Merkmale zum einen die Position im Merkmalraum festlegen, bestimmen sie auch die äußere Form des Objektes. So sind in einem zweidimensionalen Merkmalraum Objekte möglich, die Formen nach Abbildung 9 haben können. Nulldimensionale Objekte haben keine Ausdehnung und entsprechen Punkten im Merkmalraum, wie sie schon von statistischen Methoden her bekannt sind. Eindimensionale Objekte können im Raum gekrümmte Trajektorien sein. Zweidimensionale Objekte erscheinen in den meisten Fällen als Flächen und können Klassengebiete repräsentieren. In den meisten Fällen sind Abtastwerte nur zu determinierten, diskreten Zeiten verfügbar. Diese Objekte haben keine Ausdehnung und erscheinen als Punkte im Merkmalraum.

²Hierunter sollen Zustandsgrößen des Ereignisses verstanden werden, die keine Dimension des Merkmalraumes bilden.

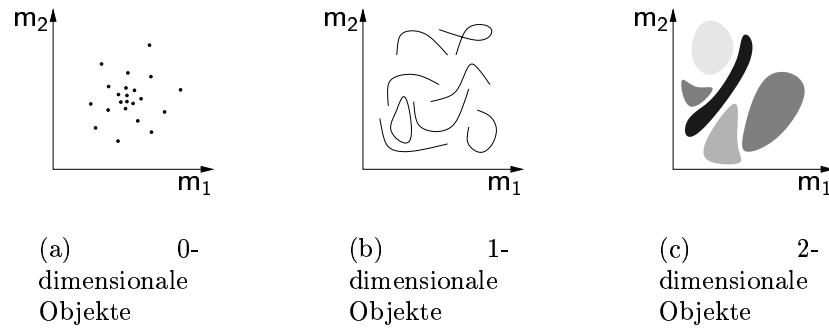


Abbildung 9: Verschiedene Objektarten im zweidimensionalen Merkmalraum

Interne Merkmale. Die wichtigste Ergänzung zum klassischen Objektmodell stellen interne Merkmale dar, die gleichbedeutend mit verschiedenen Zuständen sind. Ein Zustand e kann beispielsweise eine *Intensität* eines Phänomens widerspiegeln. Es ist günstig, für Aufgabenstellungen der Klassifikation für jede Klasse einen Zustand einzuführen, so daß nun auch die Anziehung bzw. Abstoßung eines Objektes zu den verschiedenen Klassen modelliert werden kann.

Die meisten Klassifikationstechniken – hier seien vor allem statistische und Abstandsklassifikatoren erwähnt – erlauben nur binäre Ausprägungen mit zwei Zuständen $e = [0, 1]$. Durch den Ein- bzw. den Austrag eines Objektes in bzw. aus einer Lernprobe werden diese beiden Zustände implizit definiert. Eine Erweiterung des Zustandswertebereiches auf $e = [0, \dots, 1]$ (allgemein: $e = [0, \dots, \infty]$) erlaubt die Modellierung von Fuzzymengen durch die Angabe von Zugehörigkeitswerten. Jedoch sind auch negative Zustände sinnvoll, um Phänomene auf bestimmten Positionen des Merkmalraumes sicher auszuschließen, so daß wir nun $e = [-1, \dots, 1]$ bzw. allgemein $e = [-\infty, \dots, \infty]$ schreiben können.

3.3 Mengen und Dipole

Das klassische Objektmodell eignet sich nicht für alle Anwendungsfälle, da die Bildung komplementärer Objekte, die sich auf den diskreten Ereignisbegriff stützen, zu nicht sinnvollen Ergebnissen führt.³ Hieraus ergibt sich die Motivation für die Bildung eines Konstruktes, das sich zur Beschreibung gegensätzlicher Sachverhalte eignet, die mit dem klassischen Mengenbegriff nur ungenügend wiedergegeben werden können. Im folgenden werden zwei Möglichkeiten zur Handhabung von Objekten vorgestellt.

3.3.1 Der Begriff der Menge

Eine Menge ist ein Zusammenfassung von bestimmten, unterscheidbaren Elementen eines Grundbereiches, die ein bestimmtes gemeinsames Merkmal besitzen und eine gemeinsame Aussage erfüllen. Ein Objekt x , das einer Menge \mathcal{X} zugewiesen werden

³Ein komplementäres Objekt $\neg o$ zu einem gegebenen Objekt o würde den gesamten Merkmalraum mit Ausnahme der Position des Objektes o ausfüllen.

kann, wird auch als Element $x \in \mathcal{X}$ der Menge \mathcal{X} bezeichnet. Für Mengen existiert eine spezielle Mengenalgebra. Weitere Informationen hierzu sind in [10] enthalten.

Mengen realisieren ein pluralistisches Prinzip. Verschiedene Elemente (Begriffe) werden zur Bildung eines Gattungsbegriffes herangezogen – hier die Menge selbst. Der notwendige Bezug zur Menge wird durch die gemeinsame Eigenschaft der Elemente hergestellt. Mengen sind jedoch nicht universell genug, um beliebige Objekte zusammenzufassen. Sie sind zur Beschreibung gegensätzlicher Sachverhalte ungeeignet.

3.3.2 Der Begriff des Dipols

Im Gegensatz zur Mengentheorie wird mit Hilfe des Dipols das Prinzip der dialektischen Einheit realisiert. Die dialektische Einheit besteht aus einer These und einer Antithese und schließt einen dialektischen Widerspruch ein [5]. Die These stellt eine Behauptung bzw. eine Aussage dar und wird mit der Antithese negiert. Der Widerspruch wird mit der Synthese aufgehoben. Beide Seiten des Widerspruchs werden hier

- negiert
- bewahrt, sowie
- auf eine höhere Ebene gehoben.

Die Pole des Dipols können Begriffe oder Variablen sein, welche einen gemeinsamen Bezug auf ein Merkmal mit gegensätzlichen Ausprägungen besitzen und damit einen dialektischen Widerspruch verursachen. Mit Hilfe des Dipols kann ein Dualismus aus zwei sich einander ausschließenden Begriffen in eine Einheit überführt werden. Mit Hilfe des Zeichens $\langle \blacktriangleright$ soll im folgenden dieser Widerspruch symbolisiert werden. Ein Dipol wird durch $\langle P_{\circ} \blacktriangleright P_{\bullet} \rangle$ dargestellt, wobei P_{\circ} der affirmative Pol und P_{\bullet} der negierte Pol darstellt.

In manchen Fällen gestattet die Eigenschaft die Superposition (z. B. ordinale Merkmale) und stellt damit einen zusätzlichen Bezugspunkt zur Verfügung. So bilden beispielsweise die reellen Zahlen 5 und -5 den Dipol $D = \langle 5 \blacktriangleright -5 \rangle$ mit dem Bezugspunkt 0. Beide Zahlen beziehen sich auf die Menge der reellen Zahlen und sind bezüglich 0 konträre Realisierungen dieser Menge. Diese Form der Negation, bei der ein Bezugspunkt erforderlich ist, wird im weiteren als *Kontraposition* bezeichnet. Bei ordinalen Merkmalen führt die Vereinigung beider Pole zu einer *Annihilation*, bei der beide Pole vernichtet werden und eine leere Menge entsteht

$$P_{\circ} \cup P_{\bullet} \rightarrow \emptyset + I. \tag{7}$$

Durch I wird die *Intension* dargestellt, die ein Objekt auf einer qualitativ anderen Ebene symbolisieren kann und dessen Negation der Bildung des ursprünglichen Dipols $D = \langle P_{\circ} \blacktriangleright P_{\bullet} \rangle$ entspricht. Eine Intension – hier als konkrete Ausprägung in Form des Dipols $D = \langle P_{\circ} \blacktriangleright P_{\bullet} \rangle$ – kann durch die Vereinigung nicht einfach verschwinden, sondern führt zu einer qualitativ anderen Intension.

Ist ein Bezugspunkt nicht vorhanden, ist nur die *Kontradiktion* als Negation zulässig. Bei der Kontradiktion entsteht ein negierter (komplementärer) Begriff $\neg A$ allein durch die Verneinung des Begriffes A . So bilden beispielsweise eine Kugel und eine äquivalente Hohlkugel (die in einem Körper eingebettet sein muß) einen Dipol, dessen Vereinigung zu einer neuen geometrischen Form – nämlich der Form des Körpers – führt. Diese Operation entspricht jedoch der Bildung eines Gesamtbegriffes (und ist damit gleichbedeutend mit der Bildung einer Gesamtheit). Durch den Begriff des Dipols ist nun möglich, nicht nur komplementäre Mengen, sondern auch *inverse* Mengen zu erzeugen und zu handhaben.

3.4 Generalisierung in der Analysewelt

3.4.1 Objektdichte und Wahrscheinlichkeit

Der Begriff des Zustands läßt sich auf jeden Ausschnitt des Raumes anwenden. Setzen wir eine Objektmenge von N Objekten im Analyse Raum \mathcal{A} voraus, dann kann die absolute Objektdichte des klassischen Objektmodells, bei dem nur die Zustände $e = [0, 1]$ und hier speziell $e_i = 1$ erlaubt sind, in einem Hyxel \mathcal{R} mit dem Volumen $V_{\mathcal{R}}$ mittels

$$E = \frac{\sum_{i=1}^{N_{\mathcal{R}}} e_i}{V_{\mathcal{R}}} \quad (8)$$

berechnet werden. Hieraus läßt sich die relative Objektdichte des Hyxels \mathcal{R} über

$$r_{\mathcal{R}} = \frac{\sum_{i=1}^{N_{\mathcal{R}}} e_i}{V_{\mathcal{R}} \cdot N} \quad (9)$$

bestimmen. Ausgehend von klassischem Objektmodell beträgt die relative Objekthäufigkeit

$$h_{\mathcal{R}} = \frac{\sum_{i=1}^{N_{\mathcal{R}}} e_i}{N}. \quad (10)$$

Für eine unendlich große Objektmenge $N \rightarrow \infty$ und ein gegen Null strebendes Hyxelvolumen $V_{\mathcal{R}} \rightarrow 0$ konvergiert die relative Häufigkeit an der durch das Hyxel \mathcal{R} repräsentierten Position $\mathbf{m}_{\mathcal{R}}$ gegen die Wahrscheinlichkeit P

$$P(\mathbf{m}_{\mathcal{R}}) = \lim_{N \rightarrow \infty} \left. \frac{\sum_{i=1}^{N_{\mathcal{R}}} e_i}{N} \right|_{V_{\mathcal{R}} \rightarrow 0}. \quad (11)$$

Die Wahrscheinlichkeit P ist gleichbedeutend mit einer Generalisierung der Objektverhältnisse in der betrachteten Welt und stützt sich auf die zweiwertige Aussagenlogik [11]. Eine Wahrscheinlichkeit $P(\mathbf{m}_{\mathcal{R}}) = 0$ drückt die Unmöglichkeit aus, daß ein sich ein Objekt im betrachteten Hyxel \mathcal{R} befindet.

3.4.2 Bestimmtheit und Gewißheit

Im Zwei-Welten-Modell nach Abschnitt 2 werden jedoch die durch die Sensoren nicht erfaßten Prozeßsituationen – die Noumene – in der Analysewelt W_A nicht berücksichtigt. Ein leerer Analyse Raum \mathcal{A} stellt eine *tabula rasa* dar, die keine Rückschlüsse

auf die Prozeßsituationen der Prozeßwelt W_P zuläßt. An jeder Position des Analyseraumes sind korrespondierende Objekte in W_P *möglich*, sichere Aussagen hierzu können aber nicht getroffen werden, da Informationen über die Prozeßwelt nicht verfügbar sind [5]. Die Wahrscheinlichkeit P ist jedoch in diesem Fall an jeder Stelle des Merkmalraumes gleich null und schließt Objekte in W_A damit aus. Die durch die Wahrscheinlichkeit charakterisierten Objektverhältnisse in W_A sind damit nicht auf die Prozeßwelt W_P übertragbar, da nicht von den Objektverhältnissen in W_A auf die Verhältnisse einer externen Welt W_P geschlossen werden kann. Die Wahrscheinlichkeit ist somit nur als Generalisierung in der eigenen Welt sinnvoll. Sichere Aussagen über die Prozeßwelt W_P wären ausgehend von der Analysewelt W_A möglich, wenn der gesamte Analyseraum \mathcal{A} mit Methoden der experimentellen Prozeßanalyse derart abgetastet würde, daß alle korrespondierenden Gebiete der Prozeßwelt erfaßt würden. Jedoch ist in den meisten Fällen eine aktive Prozeßbeobachtung nicht möglich, so daß der Prozeß nur passiv und unvollständig mit Hilfe einer Lernprobe beurteilt werden kann.

Um von den bekannten Objektverhältnissen in W_A auf die unbekanntenen Objekt- und Prozeßverhältnisse in W_P schließen zu können, ist ein neuer Begriff notwendig, mit dem die Notwendigkeit, die Möglichkeit und die Unmöglichkeit der Existenz eines Phänomens in W_P an der korrespondierenden Position in W_A einschätzbar ist. Dieser Begriff soll als *Bestimmtheit* D bezeichnet werden und wird analog zu Gleichung 11 über

$$D(\mathbf{m}_{\mathcal{R}}) = \lim_{N \rightarrow \infty} \frac{\sum_{i=1}^{N_{\mathcal{R}}} e_i}{N} \Bigg|_{V_{\mathcal{R}} \rightarrow 0} \quad (12)$$

berechnet, wobei nun die Beschränkung auf nur einen Zustand entfällt und die Zustände

$$e = \pm 1 \quad (13)$$

$$= \langle 1 \blacktriangleleft -1 \rangle \quad (14)$$

zugelassen sind, wobei die 1 mit einer bestimmten Ausprägung eines inneren (ordinalen) Merkmals verknüpft ist und somit aus der Ausprägung des (inneren) Merkmals ein neues Merkmal bezüglich der Bestimmtheit der Ausprägung wird. Die Bestimmtheit D kann für die Beurteilung der verallgemeinerten Verhältnisse in der Welt W_P ausgehend von den Verhältnissen in der Welt W_A benutzt werden. Sie hat die folgenden Eckwerte:

1. Um die Notwendigkeit der Existenz eines Phänomens in W_P auszudrücken, welches mit der Position $\mathbf{m}_{\mathcal{R}}$ des Analyseraumes \mathcal{A} korrespondiert, sei $D = 1$.
2. Falls sicher ausgeschlossen werden kann, daß ein Phänomen in W_P mit der Position $\mathbf{m}_{\mathcal{R}}$ des Analyseraumes in W_A korrespondiert, sei $D = -1$.
3. Ist keine sichere Aussage über die Existenz eines Phänomens möglich, d. h. ist es *möglich*, daß ein mit der Analysewelt korrespondierendes Phänomen in W_P existiert, soll $D = 0$ gewählt werden.

Allgemein ist der Wertebereich $D = [-1, \dots, 1]$ erlaubt. Mit Hilfe der Bestimmtheit ist eine Aussage über die Fundiertheit des Prozeßwissens ableitbar, aus der die Kenngrößen

$$D = \langle C_{\circ} \blacktriangleleft C_{\bullet} \rangle \quad (15)$$

ableitbar sind, wobei sich die Gewißheit C_{\circ} nach

$$C_{\circ} = |D| \quad (16)$$

und die Ungewißheit C_{\bullet} über

$$C_{\bullet} = 1 - |D| \quad (17)$$

bestimmen läßt. Damit ergibt sich für die Bestimmtheit eines Phänomens die folgende logische Form

$$D = \langle \langle \text{Notwendigkeit} \blacktriangleleft \text{Unmöglichkeit} \rangle \blacktriangleleft \text{Ungewißheit (Kontingenz)} \rangle. \quad (18)$$

Im Gegensatz zur Wahrscheinlichkeit, die den Zufall als objektive Größe ansieht, unterstützt die Gewißheit das Konzept des Nicht-Wissens bzw. der Nicht-Kennntnis determinierter Dinge, die sich außerhalb der Welt W_A (und in der Welt W_P) befinden.

4 Anwendungen des Zwei-Welten-Modells

4.1 Zwei-Welten-Modell an einem Rotationsverdampfer

Ein Beispiel für einen solchen realen Prozeß ist ein Rotationsverdampfer nach Abbildung 10, mit dem sich aus einem Stoffgemisch einzelne Stoffe wieder abtrennen lassen. Der Destillationsprozeß findet aus Energiespargründen unter Vakuum statt. Zu diesem Zweck wird an den Rotationsverdampfer eine Vakuumpumpe angeschlossen, mit der eine Evakuierung des Rotationsverdampfers eingeleitet wird. Über ein Stellventil läßt sich der Druckabfall im Rotationsverdampfer steuern. Das Stoffgemisch befindet sich in der Destillierblase, in der bei geeigneter Temperatur des Heizbades und einem entsprechenden Druckabfall der Siedeprozess beginnt. Der Dampf des Destillats strömt anschließend in den Kühler, kondensiert dort aus und sammelt sich in der Destillatvorlage. Eine detailliertere Beschreibung dieses Prozesses ist in [12] zu finden. Der Zugang zu diesem Prozeß erfolgt nun über Temperatursensoren, die die Aufgabe besitzen, die Dampftemperatur an verschiedenen Stellen des Kondensators zu messen. Ein Drucksensor gibt Auskunft über die Druckverhältnisse im Rotationsverdampfer.

Daß in einer Analysewelt nur unvollständige Informationen über den Prozeß vorhanden sein können, wird am Beispiel der Detektion des Siedebeginns deutlich. Für einen Beobachter ist der Siedebeginn klar zu erkennen. Am Stoffgemisch ist der Siedebeginn durch einsetzende Blasenbildung erkennbar, in vielen Fällen beginnt es zu schäumen. Manchmal bildet sich über dem zu destillierenden Stoffgemisch Dampf, welcher dann in den Kühler strömt. In den Analyseraum lassen sich jedoch nur die Dampftemperaturen an den durch die Sensoren bestimmten Stellen im Kühler und der Systemdruck abbilden. Aus diesen unvollständigen Informationen muß über eine Signalauswertung der Siedebeginn ebenfalls sicher detektiert werden. Anschließend ist der Destilliervorgang durchzuführen, das Destillierende zu erkennen und der gesamte Vorgang zu beenden. Diese komplexe Aufgabe wird durch eine fuzzybasierte

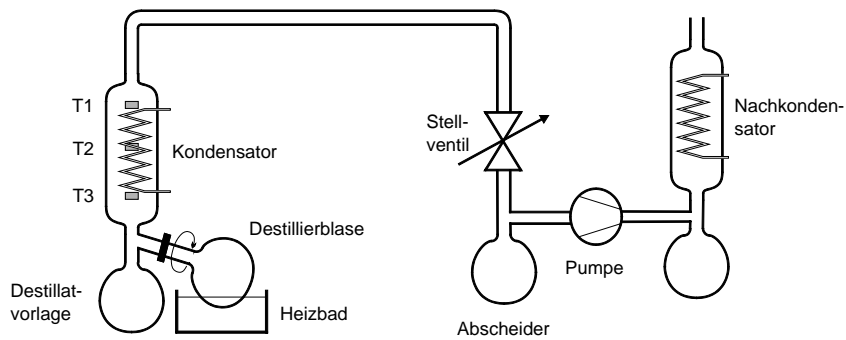


Abbildung 10: Aufbau eines Rotationsverdampfers

Klassensteuerung erreicht, die – innerhalb der Analysewelt W_A – die projizierten Prozeßdaten auswertet und den Destillationsvorgang überwacht. Der Druckverlauf eines geregelten Destillationsvorganges ist in Abbildung 11 dargestellt.

