64-544
Grundlagen der Signalverarbeitung und Robotik

http://tams.informatik.uni-hamburg.de/lectures/2012ss/vorlesung/GdSR

Jianwei Zhang

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik
Technische Aspekte Multimodaler Systeme

Sommersemester 2012

Gliederung

1. Einführung
2. Grundlagen der Robotik
3. Elementares der Sensorik
4. Verarbeitung von Scandaten
5. Rekursive Zustandsschätzung
6. Fuzzy-Logik
7. Steuerungsarchitekturen

Agenda

6. Fuzzy-Logik
- Adaptive Methoden zur Regelung
- Fuzzy-Regelung
- Charakteristische Funktion / Zugehörigkeitsfunktion
- Fuzzy-Menge
- Linguistische Variablen und Terme
- Darstellungen der Zugehörigkeitsfunktionen
- Fuzzy-Regelung
- Literatur

Einführung in die Fuzzy-Regelung

„The way we have to describe Nature is generally incomprehensive to us“

Richard P. Feynman

„It should be possible to explain the laws of physics to a barmaid“

Albert Einstein

Einführung in die Fuzzy-Regelung

- Lotfi A. Zadeh Begründer der Theorie der unscharfen Mengen (Fuzzy Sets, 1965)
  - Professor an der Universität Berkeley, Kalifornien seit 1959
  - Systemtheorie, Entscheidungstheorie, Informationssysteme
- Durchbruch der Fuzzy-Set-Theorie seit 1980er Jahren
  - insbesondere Japan
  - Beispiel: U-Bahn in Sendai (1987)
    Steuerung der Anfahr- und Bremskurven
- präzise Erfassung des Unpräzisen:
  nicht durch durch Objekte (Elemente der Menge) definiert, sondern über den Grad der Zugehörigkeit zur Menge
- wichtiges Anwendungsfeld: Regelungstechnik

Adaptive Methoden zur Regelung

- Regelung kann aufgefasst werden als Abbildung von einem Sensorraum auf Aktionen
- klassischer Lösungsansatz: Bestimmung einer Kontrollfunktion $\varphi : X_1 \times X_2 \times \ldots \times X_n \longrightarrow Y$ mit $(x_1, x_2, \ldots, x_n) \longmapsto Y$; $(x_1, x_2, \ldots, x_n)$ – Messwerte, Y – Regelgröße
- es wird der Prozess modelliert
- physikalische Kenntnisse über den Prozeß werden benötigt
- in vielen Fällen ist *a priori* nicht bekannt, welche Messgrößen besonders wichtig für die Auswahl von Aktionen sind
- manche Systeme sind nur sehr schwer mathematisch zu beschreiben

Adaptive Methoden zur Regelung (cont.)

- oft sind die Sensordaten ungenau, verrauscht und/oder hochdimensional
- Das Erstellen einer optimalen Abbildung zwischen Sonsorraum und Aktionen ist dann mit klassischen regelungstechnischen Methoden sehr schwierig.

  Alternative:
- ein Mensch kann, obwohl er kein Wissen über Differentialgleichungen hat, trotzdem gehen

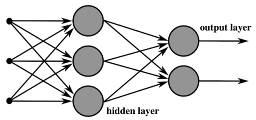
Adaptive Methoden zur Regelung (cont.)

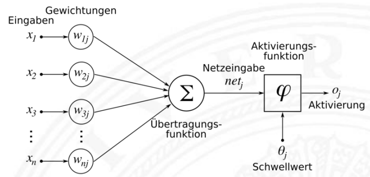
- ein Mensch kann ohne das Wissen über Differentialgleichungen gehen
  - er kann das Gleichgewicht halten, ohne zu wissen, wie der Prozess mathematisch modelliert wird
  - Idee: statt den Prozess selbst zu modellieren, soll das Verhalten eines Experten, der den Prozess regelt, modelliert und simuliert werden
- Erstellen eines Modells für das Verhalten eines menschlichen „Regelungsexperten“ heißt kognitive Analyse
- Experte formuliert sein Wissen in Form linguistischer Regeln
- es wird also eine einfachere Methode zur Beschreibung benutzt, und/oder die Regelung passt sich an die Bedingungen an

Adaptive Methoden zur Regelung (cont.)

Modelle für Adaptive Regler:

- Neuronale Netze





  Netz lernt durch:
  - Veränderung der Gewichte $w_{i,j}$
  - Anpassung der Schwellwerte θ_j
  - Anlegen neuer, bzw. Löschen alter Verbindungen
  - Anlegen neuer, bzw. Löschen alter Neuronen
- Fuzzy-Controller
  - *klassischer Regler* (Mamdani-Typ)
  - *Funktionsapproximator* (TSK-Modell oder B-Spline-Modell)

Lernverfahren

- Überwachtes Lernen
- Reinforcement-Lernen
- Unüberwachtes-Lernen

Mehr zu Lernverfahren in der Vorlesung
„*Maschinelles Lernen*“!

Vergleich: menschliches Gehirn und Computer

Kriterium	Gehirn	Computer
Parallelität	hoch	niedrig
Präzision	mäßig	hoch
Fehlertoleranz	hoch	niedrig
Speicherzugriff	global	lokal
numerische, präzise Berechnungen	schlecht	gut
Erkennung von Mustern	gut	schlecht
fehlerloses Speichern von Daten	schlecht	gut
Selbstorganisation	ja	bisher nicht
Verallgemeinern von Beispielen	gut	schlecht

Fuzzy-Regelung

- ungenaue natürlichsprachliche Abstufungen von Begriffen wie „**groß**“, „**schön**“, „**stark**“, „**jung**“, „**alt**“ ...
- menschliche Denk- und Verhaltensmodelle mit der einstufigen Logik
  - **Autofahren:** *wenn-dann*-Regeln
  - **Autoparken:** Genau auf den Millimeter? (Unschärfe gegeben)
- unscharfe Sprache statt numerischer Beschreibung
  - „Bremse 2.52 m vor der Kurve!“ $\rightarrow$ nur in Maschinensystemen
  - „Bremse kurz vor der Kurve!“ $\rightarrow$ in natürlicher Sprache

Fuzzy-Regelung (cont.)

- *Fuzzy* bedeutet: unscharf, vage, verschwommen, unklar, ...
- Fuzzy-Regelung benutzt Fuzzy-Menge/Fuzzy-Logik als Mechanismus für
  - Behandlung von Problemen, die nicht einfach mit *ja* oder *nein* beantwortet werden können
  - Modellierung von (*soft*) Konzepten ohne scharfe Grenzen
  - Abstraktion von unnötigen/zu komplexen Details
  - „Computing with words“

Charakteristische Funktion

Scharfe Mengen (analog zur klassischen Mengenlehre) lassen sich definieren durch Angabe ihrer charakteristischen Funktion:

$$\mu_{\mathcal{A}}(x) = \begin{cases} 1 & \text{für } x \in \mathcal{A} \\ 0 & \text{für } x \notin \mathcal{A}, \end{cases}$$

Der Zugehörigkeitsgrad $\mu_{\mathcal{A}}(x)$ eines Elementes x zu einer Menge $\mathcal{A}$ aus einem Universum X wird also hier beschrieben durch die Funktion: $\mu_{\mathcal{A}} : X \to \{0, 1\}$.

Zugehörigkeitsfunktion

Für **Fuzzy-Mengen** A verwendet man eine verallgemeinerte charakteristische Funktion μ_A, die jedem Element $x \in X$ eine reelle Zahl aus $[0, 1]$ zuordnet:

$$\mu_A : X \to [0, 1]$$

- Die Funktion μ_A wird als Zugehörigkeitsfunktion (ZF) bezeichnet
- Sie gibt den „Grad“ an, zu dem das Element x zur beschriebenen unscharfen Menge A gehört

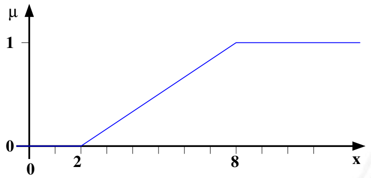
Fuzzy-Menge

Eine **Fuzzy-Menge** A über einem Universum X ist gegeben durch eine Abbildung $\mu_A : X \to [0, 1]$. Für alle $x \in X$ bezeichnet $\mu_A(x)$ den Grad der Zugehörigkeit (des Enthaltenseins) von x in A.

- Eine Fuzzy-Menge A über X heißt leer , wenn gilt:
  $$\mu_A(x) = 0 \quad \forall x \in X$$
- Eine Fuzzy-Menge A über X heißt universell , wenn gilt:
  $$\mu_A(x) = 1 \quad \forall x \in X$$
- Scharfe Mengen lassen sich als unscharfe Mengen mit den Zugehörigkeitsgraden 0 und 1 darstellen.
- In der Praxis werden Fuzzy-Menge und Zugehörigeitsfunktion synonym benutzt.

Fuzzy-Menge (cont.)

- Beschreibungsformen der Fuzzy-Mengen:
  - graphische Darstellung durch Vorgabe einer Kennlinie $\mu_A(x)$



  - parametrische Darstellung, die den Verlauf der Kennlinie beschreibt

$$\mu_A(x) = \begin{cases} 0 & \text{falls } x < 2 \\ \frac{2-x}{6} & \text{falls } x > 2 \textit{ und } x < 8 \\ 1 & \text{falls } x > 8 \end{cases}$$

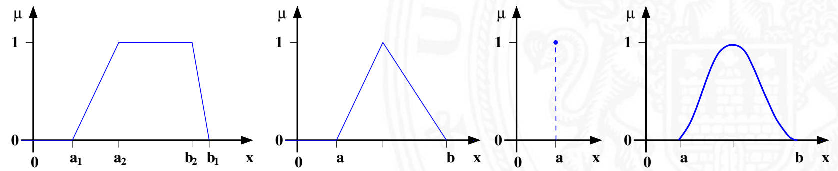
Fuzzy-Menge (cont.)