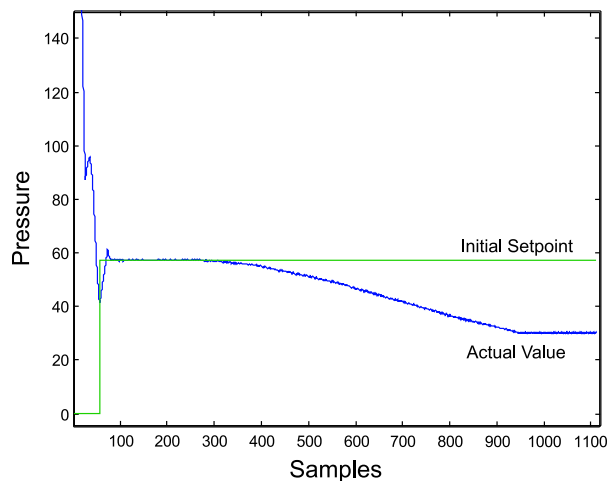


Abbildung 11: Druckverlauf im Rotationsverdampfer während der geregelten Destillation

4.2 Suche nach Attraktoren

Oft bilden die Objekte einer Lernprobe *Wolken* im Merkmalraum, für die sich ein Schwerpunkt ermitteln läßt. Diese Objektwolken repräsentieren das System- oder Prozeßverhalten um einen stabilen Arbeitspunkt. Dieser Arbeitspunkt läßt sich als *Attraktor* interpretieren, der die Ursache eines stabiles Systemverhaltens ist. Die Stabilität des Systemverhaltens kann durch die Anziehungskraft (die Affinität) des Attraktors auf den Prozeß- bzw. Systemzustand erklärt und modelliert werden. Ein Beobachter bekommt den Eindruck, als würden die Objekte der Stichprobe in W_A durch einen unbekanntem Attraktor zusammengehalten [6]. Ein Ziel soll es sein, derartige Attraktoren ausfindig zu machen und aus dem vorhandenen empirischen Prozeßwissen zu bestimmen. Dies wird mit Hilfe eines Feldmodells erreicht, das sich auf die Berechnung der Gravitation stützt.

4.2.1 Feldmodell

In [7] wurde ein Feldmodell zur Lösung von Klassifikationsaufgaben und Problemstellungen der Prozeßanalyse vorgestellt. Dieses Feldmodell stützt sich auf die Feldgleichungen aus [13, 14] und basiert auf einem Objektmodell nach Abschnitt 3.2. Die inneren Zustände e_i der Objekte lassen sich in diesem Modell als eine Art *Beladung* interpretieren. Damit ist es möglich, zwischen zwei Objekten mit den Beladungen e_1 und e_2 die Affinität \vec{F} nach

$$\vec{F} = \frac{e_1 \cdot e_2}{(d(\mathbf{m}_1, \mathbf{m}_2))^2} \cdot \vec{d}(\mathbf{m}_1, \mathbf{m}_2) \quad (19)$$

zu berechnen, wobei d den euklidischen Abstand zwischen beiden Objekten darstellt. Um ein Objekt e_1 läßt sich ein Affinitätsfeld mit der Feldstärke

$$\vec{A}(\mathbf{m}_0) = \frac{e_1}{(d(\mathbf{m}_0, \mathbf{m}_1))^3} \cdot \vec{d}(\mathbf{m}_0, \mathbf{m}_1) \quad (20)$$

vermuten. In einem Zweiraumsystem zur Vermeidung von Singularitäten bei der Feldberechnung nach Abbildung 12, bei dem der Objektraum \mathcal{O} die Beladungen aufnimmt und die Feldberechnung im Analyseraum \mathcal{A} stattfindet, kann die Gravitation im Analyseraum \mathcal{A} nach

$$\vec{G}(\mathbf{m}_0) \triangleleft \frac{e_1}{(d_\zeta(\mathbf{m}_0, \mathbf{m}_1))^3} \cdot \vec{d}(\mathbf{m}_0, \mathbf{m}_1) \quad (21)$$

$$= -\frac{e_1}{(d_\zeta(\mathbf{m}_0, \mathbf{m}_1))^3} \cdot \vec{d}(\mathbf{m}_0, \mathbf{m}_1) \quad (22)$$

berechnet werden, wobei der euklidische Abstand d_ζ diesmal auch von der Verschiebung der beiden Räume ζ bestimmt wird

$$d_\zeta = \sqrt{\sum_{i=1}^n (m_{i,0} - m_{i,1})^2 + \zeta^2}. \quad (23)$$

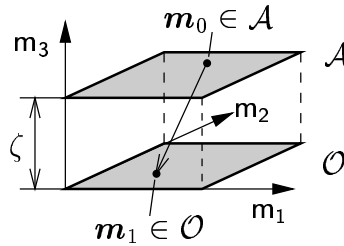


Abbildung 12: Raumsystem zur Feldberechnung

4.2.2 Simulation

Das Feldmodell wurde an einem Zwei-Klassen-Datensatz bestehend aus 200 Objekten getestet. Dieser Datensatz repräsentiert ein simuliertes Systemverhalten um zwei stabile Schwerpunkte, die in der Analysewelt W_A nicht bekannt und damit Noumene

sind, bezüglich des Systemverhaltens Attraktoren darstellen und aus dem verfügbaren empirischen Prozeßwissen rekonstruiert werden sollen. Abbildung 13 zeigt das Zweiraumsystem mit Lernstichprobe und berechnetem Gravitationsverlauf. Im Gravitationsfeld werden die Attraktoren als Feldsenken sichtbar. Beide Klassengebiete werden durch einen Sattelpunkt voneinander abgetrennt. Dieser Sattelpunkt wird nicht durch einen Attraktor verursacht, er stellt vielmehr einen Grenzpunkt dar, der auf der Trennkurve zwischen den Einzugsgebieten der beiden Klassen liegt.

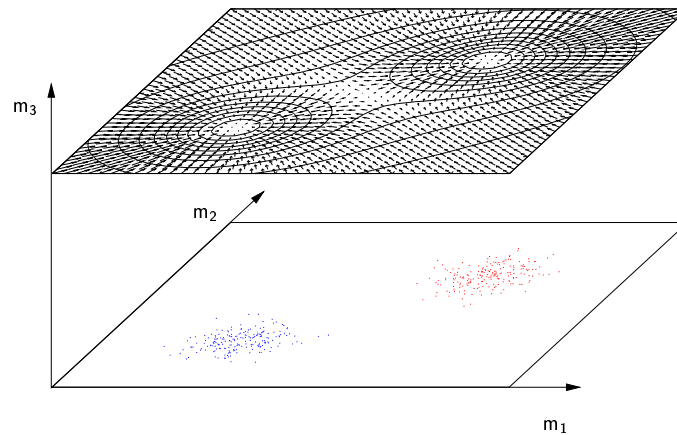


Abbildung 13: Zweiraumsystem für ein simuliertes Zwei-Klassen-Problem

5 Zusammenfassung

In diesem Beitrag wurde ein Zwei-Welten-Modell zur Lösung von Klassifikations- und Prozeßanalyseproblemen beschrieben. Es orientiert sich an der Weltanschauung des repräsentativen Realismus und geht von einer Welt der Realität aus, die vom menschlichen Bewußtsein und den Analysemethoden unabhängig ist. Dinge der Realität können nicht vollständig beschrieben werden, dies ist nur im Rahmen der zur Verfügung stehenden, begrenzten und nicht immer repräsentativen Informationen möglich. Die Informationen dienen der Projektion der realen Dinge in eine zweite Analysewelt, in welcher dann die weitere Auswertung vorgenommen werden kann. Dieses Zwei-Welten-Modell dient als Grundlage für weitere Analysemethoden.

Literatur

- [1] <http://www.phillex.de>.
- [2] *Lexikon Medizin*. Naumann & Göbel Verlagsgesellschaft, Köln.
- [3] KANT, I.: *Die drei Kritiken in ihrem Zusammenhang mit dem Gesamtwerk*. Alfred Kröner Verlag, Stuttgart, 1975.
- [4] HOFFMAN, D.: *Visuelle Intelligenz: Wie die Welt im Kopf entsteht*. Klett-Cotta, Stuttgart, 2000.
- [5] HOFFMEISTER, J.: *Wörterbuch der philosophischen Begriffe*. Verlag Felix Meiner, Hamburg, 1955.

- [6] KUHN, CHR.: *Merkmalsgenerierung und Klassifikation*. In: MIKUT, R. und J. JÄKEL (Herausgeber): *Proceedings 10 Workshop Fuzzy Control des GMA-FA 5.22*, Dortmund, 2000.
- [7] KUHN, CHR.: *Feature Extraction with Cellular Spaces*. In: *Intech' 2001*, Assumption University, Bangkok, Thailand, 2001.
- [8] BOCKLISCH, S. F.: *Prozeßanalyse mit unscharfen Verfahren*. Verlag Technik, Berlin, 1987.
- [9] STEINHAGEN, H.-E. und S. FUCHS: *Objekterkennung*. Verlag Technik, Berlin, 1976.
- [10] BRONSTEIN, I. N. und K. A. SEMENDJAJEW: *Taschenbuch der Mathematik*. BSB B. G. Teubner Verlagsgesellschaft and Nauka Verlag, Leipzig and Moskau, 1979.
- [11] GNEDENKO, B. W.: *Lehrbuch der Wahrscheinlichkeitstheorie*. Verlag Harri Deutsch, Thun, Frankfurt a.M., 1997.
- [12] KUHN, CHR.: *Automated Measurement and Control on a Rotary Evaporator*. In: *EUFIT '99: 7th European Congress on Intelligent Techniques & Soft Computing*, Aachen, Germany, 1999. Verlag Mainz.
- [13] STROPPE, H.: *Physik für Studenten der Natur- und Technikwissenschaften*. Fachbuchverlag, Leipzig, 1988.
- [14] STRASSACKER, G.: *Rotation, Divergenz und das Drumherum*. 1992.

Modellbildung und Fuzzy-Gelenkpositionsregelung für eine 5-Finger-Roboterhand mit flexiblen Fluidaktoren

Sebastian Beck, Arne Lehmann, Jan Martin, Ralf Mikut

Forschungszentrum Karlsruhe GmbH, Institut für Angewandte Informatik,
D-76021 Karlsruhe, Postfach 3640,
Telefon: (07247) 82-5749, Fax: (07247) 82-5786, E-Mail: sebastian.beck@iai.fzk.de

Abstract

Für eine anthropomorphe 5-Finger-Roboterhand, die mittels flexibler Fluidaktoren angetrieben wird und bereits in einer gesteuerten Variante in der Prothetik Anwendung findet, wird in dieser Arbeit der Übergang von der Steuerung der einzelnen Fingergelenke zur Positionsregelung dargestellt. Nach einer kurzen Beschreibung des mechatronischen Systems wird zunächst die Identifikation der pneumatisch betriebenen Strecke am Beispiel eines einzelnen Fingergelenkes erläutert. Das viskoelastische Verhalten des Aktormaterials wird in das dynamische Modell der Strecke integriert. Danach wird der Entwurf des Fuzzy-Reglers für das einzelne Gelenk, dessen Struktur auf den Ergebnissen der Streckenmodellierung basiert, beschrieben.

1 Einleitung

Roboter sollen zunehmend komplexere Aufgaben in dynamischen Umgebungen erfüllen. Manipulation von Objekten unterschiedlicher Form, Größe und Gewicht, sowie Arbeiten mit vielerlei Werkzeugen gehören dabei zu den Aufgaben der zukünftigen Roboter. Die Interaktion mit dem Menschen in der selben Umgebung stellt hohe Anforderungen an die Sicherheit der Aktionen des Roboters. Die Ausstattung des Roboters mit einem der menschlichen Hand in Erscheinung und Funktionalität nahe kommenden Greifer, der die Griffmuster des Menschen beherrscht, ist ein wichtiger Schritt in diese Richtung. Die Maschinen werden in der Lage sein, mit den selben Werkzeugen umzugehen, die der Mensch benutzt.

Im Gegensatz zu herkömmlichen Roboterhänden [1, 2, 3] werden die Freiheitsgrade der 5-Finger FZK-Hand [4, 5] nicht über Elektromotoren, sondern durch flexible Fluidaktoren [6] angetrieben. Die Aktoren der Hand können optional hydraulisch oder pneumatisch betrieben werden. Die Wahl der Fluidversorgung hängt von den Einsatzbedingungen der Hand ab. Die flexiblen Fluidaktoren ermöglichen neben den harmonischen Bewegungen der Fingergelenke eine Reduktion des Gewichts und machen den Einsatz der hydraulisch betriebenen Hand als Prothese möglich [7, 8] (Bild 1).

Für den Einsatz der Prothesenhand als Greifer in der Robotik ist die Regelung der Freiheitsgrade der Hand notwendig. Für Kräfte an den Fingern oder Positionen der einzelnen Gelenke als Führungsgrößen der Hand-Regelung sind Kraft- beziehungsweise Winkelsensoren in die vorhandene Hand zu integrieren. Eine Reglerstruktur für die Kraft- beziehungsweise Positionsregelung ist zu entwerfen und zu testen. Das

nichtlineare zeitvariante Verhalten der Aktoren erfordert den Einsatz von Methoden, die über die Ansätze der linearen Regelungstechnik hinausgehen. Bobrow [9] beschreibt die nichtlineare Regelung eines Roboterarmes mit starren pneumatischen Aktoren. Neumann [10] stellt ebenfalls Anwendungsmöglichkeiten von pneumatischen Aktoren in der Robotik vor. Van der Smagt [11] untersucht die Positionsregelung eines mittels sogenannter 'Rubbertuators' betriebenen Roboterarmes. Auch diese Aktoren haben eine nichtlineare statische Kennlinie und zusätzlich eine bauartbedingte Hysterese. Van der Smagt ersetzt wegen mangelhafter Positioniergenauigkeit den üblichen PID-Regler durch ein neuronales Netz. Ansätze zur Modellbildung flexibler Fluidaktoren sind in der Arbeit von Loose [12] zu finden.

Das in dieser Arbeit vorgestellte Regelungskonzept für die Fingergelenkregelung der mit flexiblen Fluidaktoren angetriebenen FZK-Roboterhand basiert auf einem Fuzzy-adaptiven PID-Regler. Methoden der Fuzzy-Logic sind in der Regelung von Robotern verbreitet, hier sei exemplarisch auf die Arbeiten von Kuntze [13] verwiesen. Die Grundlage für den Reglerentwurf ist die Modellbildung der zu regelnden Strecke. Nach dem strukturellen Ansatz aus einer theoretischen Modellbildung erfolgt die Identifikation der Parameter, die das Verhalten der realen Strecke am besten abbilden. Die Erkenntnisse werden auf den Fingergelenkregler übertragen. Im Einzelnen wird auf

- den mechatronische Aufbau der Roboterhand sowie die Modellbildung eines pneumatisch betriebenen Fingergelenkes der Roboterhand (Abschnitt 2),
- die Struktur Fuzzy-adaptiver PID-Regler (Abschnitt 3) sowie
- die Implementierung eines Fuzzy-adaptiven PID-Reglers für ein pneumatisch betriebenes Fingergelenk auf Basis der Ergebnisse der Modellbildung und den Vergleich mit einem PID-Regler mit vollständiger Kompensation der nichtlinearen Kennlinien (Abschnitt 4).

eingegangen.



Bild 1: Die FZK-Handprothese

2 Modellbildung

Die fluidisch betriebene Roboterhand lässt sich sowohl als pneumatisches als auch als hydraulisches System realisieren. Sie unterscheiden sich lediglich in der Fluidversorgung und deren Anordnung in der Hand.

Unabhängig von der Fluidversorgung kann die Anzahl der Freiheitsgrade der Hand gewählt werden. Zur Bewegung eines Fingergelenkes ist jeweils ein Fluidaktor notwendig. Wird die flexible Kammer des Aktors befüllt, bewegt sich das Gelenk aufgrund der Expansion. Wird die Kammer entleert kehrt das Gelenk durch die Rückstellkräfte des gedehnten Aktormaterials in seine Ausgangsposition zurück. Eine mögliche Konfiguration ist folgende: Jeder Finger wird mit drei Fluidaktoren ausgestattet. Der Daumen verfügt über einen weiteren Fluidaktor zum Erreichen der Oppositionsstellung, so dass insgesamt 16 Fluidaktoren in die Gelenke der Hand integriert sind.

Mit der Fluidmenge im Aktor erhöht sich der Innendruck p_{Aktor} und damit das vom Aktor generierte Drehmoment M_{Aktor} . Die Dehnung des Aktormaterials nimmt mit dem Winkel φ zu und verringert das Drehmoment des Aktors. Messungen [14] mit Druckluft ergeben das in Bild 5 (links) dargestellte Kennfeld des Aktormomentes $M_{Aktor}(\varphi, p_{Aktor})$. Der Gelenkwinkel in der statische Ruhelage ($M_{Aktor}(\varphi, p_{Aktor}) = 0$) des Gelenkes steigt mit zunehmendem Druck.

Für die Druckluftversorgung wird in der hier untersuchten Konfiguration ein externer Kompressor verwendet. Jeder Freiheitsgrad ist über ein Ventil, das den Druck im Aktor regelt, mit dem Kompressor verbunden. Die Pneumatik-Ventile sind wie der Kompressor aufgrund ihrer Ausmaße nicht in der Hand integriert. Die Stellgröße des einzelnen Fingergelenkreglers ist der Solldruck im Aktor. Die Komponenten der pneumatischen Regelung sind in Bild 2 (links) dargestellt. Der Einsatz von Druckluft ist vorteilhaft, wenn eine Kompressoranlage, beispielsweise in Bereich der Fertigung, bereits vorhanden ist.