- Angabe diskreter Wertepaare(Element, Zugehörigkeitswert): (bei endlicher Universalmenge)

  $\mu_A = \{(0,0),(1,0),(2,0),(3,0.167),(4,0.33),\cdots(7,0.83),$
  $(8,1.0),(9,1),(10,1)\}$

  häufig auch in kompakter Notation:
  $\mu_A = 0/0 + 1/0 + 2/0 + 3/0.167 + 4/0.33 + \cdots + 7/0.83 +$
  $8/1.0 + 9/1 + 10/1$

  Angabe häufig auch in Tabellenform
- Beispiele für Fuzzy-Mengen:



Fuzzy-Menge (cont.)

Notation:

X endlich: $A = \mu_A(x_1)/x_1 + \cdots + \mu_A(x_n)/x_n = \sum_{i=1}^{n} \mu_A(x_i)/x_i$

X unendlich: $A = \int_X \mu_A(x)/x$

Fuzzy-Menge (cont.)

Beispiele:

- Die Menge der ganzen Zahlen, die ungefähr gleich 10 sind:

$$G_{10} = 0.1/7 + 0.5/8 + 0.8/9 + 1/10 + 0.8/11 + 0.5/12 + 0.1/13$$

- Die Menge der reellen Zahlen, die ungefähr gleich 10 sind:

$$G_{10} = \int_{\Re} e^{-(x-10)^2}/x$$

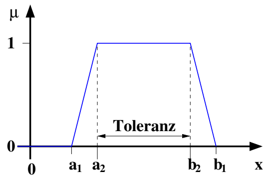
Fuzzy-Menge (cont.)

Als **Trägermenge** $supp(\mu_A)$ einer Fuzzy-Menge bezeichnet man die Menge aller Elemente aus X mit positiver Zugehörigkeit zu A:

$$supp(\mu_A) = \{x \in X \mid \mu_A(x) > 0\}$$

Die **Toleranz** $toll(\mu_A)$ beschreibt, in welchem Intervall $[a, b]$ der Zugehörigkeitsgrad gleich 1 ist:

$$toll(\mu_A) = [a, b] = \{x \in X \mid \mu_A(x) = 1\} \qquad (a, b\ const; a < b)$$

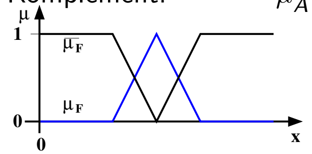


- falls $a_2 = b_2$; Dreiecksförmige ZF $\rightarrow$ keine Toleranz
- falls $a_1 = a_2 = b_2 = b_1$; Singleton $\rightarrow$ keinen Träger, keine Toleranz

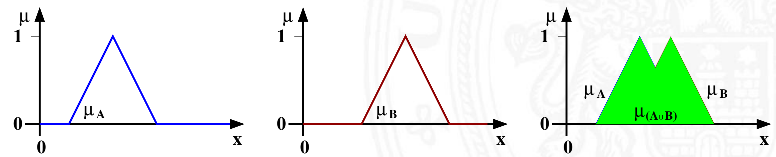
Verknüpfung von Fuzzy-Mengen

elementare Verknüpfungen für Fuzzy-Mengen A und B nach L. A. Zadeh:

- Komplement: $\mu_{\overline{A}}(x) = 1 - \mu_A(x)$

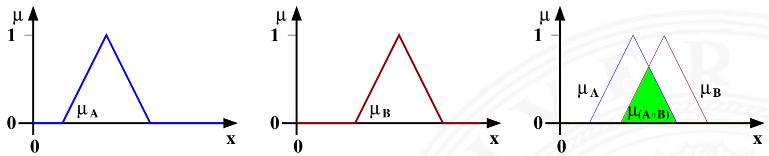


- Vereinigung: $\mu_{A \cup B}(x) = max[\mu_A(x), \mu_B(x)]$



Verknüpfung von Fuzzy-Mengen (cont.)