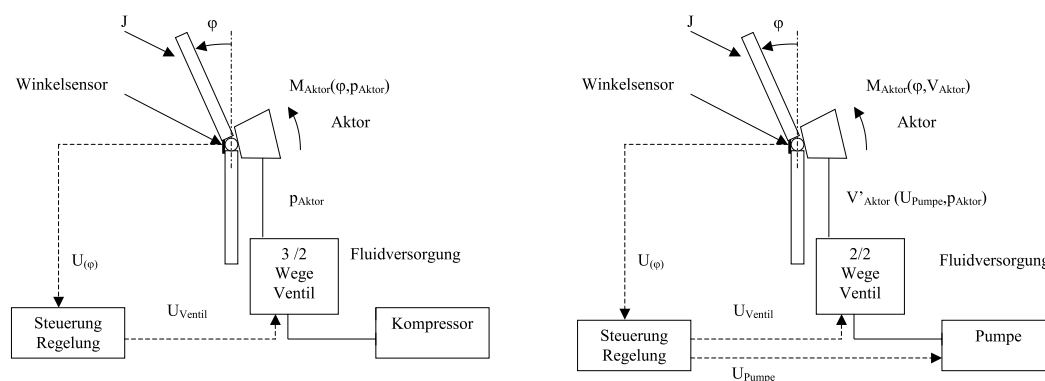


Bild 2: Schematische Darstellung der pneumatischen (links) und hydraulischen (rechts) Gelenkregelung

Weniger Platz benötigt das System mit hydraulischer Fluidversorgung, wie es bereits in der gesteuerten FZK-Handprothese eingesetzt wird (Bild 1). Die Hydraulik-Ventile und die Miniaturpumpe zur Erzeugung des Fluiddruckes sind dabei in den Handteller integriert (Bild einfügen). Der Regler des hydraulischen Systems hat im

Vergleich zum pneumatischen System eine andere Struktur, da der Druck in den Aktoren vom Regler nicht direkt als Stellgröße an die Ventile ausgegeben werden kann. Stellgröße des Reglers ist die Spannung an der Pumpe und somit deren Durchfluss. Der Durchfluss der Pumpe ist neben der Spannung auch vom Gegendruck der Flüssigkeit in den Aktoren abhängig. Bild 2 (rechts) zeigt die Komponenten der hydraulischen Regelung. Wegen der Inkompressibilität des Fluids ist das hydraulische System bei geschlossenen Ventilen steifer als das pneumatische.

Um den Regelkreis zu schließen, ist jeder Freiheitsgrad mit einem Positionssensor ausgestattet. Der Sensor ist in das Gelenk integriert und liefert eine Spannung U in Abhängigkeit vom Gelenkwinkel φ .

Im Weiteren wird ausschließlich die pneumatische Strecke betrachtet. Gemäß Bild 2 (links) besteht diese aus:

- dem Kompressor,
- dem 3/2 Wegeventil mit unterlagerter Druckregelung,
- dem Aktor,
- der Massenträgheit J des bewegten Schenkels des Gelenkes sowie
- dem Positionssensor zur Bestimmung des aktuellen Gelenkwinkels φ .

Das dynamische Verhalten der Strecke wird anhand von Druckverläufen im Aktor untersucht. Bild 3 zeigt die resultierenden Winkelverläufe für verschiedene Druckverläufe.

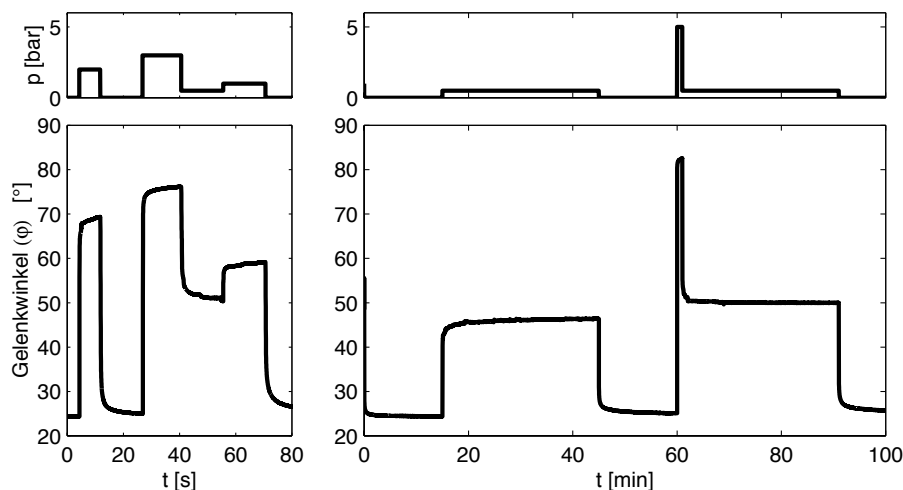


Bild 3: Dynamisches Verhalten des Aktors: Kurzzeitverhalten (links) Langzeitverhalten (rechts)

Die Modellstruktur, die auf einer theoretischer Modellbildung und auf heuristischen Ansätzen (Auswertung von Experimenten) basiert, ist in Bild 4 dargestellt.

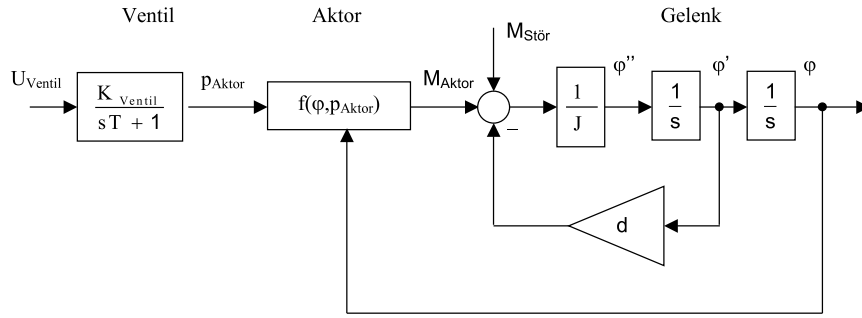


Bild 4: Modellstruktur für das Fingergelenk

Der Parameter d beschreibt dabei den zur Winkelgeschwindigkeit $\dot{\varphi}$ proportionalen Einfluss der Reibung im Gelenk.

Der Kompressor wird als Druckspeicher mit konstantem Druck $p_{Kompr.} > p_{Aktor,max}$ betrachtet und wird daher im Streckenmodell nicht berücksichtigt. Das Übertragungsverhalten des Ventils wird durch ein Verzögerungsglied 1. Ordnung (VZ₁-Glied) abgebildet. K_{Ventil} beträgt 0,6, da die Spannung $U_{Ventil} \in [0 \dots 10]$ V am Ventil proportional zum Ausgangsdruck $p_{Aktor} \in [0 \dots 6]$ bar ist.

Das statische Kennfeld $M_{Aktor}(\varphi, p_{Aktor})$ der Aktoren wird mit einem Teststand [14] aufgezeichnet. Über die schrittmotorgesteuerte Positionierung des Gelenkes werden beliebige Gelenkwinkel absolut angefahren. Das Drehmoment des Aktors in Abhängigkeit vom Innendruck wird aus den Signalen eines DMS-Sensors ermittelt. Das gemessene Kennfeld eines Aktors ist in Bild 5 (links) dargestellt.

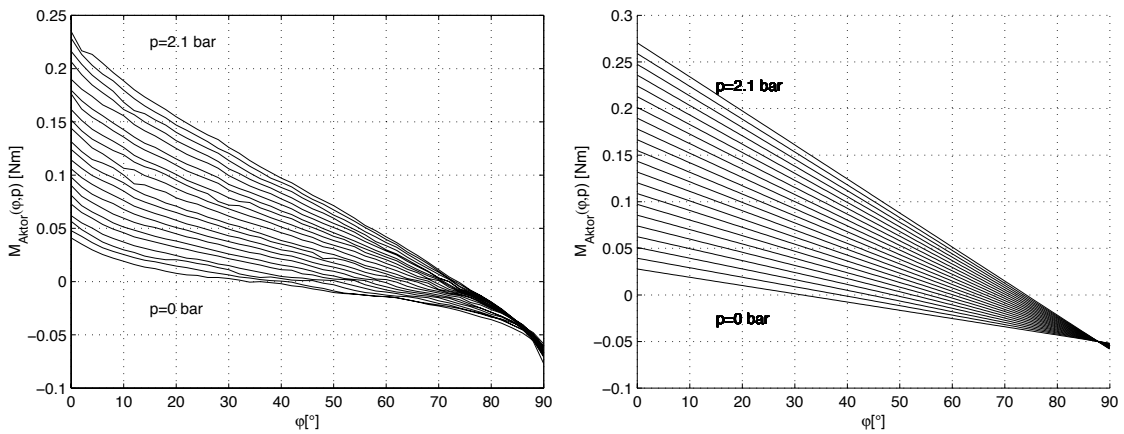


Bild 5: Ausschnitt aus dem gemessenen (links) und multilinear approximierten (rechts) statischen Aktorkennfeld

Für die Modellierung der Strecke wird das Kennfeld multilinear durch Gleichung (1) approximiert (Bild 5 (rechts)):

$$M_{Aktor}(p_{Aktor}, \varphi) = a_0 + a_1\varphi + a_2p_{Aktor} + a_3p_{Aktor}\varphi. \quad (1)$$

Die Winkelbeschleunigung $\ddot{\varphi}$ des Gelenkes ergibt sich aus dem resultierenden Moment $M_{Aktor} - M_{Stoer}$ am Gelenk und dem Trägheitsmoment J des bewegten Teils:

$$J\ddot{\varphi} + d\dot{\varphi} = M_{Aktor}(p_{Aktor}, \varphi) + M_{Stoer}. \quad (2)$$

Dabei ist M_{Stoer} ein beliebiges Moment, das von außen auf das Gelenk wirkt. $D\dot{\varphi}$ ist das Reibmoment, das der Bewegung entgegenwirkt. Die Winkelgeschwindigkeit $\dot{\varphi}$ ergibt sich durch Integration der Winkelbeschleunigung. Simulationsergebnisse mit diesem Modell sind in Bild 7 (links) dargestellt.

Bild 3 (rechts) verdeutlicht die Hysterese und Kriechvorgänge des Aktors. Die Strecke ist zunächst entlastet und wird anschließend mit 0,5 bar Druck beaufschlagt. Nach einer Erholungsphase wird kurz ein Druck von 5 bar angelegt und anschließend ohne Erholung erneut ein Druck von 0,5 bar. Die Strecke kommt beim identischen Druck von 0,5 bar in Abhängigkeit von der Druckvorgeschichte zu unterschiedlichen statischen Endlagen.

Dieses Verhalten ist mit dem Modell in (1) nicht erfassbar, aber durch das viskoelastische Verhalten des Aktormaterials zu erklären. Polymere zeigen im Gegensatz zu Metallen kein linear elastisches Verhalten $\sigma = E \cdot \varepsilon$. Damit nimmt bei konstanter Spannung σ die Dehnung $\varepsilon(t)$ asymptotisch zu (Dehnungsretardation), beziehungsweise bei konstanter Dehnung die Spannung $\sigma(t)$ asymptotisch ab (Spannungsrelaxation) [15, 16, 17].

In der Literatur wird als einfacher Ansatz für linear-viskoelastisches Verhalten ein Modell aus Hookeschen Federn und Newtonschen Dämpfern verwendet, um die Retardations- beziehungsweise die Relaxationzeit des Materials bei konstanter Spannung beziehungsweise konstanter Dehnung zu berücksichtigen [18, 19]. Das einfachste Modell dabei ist die Serienschaltung einer Feder mit der Parallelschaltung aus einer weiteren Feder und einem Dämpfer. Dieses Modell gibt die Sprungfähigkeit der Dehnung und das anschließende Kriechen des Materials qualitativ wieder. Für eine exakte Modellierung des Materials ist allerdings ein Modell mit unendlich vielen Federn und Dämpfern nötig. Folgender Ansatz für die Nachgiebigkeit $D(t)$ ($D = 1/E$, E : Elastizitätsmodul des Materials) eines Polymers ohne Spannungsvorgeschichte bei einem Spannungssprung ist verbreitet z. B. [19]:

$$D(t) = D_0 + \int_0^\infty f(\tau) \cdot [1 - e^{-t/\tau}] d\tau. \quad (3)$$

Dabei ist D_0 der zeitunabhängige Anteil der Nachgiebigkeit, das Integral beschreibt den zeitabhängigen mechanisch reversiblen Anteil. Die Integrationsvariable τ ist die Retardationszeit und die nichtnegative Funktion $f(\tau)$ ist das Retardationsspektrum. Ist das Retardationsspektrum ein Linienspektrum, dann wird aus Gleichung (3) [19]:

$$D(t) = D_0 + \sum_{i=0}^m c_i \cdot [1 - e^{-t/\tau_i}]. \quad (4)$$

τ_i sind dabei die diskreten Retardationszeiten und c_i sind die Retardationsstärken.

Zur Modellierung des Materialverhaltens wird ein Ansatz gesucht, der nicht nur für einen Drucksprung funktioniert, sondern für beliebige Druckverläufe gilt. Es werden folgende strukturelle Annahmen getroffen: Die Retardation des Materials wirkt sich nur auf die druckunabhängige Komponente des Aktormomentes aus. Der Einfluss des Druckverlaufes auf den Zustand des Materials wird über einen Kriechgrad $A(p_{Aktor}(t)) \in [0, 1]$ modelliert. Der aktuelle Kriechgrad wird durch eine Verschiebung der Geraden $M_{Aktor}(p_{Aktor} = 0, \varphi) = a_0 + a_1\varphi$ in φ berücksichtigt. Eine

Verschiebung dieser Geraden nach rechts, die durch $A > 0$ hervorgerufen wird, bedeutet eine Verringerung der Rückstellkräfte im Aktor und führt dadurch zu einem anwachsenden Gelenkwinkel. Ein weiterer Parameter $\Delta\varphi_{max}$ begrenzt die maximale Verschiebung der Geraden $\Delta\varphi$ und somit den maximalen retardationsbedingten Winkel. Es ergibt sich ein modifiziertes Aktorgesetz:

$$M_{Aktor}(p_{Aktor}(t), \varphi) = a_0 + a_1(\varphi + \underbrace{A(p_{Aktor}(t)) \cdot \Delta\varphi_{max}}_{\Delta\varphi}) + a_2 p_{Aktor} + a_3 p_{Aktor} \varphi. \quad (5)$$

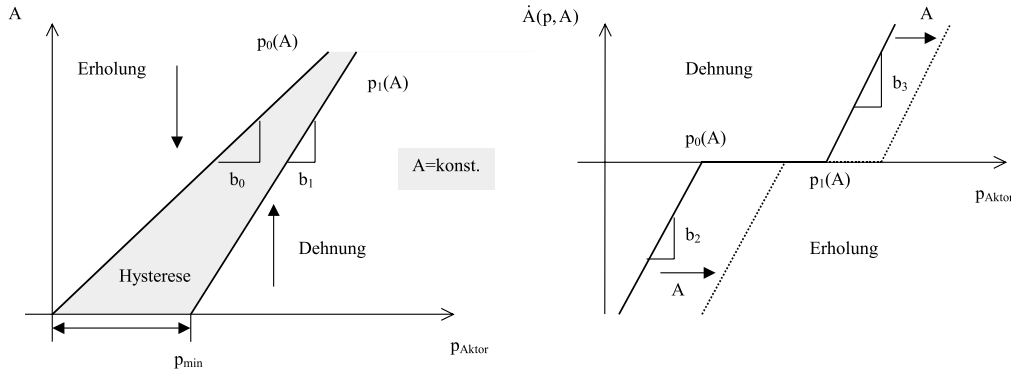


Bild 6: Darstellung der Zusammenhänge des Materialgesetzes

Für die Integration in das Streckenmodell wurde ein Materialgesetz für den Kriechgrad A entworfen, das sowohl das Kriechen als auch die Hysterese des Materials abbildet. In Bild 6 (links) lassen sich drei Bereiche für den Zustand des Materials erkennen. Links der Kurve $p_0(A)$ wird die viskoelastische Dehnung zurückgebildet, zwischen den Kurven $p_0(A)$ und $p_1(A)$ ändert sich die viskoelastische Dehnung nicht (Hysterese) und rechts der Kurve $p_1(A)$ erfolgt weitere viskoelastische Dehnung. A ist dabei der Dehnungsgrad des Materials, dessen Maximum mit steigendem Druck zunimmt und für $p_{Aktor, max}$ gleich eins ist. In Bild 6 (rechts) ist die Änderungsgeschwindigkeit des Kriechgrades (\dot{A}) dargestellt. Es ergeben sich für das Materialgesetz die folgenden Differentialgleichungen:

$$\begin{aligned} \frac{b_1}{b_3} \dot{A} + A &= b_1 \cdot (p_{Aktor} - p_{min}) & : p_{Aktor} \geq p_1(A) \\ \dot{A} &= 0 & : p_0(A) < p_{Aktor} < p_1(A) \\ \frac{b_0}{b_2} \dot{A} + A &= b_0 \cdot p_{Aktor} & : p_{Aktor} \leq p_0(A) \end{aligned} \quad (6)$$

mit der Lösung für $A(t=0) = A_0$:

$$A(t) = \begin{cases} b_1 \cdot (p_{Aktor} - p_{min}) \cdot [1 - e^{-\frac{b_3}{b_1} \cdot t}] + A_0 \cdot e^{-\frac{b_3}{b_1} \cdot t} & : p_{Aktor} \geq p_1(A) \\ A_0 & : p_0(A) < p_{Aktor} < p_1(A) \\ b_0 \cdot p_{Aktor} \cdot [1 - e^{-\frac{b_2}{b_0} \cdot t}] + A_0 \cdot e^{-\frac{b_2}{b_0} \cdot t} & : p_{Aktor} \leq p_0(A) \end{cases} \quad (7)$$

Die Lösung der Differentialgleichung hat die selbe Struktur wie Gleichung (4) mit D_0 entspricht A_0 , c_i entspricht $(b_1 \cdot (p_{Aktor} - p_{min}) - A_0)$ und τ_i entspricht $\frac{b_1}{b_3}$ usw.

Parameter	Beschreibung
T_{Ventil}	Zeitkonstante des Druckregelventils
$a_0 \dots a_3$	Parameter des multilinearen Aktorkennfeldes
J	Trägheitsmoment der bewegten Masse
d	Reibkoeffizient im Gelenk
$b_0 \dots b_3, p_{min}$	Parameter des Materialgesetzes
$\Delta\varphi_{max}$	Maximaler dehnungsbedingter Winkel

Tabelle 1: Liste der zu optimierenden Modellparameter

Die Parameter des Streckenmodells sind in Tabelle 1 zusammengefasst. Zur Identifikation der Streckenparameter werden reale Regelgrößenverläufe $\varphi(t)$ bei vorgegebenen Stellgrößenverläufen $U(t)$ aufgezeichnet und mit den Simulationsergebnissen $\varphi_{sim}(t)$ verglichen (Bild 7). Das entstehende parameternichtlineare Ausgangsfehlerproblem wird mit Matlab numerisch gelöst. Dazu wird in jedem Suchschritt für den aktuellen Parametersatz ein Gütewert $Q(\mathbf{x})$ aus dem Simulationsergebnis mit Simulink bestimmt:

$$Q(\mathbf{x}) = \frac{1}{T_{End}} \left(\lambda \sum_{k=1}^{T_{end}} (\varphi[k] - \varphi_{sim}[k])^2 + (1 - \lambda) \sum_{k=1}^{T_{end}-1} (\Delta\varphi[k] - \Delta\varphi_{sim}[k])^2 \right) \quad (8)$$

Gemäß Gleichung (8) wird aus den Zeitreihen der Messung und der Simulation der Gütewert aus dem absoluten Abstand mit der Wichtung $\lambda < 1$ und der Steigungsdifferenz mit der Wichtung $(1 - \lambda)$ der Kurven berechnet. T_{End} entspricht dabei der Anzahl der Abtastzeitpunkte, und \mathbf{x} dem Parametervektor. Ein Algorithmus bestimmt aus den vorhergehenden Gütewerten und dem aktuellen Parametersatz den nächsten Parametersatz. Es werden so lange neue Parametersätze berechnet, bis keine weitere Verbesserung des Gütewertes mehr eintritt.

In Bild 7 ist das Ergebnis der Optimierung abgebildet. Der Kurvenverlauf für die Simulationen ohne Materialgesetz (links) und mit Materialgesetz (rechts) ist jeweils zusammen mit den Messwerten dargestellt. Das Materialgesetz verbessert die Simulationsergebnisse deutlich. Eine weitere Annäherung ist nur mit einer größeren Anzahl Zeitkonstanten entsprechend Gleichung (4) möglich.

Für die Auslegung des Reglers sind weitere Untersuchungen der Differentialgleichung des Streckenmodells notwendig. Die Abhängigkeiten der Streckenverstärkung und der Zeitkonstanten des Systems vom Arbeitspunkt (Gelenkwinkel) werden untersucht. Zu diesem Zweck wird die Differentialgleichung des Gelenkes (2) linearisiert.

Mit der Approximation für das Aktorkennfeld (1) und dem Einfluss des Materialgesetzes (5) wird aus (2):

$$J\ddot{\varphi} + d\dot{\varphi} = a_0 + a_1\varphi + a_2p + a_3\varphi p - a_1A\Delta\varphi_{max} + M_{Stoer} = 0 \quad (9)$$

Für die weitere Rechnung wird der Ausdruck $-a_1A\Delta\varphi_{max} + M_{Stoer}$ zu M_{Stoer}^* zusammengefasst.

Gleichung (9) wird am Arbeitspunkt (φ_{AP}, p_{AP}) für ein konstantes $M_{Stoer,AP}^*$ linearisiert und das zugehörige Kleinsignalverhalten (φ_{KS}, p_{KS}) ausgewertet.

$$\varphi = \varphi_{AP} + \varphi_{KS}, \quad p = p_{AP} + p_{KS} \quad (10)$$

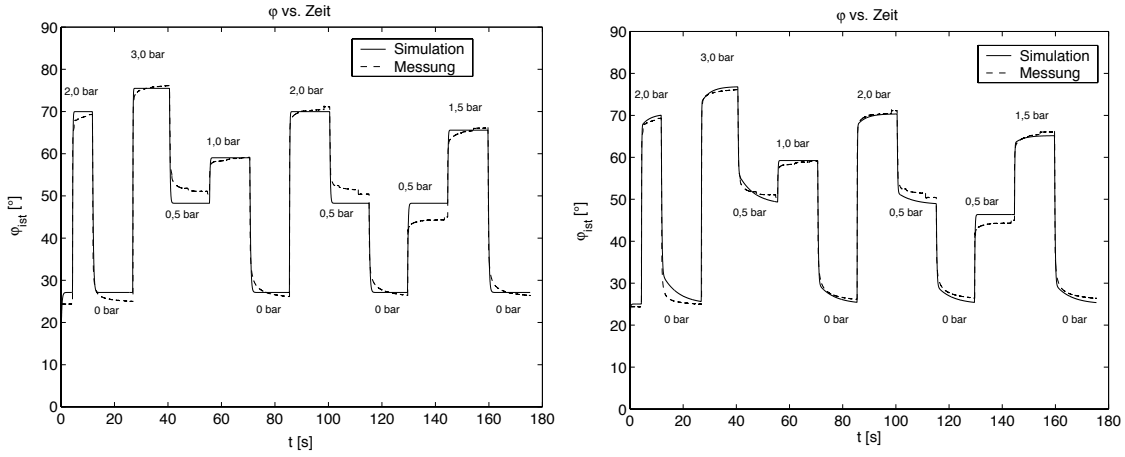


Bild 7: Ergebnisse der Optimierung ohne Materialgesetz (links) und mit Materialgesetz (rechts)

Mit der Bedingung für das statische Gleichgewicht in der Ruhelage

$$a_0 + a_1\varphi_{AP} + a_2p_{AP} + a_3\varphi_{APP_{AP}} + M_{Stoer}^* = 0 \quad (11)$$

ergibt sich:

$$J\ddot{\varphi}_{KS} + d\dot{\varphi}_{KS} = a_1\varphi_{KS} + a_2p_{KS} + a_3\varphi_{APP_{KS}} + a_3p_{AP}\varphi_{KS}. \quad (12)$$

Umstellen, Laplace-Transformation und eliminieren von p_{AP} mittels Gleichung (13) führt zur Übertragungsfunktion $G_p(s)$:

$$p_{AP}(\varphi_{AP}, M_{Stoer,AP}^*) = \frac{-M_{Stoer}^* - a_0 - a_1\varphi_{AP}}{a_2 + a_3\varphi_{AP}} \quad (13)$$

$$G_p(s) = \frac{\overbrace{\frac{K_S(\varphi_{AP})}{(a_2 + a_3\varphi_{AP})^2}}^{K_S(\varphi_{AP})}}{\underbrace{\frac{J}{(a_0 + M_{Stoer}^*)a_3 - a_1a_2}}_{T_1(\varphi_{AP}) \cdot T_2(\varphi_{AP})} s^2 + \underbrace{d \frac{a_2 + a_3\varphi_{AP}}{(a_0 + M_{Stoer}^*)a_3 - a_1a_2}}_{T_1(\varphi_{AP}) + T_2(\varphi_{AP})} s + 1}. \quad (14)$$

Für die beiden Zeitkonstanten folgt:

$$T_{1/2} = \frac{d \cdot H}{2} \pm \sqrt{\frac{(d \cdot H)^2}{4} - J \cdot H} \text{ mit } H(\varphi_{AP}) = \frac{a_2 + a_3\varphi_{AP}}{(a_0 + M_{Stoer}^*)a_3 - a_1a_2} \quad (15)$$

Die Verstärkung der Strecke ist ebenso wie die beiden Zeitkonstanten abhängig vom Arbeitspunkt φ_{AP} und dem Störmoment M_{Stoer}^* . Wegen $J \ll 1$ ist der Term $J \cdot H$ zu vernachlässigen. Damit ergibt sich für die Zeitkonstanten:

$$T_1 = d \cdot H, \quad T_2 \approx 0 \quad (16)$$

In Bild 8 (links) sind die Kurven von $K_S(\varphi_{AP})$ und $T_1(\varphi_{AP})$ für die ermittelten Parameter dargestellt. A_{min} und A_{max} sind die für den Arbeitspunkt minimal beziehungsweise maximal möglichen statischen Werte von A (Bild 6), die über M_{Stoer}^*

auf K_S , T_1 und φ_{AP} einwirken. Aus Gleichung (11) ergibt sich auch die statische Kennlinie des Gelenkes (Bild 8 rechts):

$$\varphi_{AP} = \frac{-M_{Stoer}^* - a_0 - a_2 p_{AP}}{a_1 + a_3 p_{AP}} \quad (17)$$

Zu beachten ist dabei, dass Verstärkung und Zeitkonstante der Regelstrecke in starkem Maße von Druck- und Winkelarbeitspunkt und vom Störmoment abhängen.

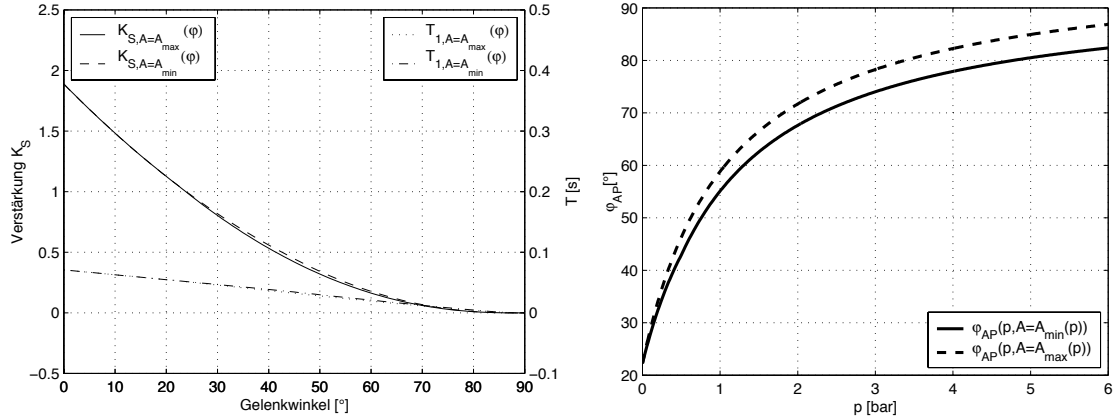


Bild 8: T_1 und K_S in Abhängigkeit von φ_{AP} (links), statische Kennlinien $\varphi_{AP}(p)$ (rechts), für $M_{Stoer} = 0$

3 Reglerstruktur

Technische PID-Regler sind durch die Übertragungsfunktion

$$G_R(s) = K_R \left(1 + \frac{1}{T_N s} + \frac{T_V s}{1 + T_Z s} \right) \quad (18)$$

gekennzeichnet. Dabei ist K_R die Reglerverstärkung, T_N die Nachstellzeit, T_V die Vorhaltezeit und T_Z die Verzögerung des realen D-Gliedes. Eine zeitdiskrete Realisierung mit der Abtastzeit T_a und Zeitpunkten $k \cdot T_a$, $k \in \mathbb{N}$ ergibt sich mittels:

$$u[k+1] = \underbrace{K_R e[k+1]}_{u_P[k+1]} + \underbrace{u_I[k] + \frac{K_R T_A}{T_N} e[k]}_{u_I[k+1]} + \frac{K_R T_V}{T_Z} \left(e^{-\frac{T_A}{T_Z}} u_D[k] + \underbrace{e[k+1] - e[k]}_{u_D[k+1]} \right) \quad (19)$$

mit den Stellgrößenanteilen $u_P[k]$ (P-), $u_I[k]$ (I-) und $u_D[k]$ (D-Anteil).

Reale Stellglieder sind stets durch eine Beschränkung des Stellbereiches charakterisiert, hier minimaler und maximaler Druck p_{Aktor} . Diese Begrenzung muss im Regelalgorithmus für den digitalen Regler berücksichtigt werden. Lediglich den Ausgang des Reglers zu begrenzen [20] löst das Problem nicht. Eine bleibende Regelabweichung an der Begrenzung des Stellgliedes lässt den I-Anteil stetig weiter hochlaufen und führt zu einer verzögerten Reaktion auf einen Sprung der Führungsgröße bis hin zur Instabilität des Reglers.

Als sogenannte "Anti-Reset-Windup"-Maßnahmen wurden verschiedene Ansätze vorgeschlagen [20, 21]. Für die Fingergelenkregelung wird eine doppelte Stellbegrenzung verwendet. Neben der Begrenzung des Reglerausgangs wird zusätzlich der I-Anteil mit den selben Schranken begrenzt.

Nichtlineares Verhalten der Strecke kann durch Adaption der Reglerparameter K_R , T_N und T_V kompensiert werden. Bei der Implementierung des Algorithmus (19) ist auf die Anordnung der Multiplikationsstellen zu achten, um unerwünschte Sprünge der Stellgröße zu vermeiden. Diese treten bei Veränderung von K_R beziehungsweise T_N auf, wenn erst nach der Summenbildung des I-Gliedes mit dem Faktor $\frac{K_R T_A}{T_N}$ multipliziert wird. Bei der T_V -Adaption darf nicht vor der Differenzbildung multipliziert werden. Alternativ zur Berechnung gemäß (19) kann der D-Anteil über die Regelgröße bestimmt werden:

$$u_D[k+1] = y[k+1] - y[k]. \quad (20)$$

Ein Unterschied ergibt sich nach einem Führungsgrößenprung lediglich in der ersten Iteration.

In Bild 9 ist ein Blockschaltbild des Algorithmus (19) dargestellt, bei dem unerwünschte Sprünge der Stellgröße nicht auftreten.

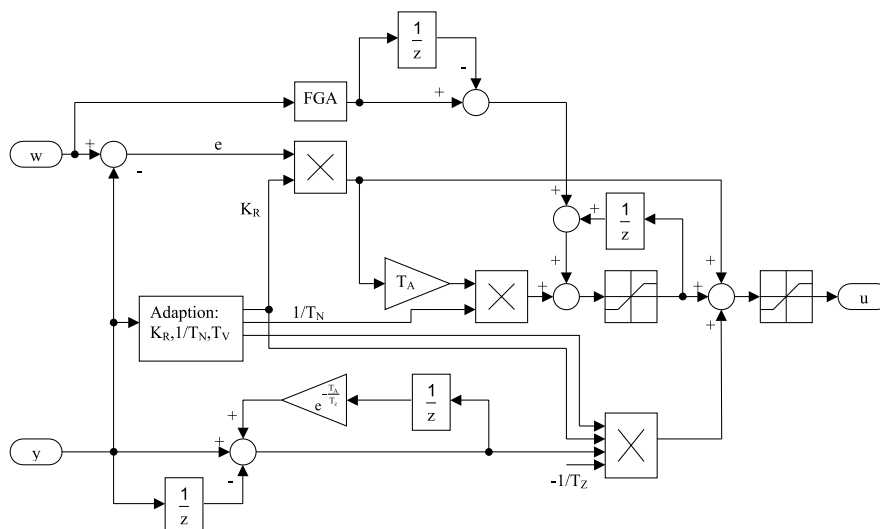


Bild 9: Blockschaltbild des diskreten adaptiven PID-Reglers mit Führungsgrößen-aufschaltung (FGA)

4 Experimentelle Erprobung

Für die Positionsregelung des Gelenkes werden Experimente mit einem adaptiven PID Regler entsprechend Abschnitt 3 durchgeführt. Die Adaption ist entweder als Fuzzy-Adaption oder als vollständige Kompensation der nichtlinearen Kennlinie der Strecke ausgeführt. Der D-Anteil wird über die Regelgröße berechnet. Der Regler ist in einem PC mittels der Prozesssteuerungssoftware LabView implementiert. Die Strecke ist über geeignete I/O-Hardware mit dem Rechner verbunden.

Führungsgröße des Reglers ist der Sollwinkel des Gelenkes. Die Stellgröße ist die Spannung am Druckregelventil.

Die viskoelastischen Eigenschaften des Aktormaterials sind dafür verantwortlich, dass sich das Gelenk stets aus einem eingeregelter Zustand heraus bewegt und so für eine fortwährende Regleraktivität sorgt. Dieser Effekt ist insbesondere bei der hydraulischen Realisierung mit ständigen Ventil- und Pumpenaktivitäten verbunden. Der Regler wird daher mit einer Totzone versehen, die bewirkt, dass die Regleraktivität unterdrückt wird, sobald das Gelenk den Sollwinkel mit einer Abweichung, die dem Betrag nach kleiner als $\Delta\varphi_{tot\,ein}$ ist, erreicht hat. Die Regleraktivität wird solange unterdrückt, bis sich das Gelenk um einen Winkel $\Delta\varphi_{tot\,aus} > \Delta\varphi_{tot\,ein}$ vom Sollwinkel entfernt hat (21). Innerhalb der Totzone wird die Regeldifferenz zu null gesetzt. Die Stellgröße des Reglers ist dann U_I . Wird das D-Glied über die Regelgröße berechnet, ist anstelle der Regeldifferenz die Reglerverstärkung K_R zu null zu setzen, um die Regleraktivität vollständig zu unterdrücken.

$$K_R[k+1] = \begin{cases} K_R^*[k+1] & \text{für } |e[k+1]| > \Delta\varphi_{tot\,aus} \\ & \text{oder } |e[k+1]| > \Delta\varphi_{tot\,ein} \text{ und } K_R[k] \neq 0 \\ 0 & \text{für } |e[k+1]| < \Delta\varphi_{tot\,ein} \\ & \text{oder } |e[k+1]| < \Delta\varphi_{tot\,aus} \text{ und } K_R[k] = 0 \end{cases} \quad (21)$$

Die Kompensation der variablen Streckenverstärkung $K_S(\varphi_{AP})$ erfolgt entweder durch eine Fuzzy-Adaption oder durch eine vollständige Kompensation von K_R . Die variable Zeitkonstante $T_1(\varphi_{AP})$ wird in beiden Fällen durch eine Fuzzy-Adaption von T_N kompensiert (22). Eine vollständige Adaption bringt keine Vorteile, da $T_1(\varphi_{AP})$ näherungsweise linear ist, siehe Gleichung (16). Bei der Fuzzy-Adaption werden für zwei Werte φ_0 und φ_1 der Regelgröße die optimalen Reglerparameter zum Beispiel K_{R_0} und K_{R_1} mittels der linearisierten Übertragungsfunktion der Strecke (14) bestimmt. $K_R(\varphi)$ berechnet sich dann wie folgt (Bild 10 (links)):

$$K_R(\varphi) = \begin{cases} K_{R_0} & : \varphi \leq \varphi_0 \\ K_{R_0} + \mu(\varphi)K_{R_1} & : \varphi_0 < \varphi < \varphi_1 \text{ mit } \mu(\varphi) = \frac{\varphi - \varphi_0}{\varphi_1 - \varphi_0}, M_{Stoer} = 0. \\ K_{R_1} & : \varphi \geq \varphi_1 \end{cases} \quad (22)$$

Bei der vollständigen Kompensation wird für jeden Wert zwischen φ_0 und φ_1 der optimale Wert für K_R bestimmt (Bild 10 (rechts)):

$$K_R(\varphi) = \begin{cases} K_{R_0} & : \varphi \leq \varphi_0 \\ \frac{K}{K_S(\varphi)} & : \varphi_0 < \varphi < \varphi_1 \text{ mit } M_{Stoer} = 0. \\ K_{R_1} & : \varphi \geq \varphi_1 \end{cases} \quad (23)$$

Dabei ist K der Sollwert für die Verstärkung des geschlossenen Regelkreises und K_S die Streckenverstärkung.