- Durchschnitt: $\mu_{A \cap B}(x) = min[\mu_A(x), \mu_B(x)]$



t-Norm (triangular norm): T:$[0, 1] \times [0, 1] \longrightarrow [0, 1]$

- 1 ist neutrales Element: $T(a, 1) = a$
- Monotonie: $T(a, b) \leq T(c, d)$, falls $a \leq c$ und $b \leq d$
- Kommutativität: $T(a, b) = T(b, a)$
- Assoziativität: $T(a, T(b, c)) = T(T(a, b), c)$

- Verallgemeinerung der klassischen Mengen-Operatoren
- alternative Realisierungen der Operatoren möglich

Linguistische Variablen und Linguistische Terme

- Fuzzy-Mengen werden zumeist zur Modellierung **linguistischer Terme** eingesetzt
- viele Begriffe der natürlichen Sprache lassen sich durch Fuzzy-Mengen charakterisieren
- ein *linguistischer Term (Wert, Label)* ist die Quantifizierung eines Begriffes der natürlichen Sprache durch eine Fuzzy-Menge
- eine **linguistische Variable** ist eine Variable, die eine Reihe linguistischer Terme annehmen kann
- häufig fünf bis zehn linguistische Terme pro linguistischer Variable

Linguistische Variablen und Linguistische Terme (cont.)

Beispiele:

- linguistische Variable: „**GESCHWINDIGKEIT**“

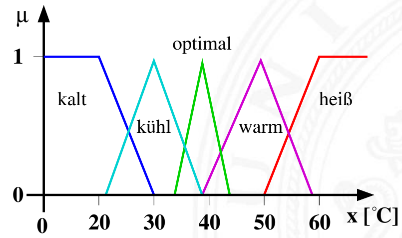
  linguistische Terme von „**GESCHWINDIGKEIT**“
  „**hoch**“, „**niedrig**“, „**rasant**“, „**ökonomisch**“

- linguistische Variable: „**GEBÄUDE**“

  linguistische Terme von „**GEBÄUDE**“
  „**Hütte**“, „**Bungalow**“, „**Wolkenkratzer**“

Linguistische Variablen und Linguistische Terme (cont.)

- linguistische Variable: „**Badewassertemperatur**
  linguistische Terme von „**Badewassertemperatur**“
  „**kalt**“, „**kühl**“, „**optimal**“, „**warm**“, „**heiß**“



Linguistische Variable

Eine **linguistische Variable** ist durch ein Quintupel charakterisiert

$$(V, T, \Omega, G, B)$$

Dabei ist:

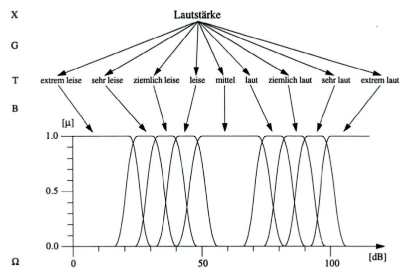
V: der Name der linguistischen Variable

T: eine Menge von linguistischen Termen von V, wobei jeder Wert eine Fuzzy-Menge in dem Universum Ω ist

G: Menge syntaktischer Regeln, die T aus einer Menge von Grundtermen erzeugt

M: Menge semantischer Regeln, mit der jedem Term eine unscharfe Menge von Ω zugewiesen werden kann

Linguistische Variable (cont.)



Syntax der linguistischen Variable „*Lautstärke*“ (G)	
Verzweigende Regeln	**Lexikalische Regeln**
(1) T → MT	(2) MT → leise
(3) T → NT	(4) NT → mittel
(5) T → PT	(6) PT → laut
(7) T → Hecke MT	(8) Hecke → sehr
(9) T → Hecke PT	(10) Hecke → ziemlich
	(11) Hecke → äußerst
Semantik der linguistischen Variable „*Lautstärke*“ (B)	
Verzweigende Regeln	**Lexikalische Regeln**
(12) M(MT) → M(leise)	(13) M(ziemlich-minus) → (m - 8, n - 8, α, β)
(14) M(NT) → M(mittel)	(15) M(sehr-minus) → (m - 16, n - 16, α, β)
(16) M(PT) → M(laut)	(17) M(extrem-minus) → (0, n - 25, 0, β)
(18) M(H PT) → M(PH) ◊ M(PT)	(19) M(ziemlich-plus) → (m + 8, n + 8, α, β)
(20) M(H MT) → M(MH) ◊ M(MT)	(21) M(sehr-plus) → (m + 16, n + 16, α, β)
(22) M(MH) → M(ziemlich-minus)	(23) M(extrem-plus) → (m + 25, 150, α, 0)
(24) M(MH) → M(sehr-minus)	(25) M(leise) → $(41, 47, 9, 9)_{SZ}$
(26) M(MH) → M(extrem-minus)	(27) M(mittel) → $(50, 72, 10. 10)_{SZ}$
(28) M(PH) → M(ziemlich-plus)	(29) M(laut) → $(75, 81, 9, 9)_{SZ}$
(30) M(PH) → M(sehr-plus)	
(31) M(PH) → M(extrem-plus)	

Quelle:Benno Biewer, Fuzzy-Methoden, Springer

Darstellungen der Zugehörigkeitsfunktionen

- Diskrete Darstellung
  - Array mit fester Größe
  - Speicherung der ZF-Werte für den gesamten x-Wertebereich
  - beliebige Formen
- Parametrische Darstellung
  - Funktionen mit Parametern
  - wenig Speicherplatz
  - typische Arten: Singleton, Dreiecksform, Trapezform, Glockenkurve, B-Spline Basisfunktionen