Reine Regelungen sind häufig zu langsam. Die Kombination der Regelung mit einer Steuerung über eine Führungsgrößenumschaltung kann von Vorteil sein. Änderungen der Führungsgröße werden dabei anhand der inversen statischen Kennlinie (13) $p_{AP}(w, 0)$ auf den I-Anteil des Reglers aufgeschaltet (siehe Bild 9):

$$u_I[k+1] = \alpha_F \cdot (p_{AP}[k+1] - p_{AP}[k]) + u_I[k] + \frac{K_R}{T_N} T_A e[k] \quad (24)$$

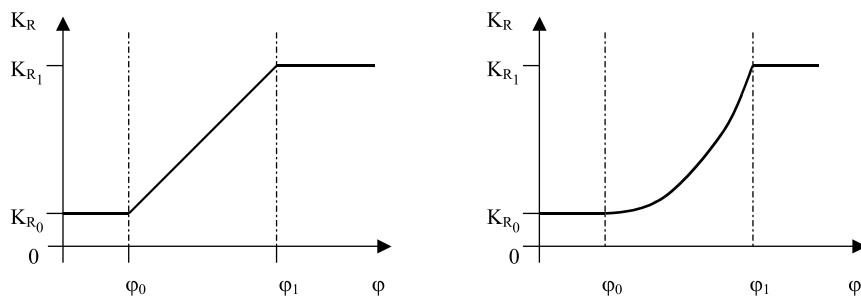
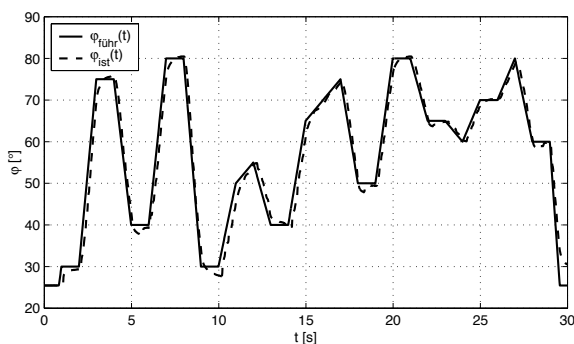


Bild 10: Berechnung der Fuzzy-Adaption (links) und der vollständigen Kompensation (rechts) am Beispiel von K_R

Dabei ist α_F ein konstanter Faktor (hier 0,5), der den Einfluss der Führungsgrößenaufschaltung bestimmt. Die Aufschaltung ist für einen Streckenzustand ausgelegt und somit empfindlich gegenüber Änderungen der Streckenparameter. Eine Änderung der Streckenparameter entspricht beim Fingergelenk der Verschiebung der statischen Kennlinie φ_{AP} durch den Einfluss des Kriechgrades des Aktormaterials (Bild 8 (rechts)) oder durch ein äußeres Störmoment.



Auswertung K_R -Adaption			
	VK	FA	ohne
$ \overline{\Delta\varphi} [^\circ]$	2,1	2,8	5,5
$\sqrt{\overline{\Delta\varphi^2}} [^\circ]$	2,8	3,8	7,3
$\sum \Delta U_{PID} $	47,9	37,5	30,5
$\sum \Delta U_{PID}^2$	6,0	2,3	1,15

Bild 11: Ergebnisse der Reglererprobung ($M_{Stoer} = 0$), VK: vollständige Kompensation, FA: Fuzzy-Adaption

Bild 11 (links) zeigt einen rampenförmigen Führungsgrößenverlauf und den Verlauf der Regelgröße. Der verwendete Regler kompensiert $K_S(\varphi)$ vollständig. Die Zeitkonstante $T_1(\varphi)$ wird über eine Fuzzy-Adaption kompensiert. In der Tabelle in Bild 11 (rechts) sind Zahlenwerte für die Regeldifferenz und den Stellaufwand angegeben. Im Vergleich zur vollständigen K_R Kompensation zeigt der Regler mit K_R Fuzzy-Adaption größere Abweichungen der Regelgröße. Der Stellaufwand ist dagegen geringer. Im Vergleich zum PID-Regler ohne Adaption ist der Vorteil beider Varianten deutlich. Wird statt eines rampenförmigen Führungsgrößenverlaufes ein sprungförmiger verwendet, verschwinden die Unterschiede in der Regeldifferenz und dem Stellaufwand zwischen vollständiger Kompensation und Fuzzy Adaption.

5 Zusammenfassung und Ausblick

In dieser Arbeit wurde die Modellbildung und der Fuzzy-adaptive PID-Regler für die Fingergelenke einer pneumatisch betriebenen Roboterhand vorgestellt. Für das

viskoelastische Verhalten des Aktormaterials wurde ein Modellansatz entwickelt und dadurch die Güte des Modells der Regelstrecke verbessert. Die nichtlineare statische Kennlinie der Strecke sowie die variable Zeitkonstante wurden herausgearbeitet und über die Adaption des Reglers kompensiert. Varianten der Adaption wurden vorgestellt und anhand von Messwerten am realen Regler bewertet.

Derzeit werden Versuche zur Identifikation der hydraulischen Strecke als Vorbereitung für die hydraulische Fingergelenkregelung durchgeführt. Ferner wird der Einfluss mehrerer gekoppelter Fingergelenke auf die Güte der Regelung des einzelnen Gelenkes untersucht.

Die Autoren danken an dieser Stelle der DFG für die finanzielle Unterstützung der Forschungsarbeiten im Rahmen des SFB 588 "Lernende und kooperierende modale Roboter"

Literatur

- [1] Butterfass, J.; Grebenstein, M.; Liu, H.; Hirzinger, G.: DLR Hand II: Next generation of a dextrous robot hand. In: *International Conference on Robotics and Automation, Seoul, Korea*. 2001.
- [2] Lovchik, C. S.; Diftler, M. A.: The Robonaut Hand: A Dexterous Robot Hand for Space. In: *Proceedings of the IEEE ICRA*, S. 907–913. Detroit. 1999.
- [3] Fukaya, N.; Toyama, S.; Asfour, T.; Dillmann, R.: Design of the TUAT/Karlsruhe Humanoid Hand. *IEEE/RSJ ICIRS* (2000).
- [4] Schulz, S.; Pylatiuk, C.; Martin, J.; Bretthauer, G.: Die Anthropomorphe FZK-Hand. In: *Robotik 2002 Leistungsstand - Anwendungen - Visionen-Trends (VDI-Berichte 1679)* (Dillmann, R.; Schraft, R.; Wörn, H., Hg.), S. 531–536. VDI/VDE- Gesellschaft Mess- und Automatisierungstechnik. 2002.
- [5] Schulz, S.; Pylatiuk, C.; Bretthauer, G.: Entwicklung einer künstlichen Hand mittels flexibler Fluidaktoren. *VDI Bericht Nr. 1680, GMA Kongress* (2001), S. 143ff.
- [6] Schulz, S.; Pylatiuk, C.; Bretthauer, G.: A New Class of Flexible Fluidic Actuators and their Applications in Medical Engineering. *at-Automatisierungstechnik* Band 47, Heft 8 (1999), S. 390ff.
- [7] Schulz, S.; Pylatiuk, C.; Bretthauer, G.: A new ultralight anthropomorphic hand. In: *International Conference on Robotics and Automation, Seoul, Korea*. 2001.
- [8] Reischl, M.; Mikut, R.; Pylatiuk, C.; Schulz, S.; Beck, S.; Bretthauer, G.: Steuerungs- und Signalverarbeitungskonzepte für eine multifunktionale Handprothese. *at-Automatisierungstechnik* Band 50, Heft 6 (2002), S. 279–286.
- [9] Bobrow, J. E.; McDonell, W.: Modeling, Identification, and Control of a Pneumatically Actuated, Force Controllable Robot. *IEEE Transactions on Robotics and Automation* 14 (1998), S. 732–742.

- [10] Neumann, R.; Göttert, M.; Ohmer, M.: Servopneumatik - eine alternative Antriebstechnik für Roboter. In: *Robotik 2002 Leistungsstand - Anwendungen - Visionen-Trends (VDI-Berichte 1679)* (Dillmann, R.; Schraft, R.; Wörn, H., Hg.), S. 537–544. VDI/VDE- Gesellschaft Mess- und Automatisierungstechnik. 2002.
- [11] Van der Smagt, P.; Groen, F.; Schulten, K.: Analysis and Control of a Rubber-tuator Arm. *Biological Cybernetics* 75 (1996), S. 433–440.
- [12] Loose, T.: *Aufbau und Integration der Positionsregelung für eine fluidische Handprothese*. Diplomarbeit, Universität Karlsruhe (TH). 2001.
- [13] Kuntze, H.-B.; Jacubasch, A.: Anwendung von Fuzzy-Logic zur intelligenten anwendungsspezifischen Regelung von Robotern. In: *Fuzzy Logik und neuronale Netze in Robotik und Automation. Anwenderseminar Fuzzy Karlsruhe '98*. Univeristät Karlsruhe (TH). 1998.
- [14] Beck, S.: *Entwicklung, Aufbau und Erprobung einer Testumgebung für die Evaluierung flexibler Fluidaktoren*. Diplomarbeit, Universität Karlsruhe (TH). 2001.
- [15] Hellrich, W.; Harsch, G.; Haenle, S.: *Werkstoffführer Kunststoffe*. Carl Hanser Verlag München Wien. 1996.
- [16] Menges, G.: *Werkstoffkunde Kunststoffe*. Carl Hanser Verlag München Wien. 1998.
- [17] Schatt, W.: *Einführung in die Werkstoffwissenschaft*. VEB Deutscher Verlag für Grundstoffindustrie, Leipzig. 1983.
- [18] Tschoegl, N.: *The Phenomenological Theory of Linear Viscoelastic Behavior*. Springer Verlag Berlin Heidelberg New York. 1989.
- [19] Schwarzl, F.: *Polymer-Mechanik*. Springer Verlag. 1990.
- [20] Föllinger, O.: *Regelungstechnik - Einführung in die Methoden und ihre Anwendung*. Hüthig Buch Verlag Heidelberg. 1994.
- [21] Unbehauen, H.: *Regelungstechnik I*. Verlag Vieweg. 2001.

Neuronale Netze zur Kompensation stückweiser stetigen Nichtlinearitäten in Regelungssystemen

D. Karimanzira

Technische Universität Ilmenau
Fakultät für Informatik und Automatisierung
Postfach 10 0565, 98684 Ilmenau
e-mail: divas.karimanzira@systemtechnik.tu-ilmenau.de

Kurzfassung

Viele industrielle Prozesse enthalten stückweise stetige Nichtlinearitäten. Diese Beinhalten Reibung, Totezone, Sättigung, Hysterese u.a. In diesem Beitrag wird ein neuronale Netzwerkkonzept zur Kompensation solcher stückweise stetige Nichtlinearitäten mit bekannten Unstetigkeitsstellen in der Regelungstechnik vorgestellt. Besonders an dem System ist, dass zur Kompensation der un stetigen Nichtlinearitäten eine spezielle neuronale Netzwerkstruktur benutzt werden wird. Diese Art von mehrschichtigem Perceptron (MLP) enthält extra Neuronen in der Zwischenschicht die mit Sprungaktivierungsfunktionen bestückt sind. Die Kompensation der un stetigen Nichtlinearitäten erfolgt auf ähnliche Weise wie bei der on-line Trainingsmethode für die Systeminverse nach Psaltis [6]. Dabei werden zwei Neuronale Netzwerke (NN) benötigt, eine normale MLP zur Modellbildung des stetigen Nichtlinearität, z.B. Totezone und ein spezielles MLP zur Kompensation der Nichtlinearitäten, da z.B. die Inverse eines Totezones, ein stückweise stetige Funktion mit einer Unstetigkeitsstelle im Ursprung ist. Im Fall einer Reibung gilt der umgekehrte Fall. Die Leistungsfähigkeit des Systems wird dann anhand eines Regelungssystems mit unbekannter, unsymmetrischen Totezone erprobt.

1. Einleitung

Stückweise stetige Nichtlinearitäten treten sehr oft in vielen physikalischen Komponenten der Regelungssysteme, solche wie, Antriebe und hydraulische Servoventile. Solche Nichtlinearitäten beinhalten Sättigung, Relais, Hysterese, und Totezone, und sind meistens unbekannt und zeitvariant, z.B. eine häufige Quelle für Nichtlinearitäten entsteht aus Reibung, die sich mit der Temperatur und Verschleiß, und unterscheidet sich signifikant für Komponente, die von der Masse abhängig sind. Deshalb ist die Untersuchung der Methode, die sich mit der Kompensation von solchen Nichtlinearitäten von großem Interesse ist. Totezonen und Sättigungen sind vor allem von großem Interesse, und erzeugen gute Modelle für viele un stetige Nichtlinearitäten, die in der Praxis gefunden werden kann.

Standard Regler wie, PD und PID-Reglern, die in der Praxis häufig benutzt werden, leiden unter schlechter Leistungsfähigkeit, wenn sie direkt an Systemen mit Totezonen angewandt wird, z.B. eine Regelabweichung (steady state error) bleibt, wenn ein PD-Regler zur Regelung eines Systems mit einer Totezone angewendet wird. – Die Größe der bleibenden Regelabweichung erhöht sich mit der Breite der Totezone. Die Regelabweichung bleibt, da ein PD-Regler nur den Ausgangsfehler und die Änderung des Ausgangsfehlers als Eingänge des Reglers benutzt. Um die bleibende Regelabweichung eliminieren zu können, ein PID-Regler kann benutzt werden. Dies enthält ein Integrator des Ausgangsfehlers als Eingang des Reglers. Jedoch, wie wir sehen könnten der Übergangsverhalten ist in diesen Fällen schlecht.

Viele fortgeschrittene Regelungsschemata zur Regelung von Systemen mit un stetigen Nichtlinearitäten, beinhalten "sliding mode control" [1], und "dithering" [2]. Motiviert durch die Beschränkungen in diesen Methoden, wie "chattering" in sliding mode control, Recker et al. [3] schlägt ein adaptives nichtlineares Regelungsschema für die Regelung von Systemen mit Totzonen vor. In [3], vollständige Zustandsmessungen wurden als vorhanden angenommen. Obwohl einige Ergebnisse zur Kompensation un stetiger Nichtlinearitäten mit neuronalen Netzen existieren, wurde beobachtet, dass Versuche stückweise stetige Funktionen mit Hilfe von MLP-Netzen mit stetigen Aktivierungsfunktionen sehr viele Neuronen und viele Trainingsschritte brauchen, und es wird dennoch kein sehr gutes Ergebnis erzielt. Es wurde herausgestellt [2], dass um solche stückweise stetigen Funktionen ausreichend approximieren zu können, notwendig sei dem MLP mit zusätzlichen Knoten mit Sprungaktivierungsfunktionen zu bestücken.

2. Approximation stückweise stetiger Funktionen mit MLP-Netzen

Zunächst wird eine allgemeine Totzone (Abb. 2.1(a)) beschrieben durch Gleichung (1) [7], betrachtet. Es wird angenommen, dass die Funktionen $h(u)$ und $g(u)$ stetig, nichtlinear und invertierbar sind, wobei $h(u)$, $g(u)$, d_+ und d_- unbekannt sein können.

$$\tau = D[u] = \begin{cases} h(u), & u > d_- \\ 0, & -d_- \leq u \leq d_+ \\ g(u), & u < -d_+ \end{cases} \quad (1)$$

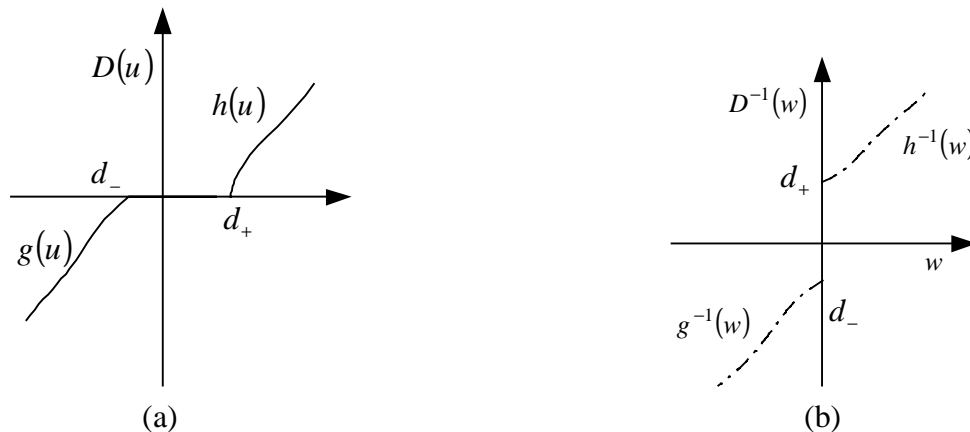


Abbildung 2.1: (a) Unsymmetrische Totzone und deren (b) Inverse

Um solche Funktionen, wie die Inverse der Totzone (Abb. 2.1(b)) genügend genau approximieren zu können, ist ein MLP mit stetigen und zusätzlichen un stetigen Aktivierungsfunktionen der Neuronen notwendig und wird zunächst vorgestellt.

Unter dem Einsatz der Ergebnisse von Cybenko [2] für stetige Funktionapproximation, kann man das folgende Ergebnis für die Approximation von Funktionen mit endlicher Sprung formulieren.

Es existiert eine Funktion $f : S \rightarrow \mathfrak{R}$, die auf der Kompaktmenge S stetig sei, außer an der Stelle $x = c$, an der die Funktion einen endlichen Sprung hat und stetig von rechts ist. Für alle $\varepsilon > 0$, existiert eine Summe F der folgenden Form:

$$F(x) = \sum_{k=0}^N \alpha_k \sigma(m_k + n_k) + \sum_{k=0}^M a_k \varphi_k(x - c), \text{ so dass}$$

$$|F(x) - f(x)| < \varepsilon, \text{ für alle } x \text{ in } S,$$

wo $\sigma(x)$ eine Sigmoid-Funktion ist, und

die Sigmoid-Sprungfunktion φ_k durch definiert wird:

$$\varphi_k(x) = \begin{cases} 0, & x < 0 \\ (1 - e^{-x})^k, & x > 0 \end{cases} \quad (2)$$

Das modifizierte MLP ist in Abb.2.2 dargestellt. Vergleicht man ein Standard-MLP mit einem modifizierten, sieht man, dass das modifizierte NN mit $\sigma(x)$ und φ_k zwei Arten von Aktivierungsfunktionen der versteckten Neuronen hat, wobei φ_k durch Gl. (2) definiert ist. Es gibt zwei Sätze von Gewichten für die erste Schicht (V_1 und V_2) und zwei Sätze von Gewichten für die zweite Schicht (W_1 und W_2). Die Gewichte V_1 werden wie beim Standard-NN trainiert. Der Bias wird entsprechend den bekannten Unstetigkeitsstellen c festgesetzt. Da die Sprungstellen in vielen industriellen Systemen (z.B. Reibung $c = 0$, die Inverse der Totzone $c = 0$ (siehe Abb. 2.1(b))) bekannt sind, sind diese Netzwerktypen sehr nützlich in industriellen Anwendungen.

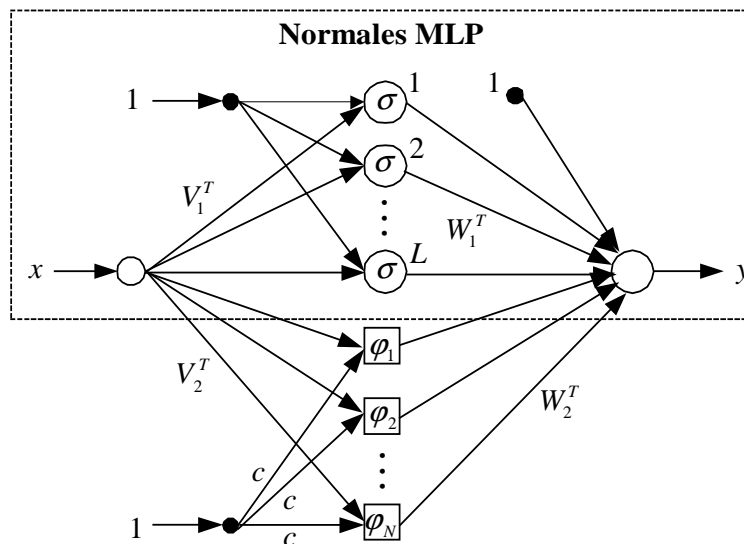


Abbildung 2.2: Struktur des modifizierten MLPs

Ein Beispiel für eine stückweise stetige Funktion mit Sprungstelle im Ursprung wird durch Gleichung (3) beschrieben.

$$v = \sin(u) - \cos^2(u) + \text{sign}(u) + 1 \quad (3)$$

Die Ergebnisse des Vergleichs der Simulation mit einem Standard-MLP und einem modifizierten MLP sind in Abb. 2.3 dargestellt. Beide MLPs haben $H = 20$ versteckte sigmoid-Neuronen und wurden mit der gleichen Epochenzahl (10 bzw. 300), wobei das modifizierte MLP 2 zusätzliche φ -Neuronen besitzt. In den beiden Diagrammen ist deutlich zu sehen, wie schnell das modifizierte MLP durch den Bias die Unstetigkeitsstelle findet.

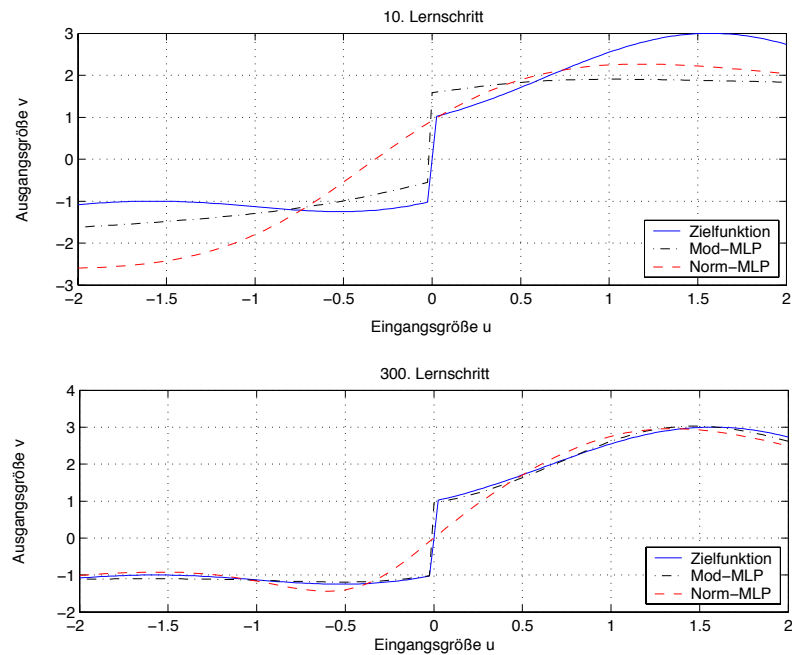


Abbildung 2.3: Ergebnisse des Vergleichs zwischen einem normalen MLP und einem modifizierten MLP

Aus der Simulation wird deutlich, dass das modifizierte MLP-Netz besser unstetige Funktionen approximieren kann und deshalb ein gutes Werkzeug für die Kompensation von Totzonen, Reibungen u.a ist, wobei zwei oder drei φ -Neuronen für viele industrielle Anwendungen ausreichend sind.

Die Anzahl der φ -Neuronen hängt von der Anzahl der Unstetigkeitsstellen und der Anzahl der Eingänge ab, d.h. $M = \sum_{i=1}^n v_i$, mit $n =$ Anzahl der Eingänge, $v_i =$ Anzahl der Unstetigkeiten für den i -ten Eingang. So ist z.B. für einen Roboterarm mit zwei Gelenken wegen der Reibung in jedem der zwei Gelenke $M = 2$. Die feste Werte der Bias sind gleich den Unstetigkeitsstellen.

3. Kompensation von Totezonen mit MLP-Netzen

Sei am Eingang zum Prozeß eine Totezone-Nichtlinearität enthalten, so dass $\tau(t) = D(u)$, mit $u(t)$ als Steuersignal. Bild 2.1(a) zeigt ein nicht symmetrischen Totezone $D(u)$, wo u und τ skalas sind. Im allgemeinen, u und τ sind Vektoren. Ein mathematisches Model, die die Charakteristik einer unsymmetrischen Totezone beschreibt ist in Gl. (4) gegeben.

Es wird angenommen, dass $g(u)$ und $h(u)$ stetige und invertierbare kontinuierliche Funktionen sind. Diese Funktionen sind sehr allgemein. Deshalb beschreibt die Gleichung in Gl. (4) eine allgemeinen Struktur von Totezonen. $g(u)$, $h(u)$ d_+ und d_- wird als unbekannt angenommen, so dass die Kompensation dadurch nicht einfach wird.

Um die schlechte Einwirkung der Totezone kompensieren zu können, wird ein Kompensator vorgeschaltet. Die gewünschte Funktion des vorgeschalteten Kompensators ist es die gesamte

Durchgangsverhalten von w zu τ gleich 1 zu machen. Um diese Effekte zu erzielen, ist es notwendig die vorgeschaltete Inverse des Totezones zu generieren.

Fuzzy und neuronale Techniken zur Kompensation von Totezonen sind in [1, 5, 7] gegeben. Durch die Annahme, dass die Funktion $D()$ recht invertierbar ist, existiert ein Inverse $D^{-1}(w)$, so dass $D(D^{-1}(w)) = w$. Die Funktion $D^{-1}(w)$ ist in Bild 2.1(b) dargestellt. Das mathematische Modell für die Funktion in Bild 2.1(b) ist in Gl. 4 gegeben.

$$D^{-1}(w) = \begin{cases} h^{-1}(w), & w > 0 \\ 0, & w = 0 \\ g^{-1}(w), & w < 0 \end{cases} \quad (4)$$

Die Inverse des Totezones $D^{-1}(w)$ kann in Äquivalenzform wie in Gl. 6 ausgedrückt werden: $D^{-1}(w) = w + w_{NN}(w)$, die aus eine direkte Feedforward-Term plus eine Korrekturterm besteht. Hierbei die modifizierte Inverse des Totezones w_{NN} ist gleich:

$$w_{NN}(w) = \begin{cases} h^{-1}(w) - w, & w > 0 \\ 0, & w = 0 \\ g^{-1}(w) - w, & w < 0 \end{cases} \quad (5)$$

Die Funktion $w_{NN}(w)$ ist diskontinuierlich im Arbeitspunkt 0. Durch eine neuronale Netzwerkapproximation des Totezones folgt:

$$\tau = D(u) = W^T \sigma(V^T u + v_0) + \varepsilon(u) \quad (6)$$

Ein zweites neuronale Netzwerk beschreibt die Approximation des modifizierten Inversefunktion des Totezones:

$$w_{NN}(w) = W_i^T \sigma_i(V_i^T w + v_{0i}) + \varepsilon_i(w) \quad (7)$$

mit: ε_u , $\varepsilon_i(w)$ – Netzwerkfehler,
 W , W_i – ideale Gewichte des Netzwerkes.

Die ideale Gewichte des Netzwerkes sind unbekannt. Die Approximation des nichtlinearen Totezones und modifizierte Totezoneninverse sind in Gleichungen (Gl. 8 bzw. Gl. 9) gegeben.

$$\hat{\tau} = \hat{D}(u) = \hat{W}^T \sigma(V^T u + v_0) \quad (8)$$

$$\hat{w}_{NN}(w) = \hat{W}_i^T \sigma_i(V_i^T w + v_{0i}) \quad (9)$$

$\hat{}$ - heißt aktuelle Netzwerkgewichte.

Am Ende des Trainings, wenn die Gewichte ausreichend eingestellt sind, entsprechen sie dann die ideale Netzwerk Gewichte. Signal \hat{w}_{NN} wird zur Kompensation benutzt, und $\hat{\tau}$ ist die Approximation von τ . In jedem Simulationsschritt wird die Totezone von Emulator-

Netzwerk identifiziert, der Fehler am Ausgang wird zurückpropagiert und benutzt wird zur Anpassung der Gewichte des Kompensator-Netzwerkes.

4. Realisierung des Konzepts am Beispiel

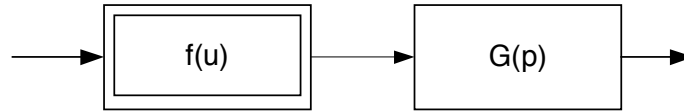


Abbildung 4.1: Beispielsystem

Als Beispiel wird das System in Abb. 4.1 betrachtet, bei dem das Stellglied eine Totzone-Charakteristik Gl. (11) besitzt und die lineare Dynamik durch Gleichung (10) beschrieben wird.

$$G(p) = \frac{10}{p^2 + p + 1} \quad (10)$$

$$f(u) = D[u] = \begin{cases} h(u) = \sin\left(\frac{2\pi}{10}\right), & u > d_- \\ 0, & -d_- \leq u \leq d_+ \\ g(u) = \cos\left(\frac{2\pi}{100}\right), & u < -d_+ \end{cases} \quad (11)$$

Das System wird mit einer Abtastzeit von 0.025 Sekunden diskretisiert. Ein PID-Regler wird nach dem Ziegler und Nichols-Verfahren zur Regelung eingestellt ($K_p = 0.7, K_I = 0.1$ und $K_D = 39.2$). Die Ergebnisse der Simulation sind in Abb. 4.2 (a) dargestellt. Es ist deutlich zu sehen, dass die Totezone eine relativ große Überschwingweite und Überschwingzeit hervorruft. Die zwei Kennwerte erhöhen sich mit der Vergrößerung der Totzonenbreite Abb. 4.2 (b).

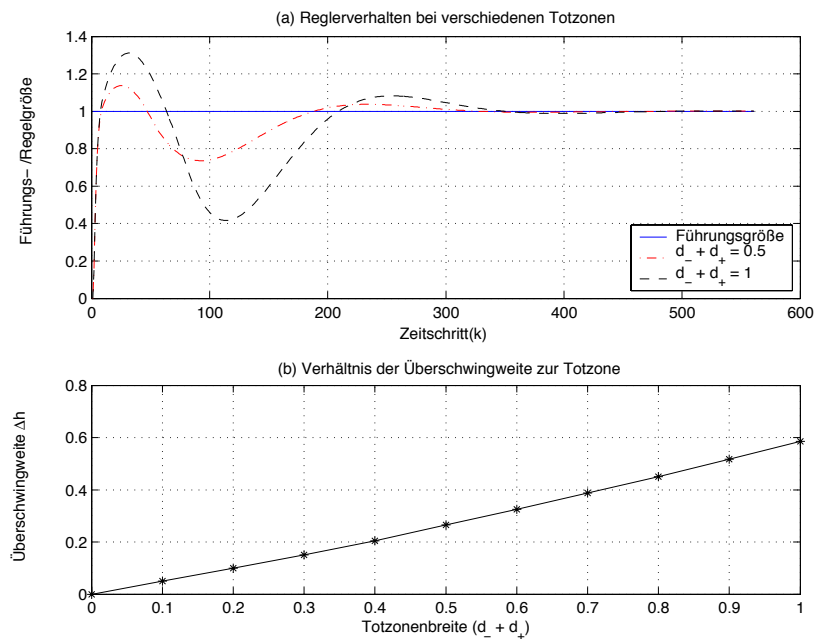


Abbildung 4.2: Führungsverhalten des Regelungssystems ohne Kompensation der statischen Nichtlinearität

Mit Hilfe von neuronalen Netzen wird die Nichtlinearität kompensiert. Das neuronale Netzwerk lernt das Verhalten der Inversen der Totzone. Bei bekannten Totzonen kann dieses Lernen offline erfolgen und bei unbekanntem Totzonen wird das Netzwerk online trainiert. Die Ergebnisse der Simulationen, bei denen die Totzone ($d_- + d_+ = 1.0$) durch neuronale Netzwerke kompensiert werden, sind in Abb. 4.3 dargestellt. Die Ergebnisse sind zufriedenstellend. Es gibt kaum noch Überschwingen.

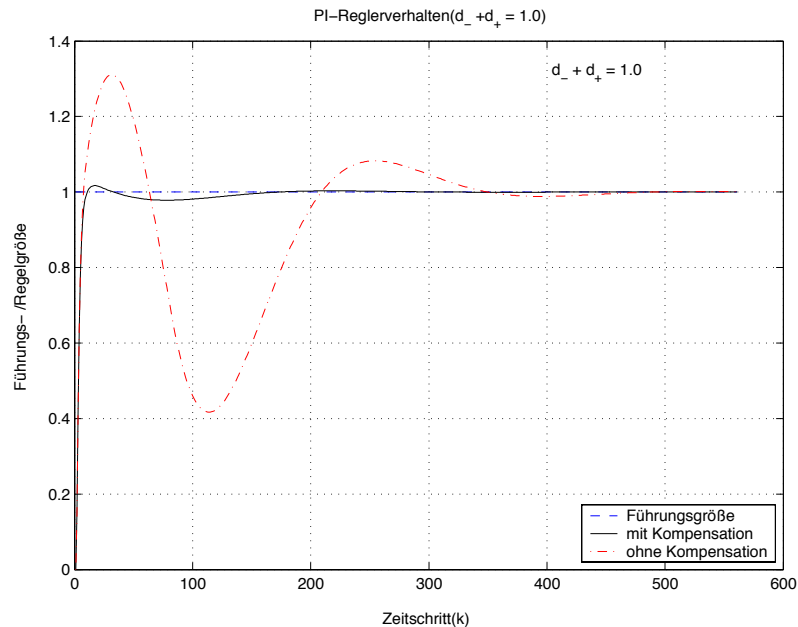


Abbildung 4.3: Regelverhalten bei Kompensation der Totzone mit modifizierten MLP-Netze

5. Zusammenfassung

Von der Simulation, ist es offensichtlich, dass der hier vorgestellte neuronale Totzone-Kompensator eine effektive Weg zur Kompensation von Totzone-Nichtlinearitäten aller Art, ohne Restriktionen am Totzonenmodell selbst, präsentiert. Mehrere Simulationen mit verschiedenen stückweisen stetigen Funktionen wurde durchgeführt und es wurde immer bessere Ergebnisse mit dem modifizierten MLP erzielt im Gegensatz zum normalen MLP mit kontinuierlichen Aktivierungsfunktionen. Es konnte auch im Beispiel gezeigt werden, dass ein PD-Regler in Kombination mit diesem vorgestellten System zur Kompensation von stückweise stetigen Funktionen einwandfrei ein System mit Totzone steuern konnte. Dieses System kann ohne Schwierigkeiten auf andere ähnlichen Systemen angewendet werden. Mit einem solchen System ist man gerüstet für komplexeren Aufgaben, da man Wahlweise, je nach Kontinuitätseigenschaften des zu untersuchenden Systems, das modifizierte oder normale MLP anwenden kann.

6. Literatur

- [1] C. A Detoer and S:M. Shahruz, „Stability of dithered nonlinear Systems with backlas or hysteresis“, Int. J. Control, Vol. 43, No.4, pp. 1045-1060,1986.
- [2] G. Cybenko, „Approximation by Supperpositions of a sigmoidal function“ Math. Contr. Signals Syst, Vol., Vol.2. No.4, pp. 303-314, 1989

- [3] J.H. Kim, J.H. Park, S.W, Lee and E.K.P, Chong, „Fuzzy precompensation of PD controllers for Systems with deadzones“, J. Int. Fuzzy Syst. Vol.1, pp. 125-133, 1993
- [4] S.W. Lee and J.H.Kim, „Control of Systems with deadzones using neural-network based learning control“, Proc. IEEE Int. Conf. Neural Networks, June 1994, pp. 2535-2538.
- [5] Recker, P.V. Kokotovic, D.Rhode, and J Winkelman, „Adaptive Control of unknown plants using dynamical neural networks“, IEEE trans Systems, Man and Cybernetics Vol.24, no.3, pp.400-412, 1994.
- [6] Psaltis, D., Sideris, A., Yamamura, A. (1987) : Neural Controllers. In: IEEE International Joint Conference on Neural Networks, Vol. IV (1987), S.551-558.
- [7] Rastko R. Selmic and Frank L. Lewis (2000), "Deadzone compensation in motion control systems using neural networks," IEEE Trans. Automat. Contr., vol. 45, no. 4, pp. 602-613, April 2000.

Computational Intelligence auf Mikrocontrollersystemen am Beispiel eines Fuzzy-Systems zur Optimierung von Biogasanlagen

Steffen Patzwahl, Klaus-Dietrich Kramer, Thomas Nacke*

Hochschule Harz - University of Applied Sciences
Fachbereich Automatisierung und Informatik
Friedrichstraße 57-59, D-38855 Wernigerode
Tel.: +49(0)3943 / 659-317
Fax: +49(0)3943 / 659-107

E-mail: spatzwahl@hs-harz.de, kkramer@hs-harz.de

*Institut für Bioprozess- und Analysenmesstechnik e.V. (iba)
Rosenhof, D-37308 Heiligenstadt
Tel.: +49(0)3606 / 671-163
Fax: +49(0)3606 / 671-200
E-mail: thomas.nacke@iba-heiligenstadt.de

1 Einführung - Computational Intelligence auf Mikrocontrollern

Innerhalb der letzten Jahre konnten Verfahren der Computational Intelligence (CI) neben klassischen Verfahren oder in Ergänzung dieser in der Anwendung zur Modellierung zunehmend an Bedeutung gewinnen. Fuzzy-Control (FC) -Systeme oder Applikationen von Künstlichen Neuronalen Netzen gehören zum Stand der Technik. Gerade zur Lösung von Aufgabenstellungen der Prozessklassifikation, -modellierung oder -regelung in Bereichen der Bioverfahrenstechnik sind CI-Verfahren bedingt durch stark nichtlineare Zusammenhänge der Prozessgrößen häufig in Verbindung mit nicht oder kaum vorhandenem Prozesswissen unverzichtbar geworden. Einige Applikationen sind in [5],[9],[10],[13] und [14] beschrieben. Zum gegenwärtigen Zeitpunkt bleiben viele Aufgabenstellungen, bei denen Verfahren der CI zum Einsatz kommen, mit Ausnahme von Prozessleitsystemen im Großmaßstab, auf PC-basierte Lösungen beschränkt. Die direkte praktische Anwendung, speziell bei niedrig- und mittelkostenintensiven Regelungsaufgaben, ähnlich wie diese beispielsweise durch PID-Regelmodule der klassischen Regelungstechnik möglich ist, konnte mit CI-Strategien bislang nicht oder nur unterverhältnismäßig erreicht werden. An dieser Stelle knüpfen die Arbeiten des hier vorgestellten Beitrages an.