Erstellen der Zugehörigkeitsfunktionen

- Kontext-abhängige Spezifikation
  - experimentell, unter Berücksichtigung der jeweiligen Anwendung
- Aufbau durch Sample-Daten
  - Clustering
  - Lagrange-Interpolation
  - „Least-square Curve Fitting“
  - Neuronale Netze
- Wissenserwerb durch Experten
  - ein Experte oder mehrere
  - direkt und indirekt

Grundidee der Fuzzy-Regelung

- Beschreibung des gewünschten Reglerverhaltens mit Hilfe umgangssprachlicher, qualitativer Regeln
- Quantifizierung linguistischer Werte durch Fuzzy-Mengen
- Regel–Auswertung durch Verfahren der Fuzzy-Logik bzw. der Interpolation

Fuzzy-Regeln

In einer Fuzzy-Regelung wird die Einflußnahme auf die dynamischen Verhältnisse eines Fuzzy-Systems durch eine Menge linguistischer Beschreibungsregeln in der folgenden Form charakterisiert

IF (eine Menge Konditionen werden erfüllt)
THEN (eine Menge Konsequenzen können bestimmt werden)

In den Prämissen (Antecedenten) vom IF-Teil:
linguistische Variablen aus der Domäne der Prozesszustände

In den Konklusionen (Konsequenten) vom THEN-Teil:
linguistische Variablen aus der Regelungsdomäne

Prinzip der Fuzzy-Regelung

- **intelligente Regelung**
  - Verwendung von Expertenwissen
- **linguistische Regelung**
  - Regelung ist transparent
  - ein Pluspunkt für Mensch-Maschine-Schnittstelle
- **parallele Regelung**
  - Modularisierung
  - hohe Verarbeitungsgeschwindigkeit

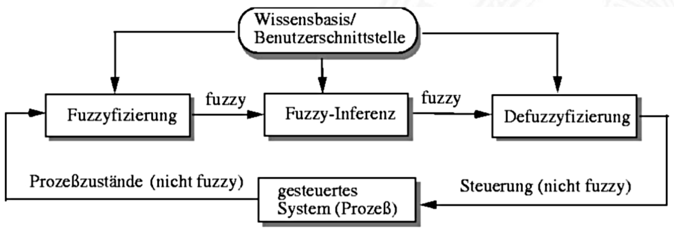
Vorteile der Fuzzy-Regelung

- Reglerentwurf ohne besondere Modellkenntnisse möglich
- Reglerentwurf effizient
- Echtzeit-Anforderungen erfüllt
- Robustheit auch beim Einsatz von billigen Sensoren

Komponenten der Fuzzy-Regelung

Ein kompletter Fuzzy-Controller besteht aus insgesamt vier Komponenten

- einer Wissensbasis
- einem Fuzzyfizierer
- einer Inferenz-Maschine
- und einem Defuzzyfizierer



Beispiel eines Fuzzy-Reglers

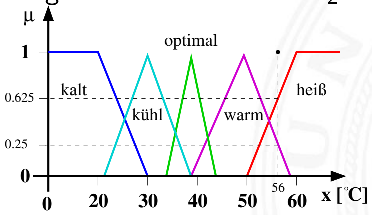
Regelung der Badewassertemperatur über die Zuführung kalten oder warmen Wassers.

- Sensoren: Temperatursensor
- Stellglieder: Zulauf_kalt, Zulauf_warm
- Linguistische Variablen: H_2Otemp
  (Terme: kalt, kühl, optimal, warm, heiß)
  Zul_kW (Terme: zu, mittel, offen)
  Zul_wW (Terme: zu, mittel, offen)

Beispiel eines Fuzzy-Reglers (cont.)

Aktuelle Ausgabe des Temp-Sensors: 56 °C

- Fuzzifizierung
  - hier: Fuzzifizierung des Eingabewertes als Singleton
- Ermittlung des Zugehörigkeitsgrades der Fuzzy-Menge des Eingabewertes (hier Singleton) mit den Fuzzy-Mengen der Terme der linguistischen Variablen H_2Otemp:



- Vektor der Zugehörigkeitsgrade: (0.0, 0.0, 0.0, 0.25, 0.625)

Beispiel eines Fuzzy-Reglers (cont.)