2 Gesamtalgorithmus

Für die Optimierung von Prozessen sind zielgerichtete Prozesseingriffe notwendig. Diese Prozesseingriffe können erfolgen:

- durch den Anlagenbediener (manuelle Fahrweise des Prozesses),
- durch reglerbasierte Stellglieder (automatische Fahrweise des Prozesses) oder
- mit Hilfe eines den Anlagenbediener unterstützenden Monitoringsystems (manuelle bzw. teilautomatische Fahrweise des Prozesses).

Prozessmonitore und Prozessregler benötigen Modelle, die das Prozessverhalten oder das gewünschte Reglerverhalten beschreiben. Die Erstellung der Modell- oder Regler-einheiten kann gemäß dem in Bild 1 gezeigten Gesamtalgorithmus erfolgen. Die Basis des Gesamtalgorithmus bilden *on-line* erfasste Prozessmessgrößen und ggf. mitprotokollierte Prozesseingriffe eines Anlagenfahrers. Nach der Datenverarbeitung müssen die Zeitreihen der *on-line* Prozessdaten, in Verbindung mit Expertenwissen über den Prozess ausgewertet werden (Prozessanalyse [2] [18]). Den Abschluss der Data-Mining-Phase bildet nach der Datenvorverarbeitung und Datenverarbeitung die Phase der Merkmalsbildung. Merkmale sind im Allgemeinen numerische Kennzahlen, welche die Zeitreihen der *on-line* Prozessgrößen möglichst gut charakterisieren. I.d.R. sind Merkmale numerische Kenngrößen, die mit Methoden der digitalen Signalverarbeitung und der Zeitreihenanalyse aus den Messdaten gewonnen werden [4].

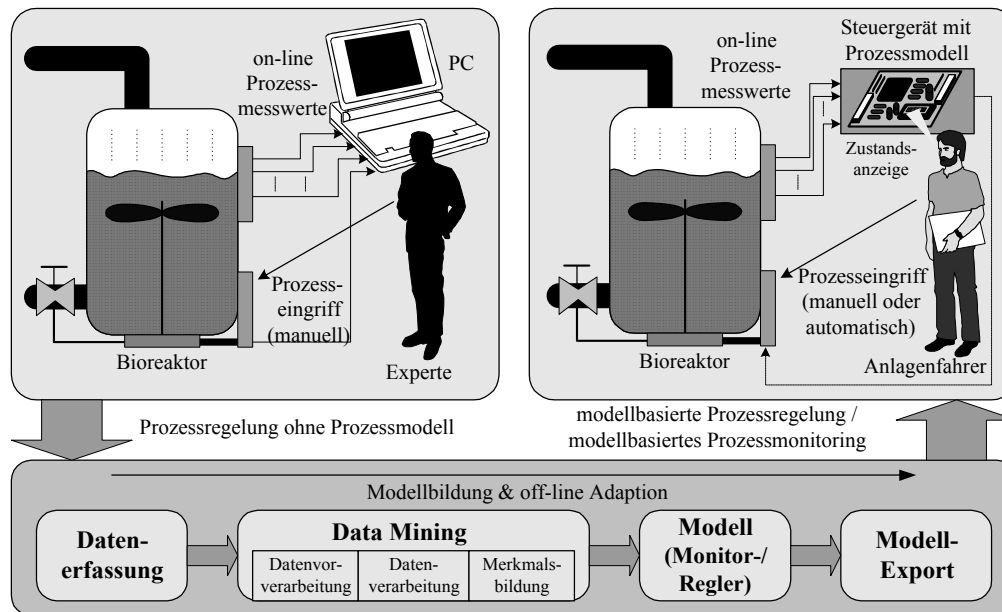


Bild 1: Gesamtalgorithmus zur Erstellung eines Modells (Prozessmodell / -regler)

Die zentrale Zielstellung des Gesamtalgorithmus bildet die Erstellung eines Modells (Prozessmodell oder Modell für den Prozessregler). Dabei ist die Modellierungsart von entscheidender Bedeutung. In Abhängigkeit von der Zielstellung (Modell mit analoger Ausgangsgröße oder Modell mit diskreter Ausgangsgröße (Klassifikator)) existieren eine ganze Reihe von Modellbildungsmöglichkeiten. Klassische Modellierungsverfahren arbeiten z.B. mit Differentialgleichungen oder statistischen Klassifikatoren, währenddessen Modelle basierend auf CI, Künstliche Neuronale Netze und/oder Fuzzy-Logik nutzen. Häufig entsteht gerade durch die Kombination von klassischen und CI-basierenden Modellen die gewünschte Lösung des Prozessmodells oder der Regelstrategie. Nach einer Verifikation der Modelle auf einem PC ist der den Modellen zu Grunde liegende Algorithmus in Verbindung mit den Vorschriften des Data-Mining auf ein Zielhardwaresystem zu übertragen. Das Zielhardwaresystem soll direkt am Prozess eingesetzt werden, um diesen modellbasiert zu regeln und/oder zu überwachen. Um Forderungen von Anlagenbetreibern entgegen zukommen, wurde für den hier beschriebenen Gesamtalgorithmus ein Mikrocontrollersystem als Zielplattform ausgewählt.

3 Softwarewerkzeug zur Umsetzung des Gesamtalgorithmus

Speziell für die Bildung von Modellen (Prozessmodelle, Reglermodelle) wurde das Tool „Model-Builder“ als Teilanwendung des Softwaresystems „EmSoft“ (Embedded Softcomputing) zur Realisierung des Gesamtalgorithmus erstellt (Bild 2). Der Model-Builder vereinigt eine Reihe an Modellierungsmethoden und stellt diese dem Anwender in Form einer grafischen Entwicklungsoberfläche zur Verfügung. Bei der Implementierung der Methoden für die Modellbildung wurden schwerpunktmäßig verschiedene Verfahren der Computational Intelligence (CI) berücksichtigt:

- Fuzzy und Fuzzy-Control [6] [19],
- Neuronale Netze in Form von Multi-Layer-Perceptrons [12] sowie
- Neuronale Klassifikatoren in Form von Selbstorganisierenden Netzen und Fuzzy-Kohonen-Konzepten [1] [7] [20].

Zusätzlich stehen Klassifikatoren auf Basis von Distanzmaßen nach Euklid, nach Mahalanobis und mit Hilfe des Bayes-Verfahrens zur Verfügung [3] [8].

Jedes Modell kann entsprechend der Anforderungen parametrisiert und datenbasiert getestet werden. Durch die leichte und schnelle Modellerstellung und die simulative Verifizierung ist der Nutzer in der Lage, sehr schnell verschiedene Modellierungsansätze für die Aufgabenstellung zu realisieren. Insbesondere eine leichte Gegenüberstellbarkeit der entwickelten Modelle erleichtert das Finden einer optimalen Modellierungsart erheblich. Gleichfalls findet die Realisierung von Reglerstrukturen Berücksichtigung.

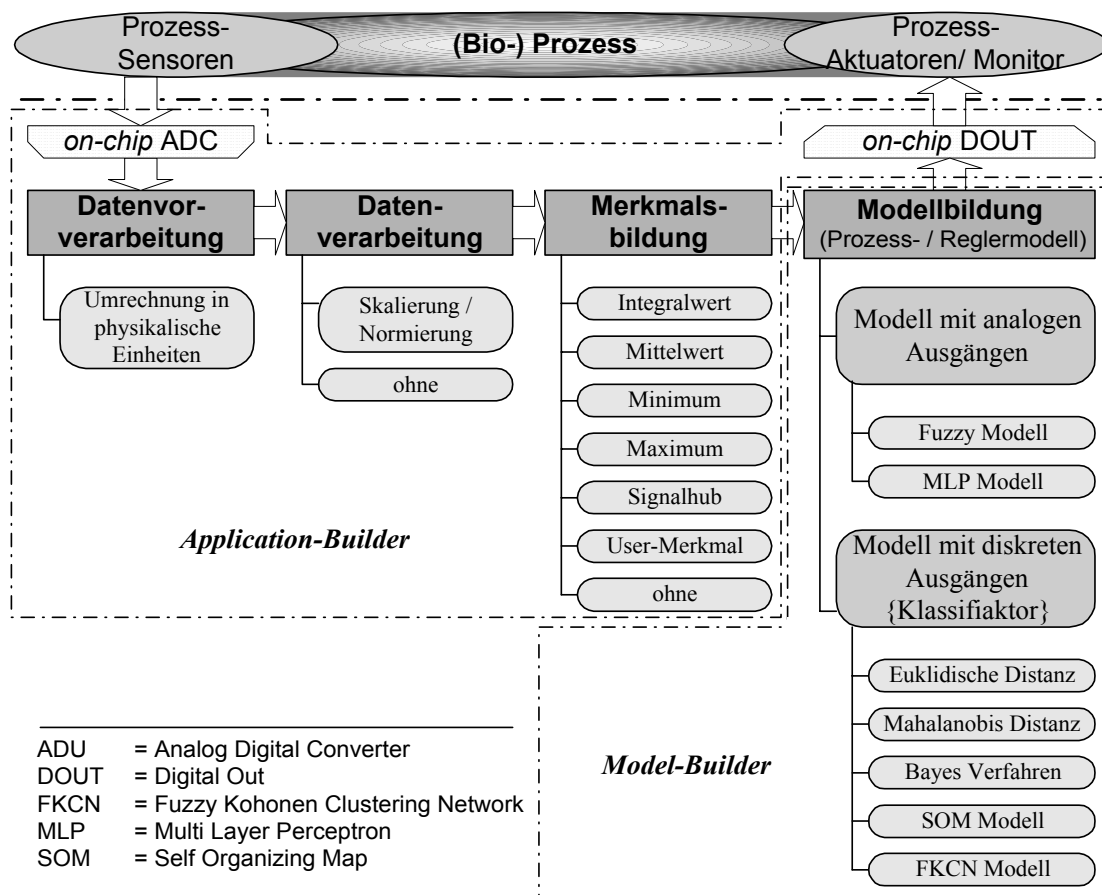


Bild 2: Struktur und Teilalgorithmen des Softwarewerkzeugs „EmSoft“

Die zweite Teilanwendung des Softwaresystems bildet das Tool „Application-Builder“. Aufgabe dieses Softwarewerkzeuges ist die Einbindung des erstellten Modells in den unter Punkt 2 beschriebenen Gesamtalgorithmus. Wesentlich ist dabei das Data-Mining [17] [20]. In vielen Fällen, gerade auch bei Bioprozessen, lassen sich optimale Modell-eingangsgrößen erst durch eine vorgeschaltete Merkmalsbildung ermitteln. Aus diesem Grund sind einfache Verfahren zur Merkmalsextraktion aus Zeitdatenreihen in das Softwarewerkzeug implementiert worden. Um das Data-Mining zu vervollständigen stehen auch Datenvor- und Datenverarbeitungsfunktionen zur Verfügung. Weiterhin wird die *on-chip* Peripherie der Zielhardware, insbesondere die Einheiten der Analog-Digital-Umformung (ADC), der digitalen Ausgabe (DOOUT) und der Timer mit dem Application-Builder parametrisiert.

Der Gesamtalgorithmus schließt mit der Überführung der Modell-/Reglerapplikation in einen Run-Time (RT) -C-Code. Das Entwicklungstool EmSoft generiert einen RT-C-Code, welcher alle Aspekte des Gesamtalgorithmus in Form von Programm-Modulen enthält. Auch die Modellbildungsstrategien werden in den C-Code eingebunden. Die Gesamtlösung lässt sich nach einer Compilierung (C-Compiler für Zielhardware) direkt auf das Zielhardwaresystem übertragen. Damit kann das Mikrocontroller (μC) -System mit der entsprechenden Firmware PC-unabhängig direkt am Prozess eingesetzt werden. Das μC -System erzielt wesentliche Vorteile in Hinblick auf Robustheit und Preisgünstigkeit gegenüber PC-basierten Lösungen.

4 Biogasanlagen

4.1 Biogaserzeugung und regelungstechnische Probleme

Die Erzeugung von Biogas aus organischen Reststoffen ist besonders bei der anaeroben Co-Fermentation komplex und geschieht über eine Vielzahl von Zwischenprodukten. Die Substratzusammensetzung ist nicht konstant und unterliegt stofflichen und jahreszeitlichen Schwankungen. Um Biogasanlagen ökonomisch zur Energiegewinnung einsetzen zu können, muss eine möglichst kontinuierliche Gasausbeute bei optimalem Kohlenstoffabbau erzielt werden. Zur Erfüllung dieser Forderung ist die Menge des zugeführten Substrates in Abhängigkeit der gewünschten Gasertragsmenge zu regeln. Die konventionelle Regelungstechnik stößt bei der Steuerung solcher anaeroben Fermentationsprozesse zur Biogaserzeugung an ihre technischen Grenzen. Diese ergeben sich durch die nichtlinearen Zusammenhänge zwischen Substratzuführung und Biogasgewinnung, die mit großen Zeitkonstanten verbunden sind. Außerdem werden diese regelungstechnischen Probleme bei der anaeroben Fermentation durch die auftragsbedingte Co-Fermentation unterschiedlichster organischer Abfälle verstärkt. Dies führt wiederum zu unterschiedlichen Raumbelastungen der Fermentationsanlage. Der Prozess der anaeroben Fermentation ist relativ komplex und aufgrund der Nichtlinearitäten zwischen Substratzuführung, dem Biomassewachstum und der Gasausbeute mathematisch äußerst schwer und nur modellhaft zu beschreiben [10] [16].

4.2 Stand der Technik bei der Anlagenführung

Die modernen Fermentationsprozesse zur anaeroben Biogasgewinnung werden bei großtechnischen Anlagen durch eine große Anzahl von Messungen überwacht und entsprechend gesteuert. Dabei werden als wesentliche Kenngrößen für die Beurteilung des Prozesses die Messung des zeitlichen Verlaufes der Messwerte für den pH-Wert,

des Redox-Potenzials, der Konzentration wichtiger Substrate in Zuführung und Abführung aus dem Biogasreaktor und Metabolite bzw. Produkte des Stoffwechsels oder deren Verknüpfung betrachtet. Gerade die Stoffkonzentrationen sind jedoch messtechnisch nicht oder nur schwer als *on-line* Messgrößen zu erfassen, so dass diese Prozesskennwerte nicht in die Regelung der Anlage einfließen können. Kleine und mittlere Biogasanlagen sind i.d.R. unterinstrumentiert.

Seit mehreren Jahren wird mit verschiedenen mathematischen Modellansätzen versucht, eine geeignete Prozessstrategie zur Prozessoptimierung anaerober Fermentationsabläufe zu entwerfen. Grundlage dieser Modelle bildeten die genaue physikalische und chemische Analyse der Ausgangsmaterialien. Diese wurden in Verbindung mit der genutzten Reaktortechnik zur Bildung kinetischer Modelle mit Wachstums- und Ertragskoeffizienten genutzt. Die Überführung in die industrielle Anwendung erwies sich oft als sehr schwierig, da diese Systeme nur begrenzt auf Schwankungen in der Substratzusammensetzung reagieren konnten [15]. In der Praxis führte dies u.a. zu einer organischen Überlastung der Anlage, oftmals auch verbunden mit einem Prozessversagen. Durch Schaffung von Überkapazitäten, d.h. Nutzung größerer Reaktolvolumina, kann dieser Mangel auf Dauer nicht ausgeglichen werden.

5 Fuzzy-Control-Modell für Biogasanlagen

5.1 Datenerfassung an Labor-Biogasanlagen

Als Grundlage für die Modellbildung wurde eine Datenbasis zur Prozessbeschreibung generiert. Die Labor-Biogasanlage (Fermentervolumen 6 Liter) mit umfangreichem Mess-Equipment zur *on-line* Messdatenerfassung zeigt Bild 4.



Bild 4: Foto der Biogas-Laborpilotanlage (iba – Heiligenstadt)

Nach der Prozess- und Datenanalyse in Verbindung mit Korrelationsanalysen wurden die nachfolgend aufgeführten Messgrößen als prozess- und modellrelevant eingestuft:

- Gasproduktion an Biogas (GP [l])
- Konzentration an CH₄ im entstandenen Gas (CH₄) [Vol%]
- Konzentration an CO₂ im entstandenen Gas (CO₂) [Vol%]
- Konzentration an O₂ im entstandenen Gas (O₂) [Vol%]
- pH-Wert (pH) [-]

5.2 Data-Mining und Modellbildung

Das mit Hilfe des Model-Builders und Application-Builders erarbeitete Fuzzy-Control (FC) -Modell [10][16] zeigt Bild 5 im Gesamtkonzept der Anlage. Dargestellt ist die Anwendungsphase des Gesamtalgorithmus (2.) mit dem μ C-System zur automatischen und PC-unabhängigen Anlagenregelung.

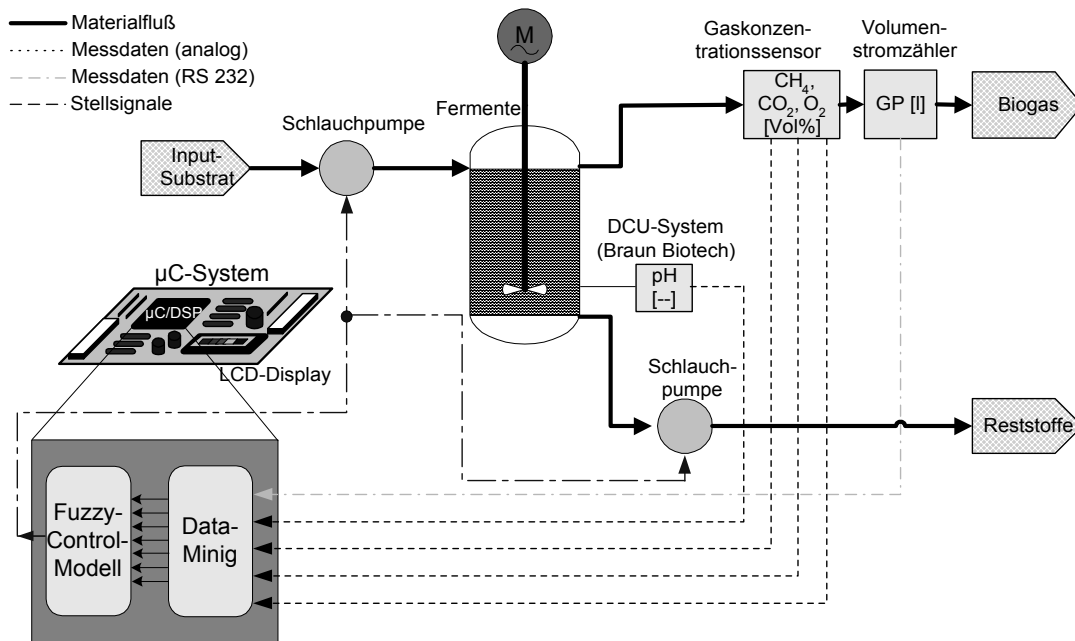


Bild 5: FC-Modell zur Automatisierung von Biogasanlagen im Anlagenkontext

Das FC-Modell arbeitet mit sieben Eingangsgrößen, die durch die Merkmalsbildung des Data-Mining generiert werden:

- Gasproduktionsrate pro Tag bezogen auf das Fermentervolumen GPR [NI/(l*d)]
- CO₄-Gehalt pro Tag (CH₄_d) [%]
- CO₂-Gehalt pro Tag (CO₂_d) [%]
- O₂-Gehalt pro Tag (O₂_d) [%]
- Quotient aus CH₄- und CO₂-Gehalt pro Tag (CH₄/CO₂_d) [-]
- Änderung der GPR zwischen aktuellen Tag und Vortag (dGPR) [NI/(l*d)]
- Arithmetisches Mittel des pH-Wertes (pH_m) [-]

Die optimale Zudosierate (QS) wird als Ausgangsgröße bestimmt und kann direkt zur Prozessregelung als Stellgröße eingesetzt werden. Die Regelbasis (22 Regeln) und die Parameter der Zugehörigkeitsfunktionen der linguistischen Terme wurden mit Prozessexperten aufgestellt (Fuzzy-Experten-System).

Das Modell bezieht sich auf die Substrate Fette und Hydrolyse als Inputmaterial in den Vergärprozess. Durch das FC-Modell wird in erster Linie eine Prozess-Sicherheit gewährleistet, d.h. ein Prozessversagen wird verhindert. Weiterhin optimiert das Modell die Gas- und CH₄-Ausbeute [10][16].

5.3 Mikrocontrollerbasiertes Kompaktgerät

Als Zielhardwaresystem für den Gesamtalgorithmus und damit für das FC-Modell zur Regelung von Biogasanlagen wurde das MiniModul167[®] [11] der Firma phytec Mess-

technik GmbH ausgewählt und eingesetzt. Diese Systemeinheit ist mit dem Mikrocontroller SAB 80C167 der Firma Infineon Technologies AG bestückt. Der Controller weist folgende Leistungsmerkmale auf [11]:

- Verarbeitungsbreite: 16 bit, Taktfrequenz (CPU): 20 MHz
- 16-Bit ALU
- 10-Bit ADC, per Multiplexbetrieb für 16 Kanäle nutzbar
- zwei 16-Bit-Timer/Counter
- parallele Ein- und Ausgangsports (insgesamt 111 Leitungen)
- interner Taktgeber
- zwei serielle Schnittstellen

Der mit Hilfe des Softwarewerkzeuges EmSoft erstellte modellbasierte Regelalgorithmus (für die Biogasanlage der FC-Algorithmus) wurde mit der Codegenerierungsfunktion in einen Run-Time (RT) -C-Code überführt. Der generierte RT-C-Code enthält dabei auch die Programmblöcke für die Steuerung der *on-chip* Peripherie der Zielhardware. Das sind vor allem die ADC, die digitale Portausgabe und die Timerfunktionen. Weiterhin wird das Interruptsystem initialisiert und im Algorithmus verwendet. Der C-Code ist für die Zielhardware optimiert und aufgrund der Hardwarebesonderheiten jedes Zielhardwaresystems nur auf den angegebenen Systemen nach der Compilierung mit einem geeigneten C-Compiler lauffähig.

Speziell für den Einsatz an Biogasanlagen wurde im Rahmen der vorgestellten Forschungsarbeiten ein Kompaktgerät entwickelt und als Prototyp gefertigt. Dieses Kompaktgerät enthält als Mikrocontrollereinheit das MiniModul167[®] der Firma phytec Messtechnik GmbH. Schaltungstechnisch wurden Ergänzungen zur Stromversorgung, zur Pegelanpassung und zur Realisierung weiterer Sonderfunktionen vorgenommen. Weiterhin sind zur Nutzerkommunikation ein LCD-Display und einige Taster installiert. Zur Prozessankopplung verfügt das Gerät über eine Reihe von Schnittstellen. Das Kompaktgerät (Bild 6) kann direkt am Prozess eingesetzt werden um diesen PC-unabhängig zu überwachen und zu regeln.



Bild 6: Mikrocontrollerbasiertes Kompaktgerät (iba Heiligenstadt)

5.4 Testläufe an der Biogas-Laborpilotanlage

Zum Praxistest des FC-Modells wurden die Algorithmen des Data-Mining sowie das FC-Modell in Form des generierten RT-C-Codes in den Flash-Speicher des Kompaktgerätes übertragen. Das Kompaktgerät wurde über die Prozessschnittstellen mit den Anlagensensoren und der Substratzuführungspumpe verbunden. Zur Kontrolle des Systems wurde außerdem der Mess-PC parallel eingesetzt um die Mess- und Stell-signale zu protokollieren.

Bild 8 zeigt die Ergebnisse des Testlaufes über den Zeitraum von 25 Tagen, wobei die oberen sieben Graphen den Verlauf der Prozessmerkmale wiedergeben. Jedes Prozess-

merkmal stellt einen Tageswert dar und repräsentiert 144 *on-line* Messwerte (Abtastrate der ADC = 10 min). In der unteren Graphenreihe ist die Zudosieraten (Substratzufuhr) dargestellt. Bild 7 visualisiert die Struktur des Systemtests.

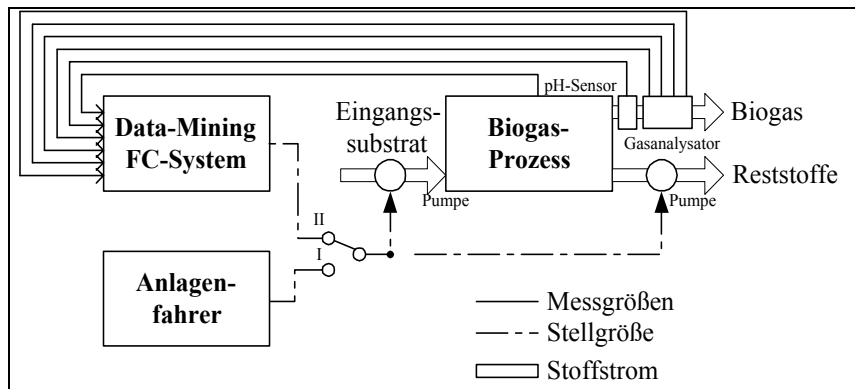


Bild 7: Struktur des Systemtests

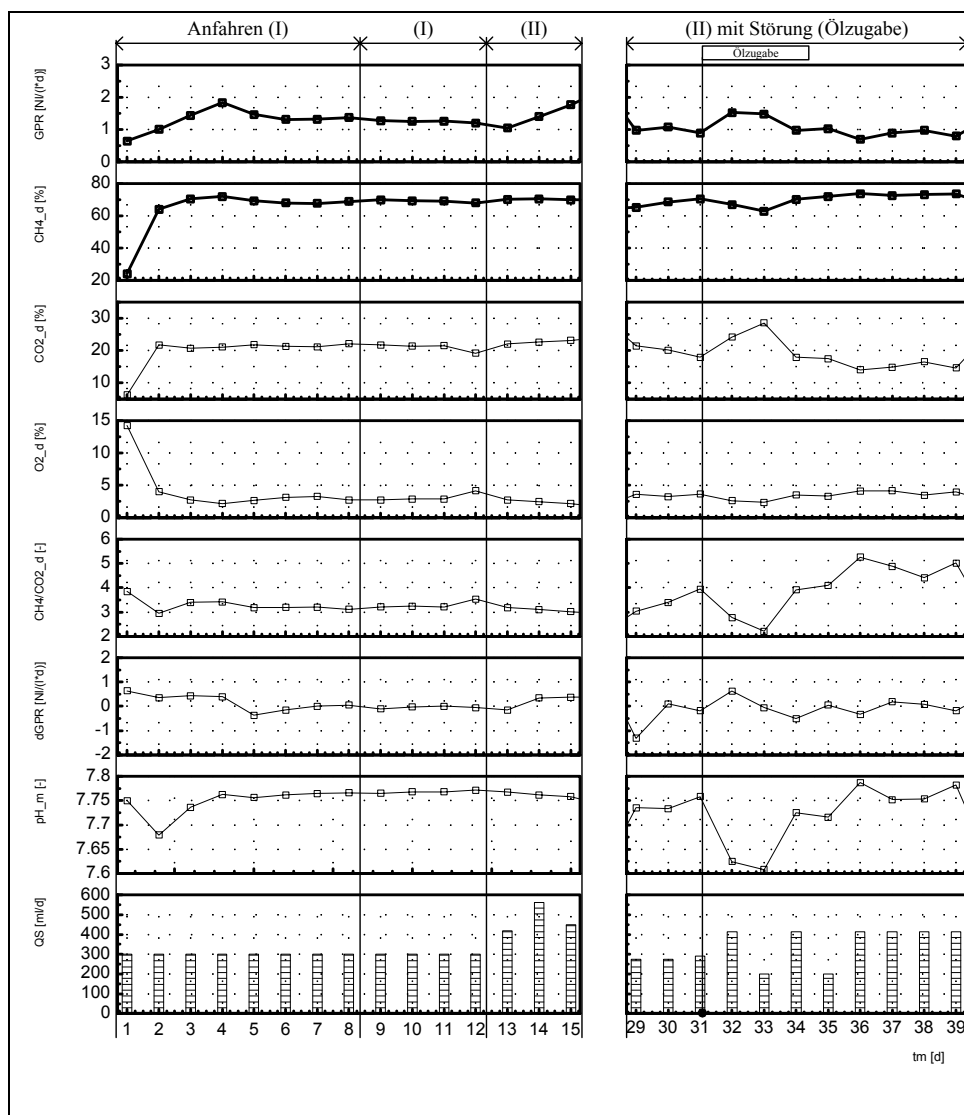


Bild 8: Testergebnisse (Verlauf der Prozessmerkmale und der Stellgröße)

Während der ersten 12 Tage (Anfahrphase des Prozesses) wurde der Fermenter mit einer konstanten Zudosieraten an Substrat (QS) von 300 ml/d beschickt. Die Steuerung

der Beschickung wurde dabei manuell vom Anlagenfahrer ausgeführt (Schalterstellung I – Systemstruktur nach Bild 7). Für die Tage 13 bis 15 übernahm das Kompaktgerät die Steuerung der Zudosierung automatisch (Schalterstellung II – Systemstruktur nach Bild 7).

Die rechte Seite in Bild 8 enthält einen weiteren Praxistest, beim dem der Prozess gezielt gestört wurde. Eine Prozessstörung der Biogasfermentation kann durch einen Eingriff in die Substratzufuhr oder durch Änderung der Substrateigenschaften vollzogen werden. Da die Substratzufuhr als Zielgröße geregelt wird, kam für die Systemstörung nur eine Beeinflussung der Substrateigenschaften in Frage. Im Testfall wurde dies durch die Zugabe von Öl erreicht. Öl ist bezogen auf die anaerobe Fermentation sehr energiereich, da es einen hohen Fettgehalt aufweist. Dadurch ist es möglich mit einer relativ kleinen zusätzlichen Zudosiermenge die kritische Beladungsrate des Fermenters zu erreichen. Eine Ölzugabe oberhalb der kritischen Beladungsrate beeinflusst die anaerobe Fermentation negativ, die Reaktion versäuert und die Prozessstabilität wird erheblich gefährdet.

Die durch die Regelung zu optimierenden Zielgrößen sind der Methangehalt pro Tag CH_4_d [%] und die Gasproduktionsrate GPR [$l/(l*d)$]. Die weiteren in Bild 8 gezeigten Merkmale stellen Indikatoren für den Prozesszustand dar.

Die Auswertung ergibt, dass der Prozess optimal vom Kompaktgerät geführt wurde. Dies ist erkennbar an:

- einer relativ konstanten und hohen CH_4 -Konzentration im produzierten Biogas (CH_4_d [%]),
- einer Gasproduktionsrate (GPR) von größer $0,9$ $l/(l*d)$,
- einem pH-Wert (pH_m [--] im Prozessoptimum und
- einem niedrigen Koeffizienten aus CH_4 - und CO_2 - Konzentration (CH_4/CO_2_d [--]) als Indiz für die Prozesssicherheit.

6 Zusammenfassung

Dieser Beitrag stellte einen Gesamtalgorithmus zur Realisierung modellbasierter Regelungs- und Monitoringstrategien unter Verwendung von Strategien der Computational Intelligence (CI) auf Mikrocontrollersystemen vor. Dabei wurden alle Teilaspekte ausgehend von den Prozesssignalen, über das Data-Mining bis zum Prozessmodell bzw. dem Regelalgorithmus mit der Signalausgabe beleuchtet. Einen weiteren Schwerpunkt bildete die Überführung des am PC unter Einsatz der beschriebenen Softwaretools und unter Beachtung der Designschritte erstellten Monitoring- oder Regelalgorithmus in einen Run-Time (RT) -C-Code für die Zielhardware. Der als Resultat des Gesamtalgorithmus entstandene RT-C-Code bildet auf dem Zielhardwaresystem mit μC eine eigene Applikation, die PC-unabhängig zum Prozessmonitoring oder zur Anlagenregelung eingesetzt werden kann.

Am Beispiel der anaeroben Fermentation von Ausgangssubstraten zu Biogas wurde gezeigt, dass ein mikrocontrollerbasiertes Steuergerät, auf dem alle Teilschritte des Gesamtalgorithmus in Form eines RT-C-Codes portiert wurden, in der Lage ist, die Prozessführung zu automatisieren und dabei die Prozesseffizienz aber auch die Prozesssicherheit zu erhöhen.

Die Arbeiten des vorgestellten Themas wurden im Rahmen eines Forschungsprojektes von der Arbeitsgemeinschaft industrieller Forschungsvereinigungen „Otto von

Guericke“ e.V. (aif) – Projektträger des BMBF im Rahmen des Programms aFuE gefördert (Förderkennzeichen: 17.113.01).

7 Weiterführende Arbeiten

Zum gegenwärtigen Zeitpunkt finden weitere Versuche an Labor-Pilotanlagen statt um das System weiter zu verifizieren und zu optimieren. Dabei werden auch weitere Substrate eingesetzt um die Modelle an die unterschiedlichen Substratarten anzupassen.

Außerdem wird an weiteren Modellen zur Optimierung des Biogasprozesses gearbeitet. Wichtige Systeme dabei werden ein Modell zur Vorhersage der Gasproduktionsrate für den nächsten Zyklus und ein Prozessmonitor zur *on-line* Klassifikation des Prozesszustandes bilden.

Ein weiterer Punkt liegt in der Übertragung der Ergebnisse der Labor-Pilotanlagen auf reale Biogasanlagen. Hauptziel sind dabei kleine und mittlere Biogasanlagen, überwiegend im ländlichen Bereich, bei denen die Vorteile eines Mikrocontrollersystems zur Anlagenüberwachung und / oder -regelung ausgenutzt werden können.

8 Literatur

- [1] Bezdek, J., Pal, S. K.: *Fuzzy Models for Pattern Recognition*. IEEE Press, New York; 1992
- [2] Bocklisch, S. F.: *Prozessanalyse mit unscharfen Verfahren*. Verlag Technik, Berlin; 1987
- [3] Canty, M. J.: *Fernerkennung mit neuronalen Netzen – Theorie – Algorithmen – Programme*. Expert-Verlag, Renningen-Malmsheim, S. 122-125; 1999
- [4] Gomersall, R.: *Entwicklung von Expertensystemen zur Prozessüberwachung mit der Entwicklungsumgebung G2 unter besonderer Berücksichtigung der Wissensakquisition und – implementierung*. Dissertation, Universität Hannover, Hannover; 1998
- [5] Isermann, R.: *Zur Anwendung der Fuzzy-Logik in der Regelungstechnik*. Automatisierungstechnische Praxis 38, Oldenbourg Verlag, München, Wien, S. 24-26; 1996
- [6] Koch, M., Kuhn, T., Wernstedt, J.: *Fuzzy Control – Optimale Nachbildung und Entwurf optimaler Entscheidungen*. Oldenbourg-Verlag, München, Wien; 1996
- [7] Kohonen, T.: *Self-organizing and associative memory*. Springer Verlag, Heidelberg, Berlin, New York; 1989
- [8] Mahalanobis, P. C.: *On the Generalized Distance on Statistic*. Proc. Nat. Inst. India, No. 2, p. 49-55; 1936
- [9] Patzwahl, S., Nacke, T., Kramer, K.-D.: *Selforganizing neural networks to supervise the content of water in biological filtration plants*. 7th European Congress on Intelligent Techniques and Soft Computing, Proceedings of EUFIT '99 Aachen, Germany, Final Programme, Abstracts of the Papers & Proceedings on CD-ROM, Aachen, 1. Edition, Wissenschaftsverlag, Aachen, S. 104 ff.; 1999
- [10] Patzwahl, S., Nacke, T., Frense, D., Beckmann, D., Vollmer, R., Tautz, T., Kramer, K.-D.: *Microcontroller-Based Fuzzy System to Optimize the Anaerobic*

- Digestion in Biogas Reactors*. Computational Intelligence, International Conference, 7th Fuzzy Days, Dortmund, Germany, October 1-3, 2001, Proceedings, Springer-Verlag, London, Heidelberg, Berlin, p. 2-10; 2001
- [11] Phytec: *miniMODUL-167, Hardwaremanual*. phytec Messtechnik GmbH, Mainz; 1999
- [12] Schmitter, E.-D.: *Neuronale Netze - Einführung, Programmierbeispiele, praktische Anwendungen*. 1. Auflage, W. Hofacker Verlag, Holzkirchen; 1991
- [13] Souza De, M. B., Ferreira, L. S., Folly, R. O. M.: *Development of an alcohol fermentation control system based on biosensor measurements interpreted by neural networks*. Sensors and Actuators B 3713, p.1-6; 2001
- [14] Stenz, R., Kuhn, J.: *Vergleich: Fuzzy-Automatisierung und konventionelle Automatisierung einer Batch-Destillationskolonne*. Automatisierungstechnische Praxis 35, Oldenbourg Verlag, München, Wien, S. 228-295; 1993
- [15] Steyer, J.-P., Buffière, P., Rolland, D. and Molletta R.: *Advanced control of anaerobic digestion process through disturbances monitoring*. Wat. Res. Vol. 33, No. 9, Elsevier Science Ltd, p. 2059 – 2068; 1999
- [16] Tautz, T.: *Entwicklung und praktische Realisierung eines mikrocontrollerbasierten CI-Regelungssystems für Biogasanlagen*. Diplomarbeit, Hochschule Harz, Wernigerode; 2001
- [17] Wang, X. Z.: *Data mining and knowledge discovery for process monitoring and control*. London, Springer-Verlag, Heidelberg, Berlin; 1999
- [18] Wernstedt, J.: *Experimentelle Prozessanalyse*. Verlag Technik, Berlin; 1989
- [19] Zadeh, L.: *Fuzzy Sets. Information and Control*. 8, S.338-353; 1965
- [20] Zimmermann, H.-J.: *Datenanalyse – Anwendung von DataEngine mit Fuzzy-Technologien und Neuronalen Netzen*. VDI-Verlag, Düsseldorf; 1995