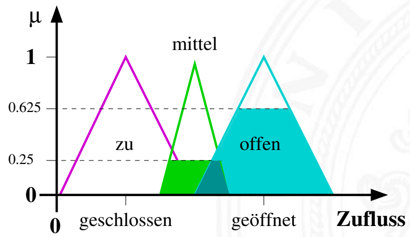
Regelbasis

1. WENN H_2Otemp = „heiß“, DANN Zul_kW = „offen“, Zul_wW = „zu“
2. WENN H_2Otemp = „warm“, DANN Zul_kW = „mittel“, Zul_wW = „zu“
3. WENN H_2Otemp = „kühl“, DANN Zul_kW = „zu“, Zul_wW = „mittel“
4. WENN H_2Otemp = „kalt“, DANN Zul_kW = „zu“, Zul_wW = „offen “

Regelauswertung mittels Min-Max Operators entsprechend Erfüllungsgrad:
Min-Max Operator schneidet Ergebnismenge auf Höhe des Erfüllungsgrades der Regel ab.

Beispiel eines Fuzzy-Reglers (cont.)

- Regel 1 zu 0,625 erfüllt ⟶ linguistischer Term „offen“ der Linguistischen Variable *Zul_kW* wird auf 0.625 begrenzt
- Regel 2 zu 0,25 erfüllt ⟶ inguistischer Term „mittel“ der Linguistischen Variable *Zul_kW* wird auf 0.25 begrenzt

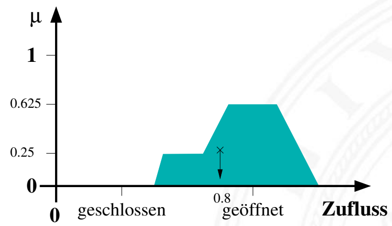


Linguistische Variable *Zul_kW*

Beispiel eines Fuzzy-Reglers (cont.)

Defuzzifizierung

- Defuzzifiziereung hier mittels Schwerpunktbildung:

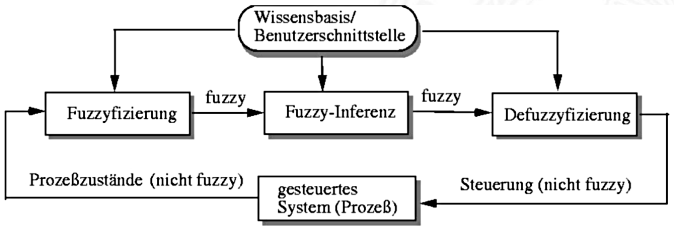


- das Ventil Zulauf_kalt wird etwa zu 80% geöffnet
- das Ventil Zulauf_warm bleibt geschlossen (keine Regel feuert und der Schwerpunkt liegt somit über geschlossen)

Komponenten der Fuzzy-Regelung

Ein kompletter Fuzzy-Controller besteht aus insgesamt vier Komponenten

- einer Wissensbasis
- einem Fuzzyfizierer
- einer Inferenz-Maschine
- und einem Defuzzyfizierer



Wissensbasis

In der Wissensbasis ist das Expertenwissen abgelegt, auf das sich ein Fuzzy-System während einer Regelung stützt, das sind

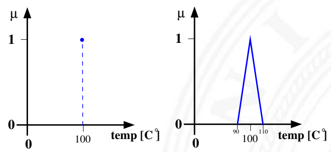
1. die Zugehörigkeitsfunktionen des Fuzzyfizierers
   in rechner-internen Darstellungen
2. die Zugehörigkeitsfunktionen
   mit denen die linguistischen Terme der linguistischen Variablen (die Ein- und Ausgangsgrößen) mathematisch beschrieben werden
3. die Regelungsstrategien
   in Form von *wenn-dann*-Regeln abgespeichert

Partitionierung

- linguistische Terme der linguistischen Variablen werden festgelegt
- jede der Variablen $X_1, \cdots, X_n$ und Y wird mit Hilfe von Fuzzy-Mengen partitioniert
- auf X_1 werden so t verschiedene Fuzzy-Mengen definiert, mit $\mu_1^1, \ldots, \mu_t^1 \in \mathcal{F}(X_1)$
- jede dieser Mengen wird mit einem linguistischen Term (z. B. kalt, kühl, warm, heiss) assoziiert

Fuzzyfizierer

- Der Fuzzyfizierer wandelt die „scharfen“ Eingangsgrößen in Fuzzy-Mengen um.
- Die dafür vorgesehenen Zugehörigkeitsfunktionen werden dazu wie eine Hülle um den jeweiligen Eingangswert gelegt.

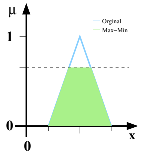


- Mit dem Fuzzyfizierer wird es möglich, Unschärfen der Eingangsgrößen, wie z.B. Fehlertoleranzen von Sensoren, zu berücksichtigen
  (Beispiel oben rechts: Messungenauigkeit $\pm 10\%$).

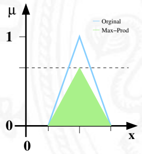
Inferenz-Maschine

- Die Inferenz-Maschine vergleicht die fuzzyfizierten Eingangswerte mit den Zugehörigkeitsfunktionen der Antecedenten für jede Regel.
- Daraus erschließt sie durch geeignete Kombination die Fuzzy-Mengen der Ausgangsvariablen (Konsequenten).
- Für die mathematische Modellierung des Vergleichs und des Schlußfolgerns existieren viele Lösungsvorschläge, Z. B.:

Max-Min-Inferenz



Max-Prod-Inferenz



Inferenz-Maschine (cont.)

Beispiel: Gegeben sei ein Regelsystem mit zwei Antecedenten A und B und einer Konsequenten C:

R_1: IF (x is A_1 and y is B_1) THEN (z is C_1)

R_2: IF (x is A_2 and y is B_2) THEN (z is C_2)

...

R_k: IF (x is A_k and y is B_k) THEN (z is C_k)

Inferenz-Maschine

MAX-MIN-Inferenz

- Zunächst werden die fuzzyfizierten Eingangsdaten A' und B' mit den ZF A_i und B_i der i-ten Regel verglichen, und man erhält so für jede Regel die *Übereinstimmungsmaße* α_{A_i} und α_{B_i}

$$\alpha_{A_i} = \max(\min_j(A', A_{ij}))$$
$$\alpha_{B_i} = \max(\min_j(B', B_{ij}))$$

- Diese Übereinstimmungsmaße werden schließlich zu einem Gesamtmaß ω'_i verknüpft, das den Erfüllungsgrad der gesamten Eingangsbedingungen der i-ten Regel angibt

$$\omega'_i = \min(\alpha_{A_i}, \alpha_{B_i})$$

Inferenz-Maschine (cont.)

MAX-MIN-Inferenz

- Der Erfüllungsgrad kann noch zusätzlich mit einem Regelgewicht $r_i \in [0, 1]$ multipliziert werden
- Regeln, die z.B. in Alarmfällen die Sicherheit gewährleisten sollen, können dadurch gegenüber anderen Regeln stärker gewichtet werden. Man erhält somit

$$\omega_i = r_i \cdot \omega'_i$$

- Die tatsächliche Schlußfolgerungsfunktion C'_i des Konsequenten C_i errechnet sich aus

$$C'_i = \min(\omega_i, C_i)$$

Inferenz-Maschine (cont.)

MAX-MIN-Inferenz

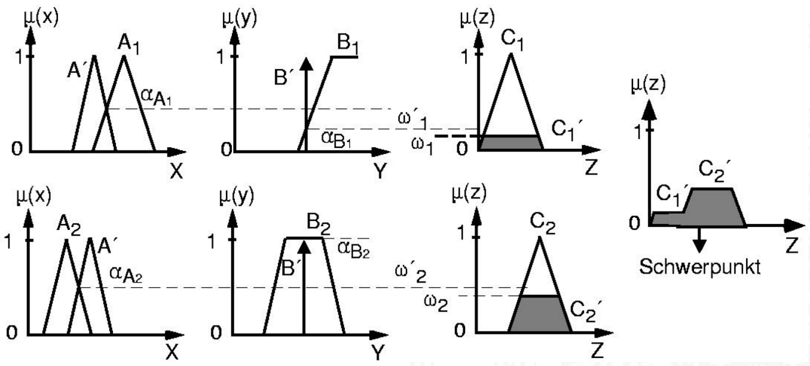
- Zuletzt faßt man alle Schlußfolgerungen C_i' zusammen und erhält die Ausgangsfunktion C_A

$$C_A = \max(C_1', C_2', \ldots, C_k')$$

- Bei Regelsystemen mit mehreren Ausgangsvariablen können die Ausgangsfunktionen unabhängig voneinander nach obigem Schema bestimmt werden

Inferenz-Maschine (cont.)

MAX-MIN-Inferenz



Defuzzyfikation

- Um in einem Regelungsprozeß konkrete Stellgrößen an die Aktuatoren senden zu können, müssen aus den durch die Inferenz gewonnenen Ausgangsfunktionen „scharfe“ Ausgangswerte gebildet werden
- übliche Vorgehensweise ist die Schwerpunktmethode
  - Ausgangswert wird hierbei als Schwerpunkt der Ausgangsfunktion bezüglich ihrer Abszisse berechnet
  - andere Strategien z. B. Mittelwert-Max, MaxLeft, MaxRight,...

Fuzzy-Regler: Mamdani-Typ (MAX-MIN-Inferenz)

- nach Ebrahim Mamdani, London University, 1975 vorgestellt
- der klassische Fuzzy-Regeler des Mamdani-Typs basiert auf einer endlichen Menge $\mathcal{R}$ symbolischer Regeln $R \in \mathcal{R}$:

$$\begin{array}{lll} R_k: & \text{IF} & (x_1 \text{ is } A^1_{R_k}) \text{ and } (x_2 \text{ is } A^2_{R_k}) \text{ and } \ldots \text{ and } (x_n \text{ is } A^n_{R_k}) \\ & \text{THEN} & y \text{ is } B_k \end{array}$$

wobei $A^i_{R_k}$ den in Regel R_k berücksichtigten linguistischen Term der linguistischen Variablen i bedeutet und B_k eine Fuzzy-Menge mit den gleichen Eigenschaften wie im IF-Teil ist; mit $k = 1, \ldots, t$, und t die Anzahl der Linguistischen Terme, die y modellieren

Fuzzy-Regler: Mamdani-Typ (MAX-MIN-Inferenz) (cont.)

- Kontrollregeln nicht als logische Implikation (modus ponens), sondern im Sinne einer stückweise definierten Funktion auffassen
- die Teilprämissen einer Regel werden mit dem MIN-Operator zusammengefasst (UND-Verknüpfung); falls ODER-Verknüpfung enthalten, Regel splitten
- Zusammenfassen der Ausgabewerte aller aktiven Regeln geschieht mit dem *MAX*-Operator

Probleme der Regler des Mamdani-Typs

- viele Freiheitsgrade beim Entwurf
  - Implikations-Relation
  - Inferenz-Mechanismen
  - Fuzzyfikation- und Defuzzyfikationsstrategie
- Auswahl und Quantifizierung der linguistischen Werte schwierig
  - keine systematischen Richtlinien $\Rightarrow$ Erfahrungswerte
- Auswirkung der Wahl der Zugehörigkeitsfunktions-Form
  - warum Dreiecke/Trapeze?
  - andere Funktionen?
- Bewertungskriterien für einen optimalen Regler
  - Glätte
  - Approximations-Genauigkeit
- Nachweis der Stabilität
  - wie bei fast allen nicht-linearen Systemen

Zwei Typen adaptiver Fuzzy-Regler

- **Sugeno-Typ**
  - nach Michio Sugeno, etwa 1985 vorgestellt
  - Singleton als Fuzzy-Set der Ausgabemenge (Konsequenz)
    Ausgabe: Funktion der Eingabewerte
  - endliche Menge $\mathcal{R}$ symbolischer Regeln $R \in \mathcal{R}$:

$$\begin{array}{lll} R_k: & \text{IF} & (x_1 \text{ is } A^1_{R_k}) \text{ and } (x_2 \text{ is } A^2_{R_k}) \text{ and } \ldots \text{and } (x_n \text{ is } A^n_{R_k}) \\ & \text{THEN} & y_k = f(x_1, \ldots, x_n) \end{array}$$

  - beim Sugeno-Regler 0ter Ordnung degeneriert die Funktion f zur Konstanten
  - Parameter der Funktion f im allg. Fall können adaptiv angepasst werden
  - Sugeno-Inferenz ist ähnlich der Mamdani-Inferenz
  - Erfolgreich eingesetzt bei Funktionsapproximation und überwachtem Lernen

Zwei Typen adaptiver Fuzzy-Regler (cont.)

- **B-Spline-Typ**
  - Nachbildung der B-Spline-Interpolation mittels *a priori* Wissen
  - ein spezieller Sugeno-Typ, aber effektiver, schneller
  - geeignet für überwachtes Lernen und unüberwachtes Lernen
  - B-Spline-Basisfunktionen für Modellierung der lingu. Terme
  - Fuzzy-Singletons als Zugehörigkeitsfunktion der Ausgänge
  - sehr gut geeignet für adaptive Reger, die automatisch aus Trainingsdaten konstruiert werden können

aber: Regler ist bei n Eingängen x_n und einem Ausgang y vollständig über einem n-dimensionalen Gitter definiert
  - Regelbasis hängt exponentiell von der Dimension des Eingangsraumes ab
  - nur für niedrigdimensionale Probleme geeignet
  - *Fluch der Dimensionalität*

Vergleich von Fuzzy-Controller Modellen

IF-Teil:

- alle Fuzzy-Controller setzen Fuzzy-Mengen zur Modellierung von linguistischen Termen für die Eingabe ein
- Eingabebereich wird überlappend partitioniert
- dies reflektiert die vage Modellierung durch linguistische Konzepte
- ein kontinuierlicher Übergang der Ausgabewerte wird ermöglicht

Vergleich von Fuzzy-Controller Modellen (cont.)

IF-Teil:

- IF-Teil einer Regel wird folgendermaßen modelliert:
$$(x_1 \text{ is } A^1_{i_1}) \text{ and } (x_2 \text{ is } A^2_{i_2}) \text{ and } \ldots (x_n \text{ is } A^n_{i_n})$$
wobei x_j die j-te Eingabe ($j = 1, \ldots, n$) und $A^j_{i_j}$ der i-te linguistische Term definiert auf x_j ist
- „und“-Operation wird als so genannte t-Norm implementiert
- in den meisten Anwendungen handelt es sich dabei um eine Minimum- oder Produkt-Operation

Vergleich von Fuzzy-Controller Modellen (cont.)

Zugehörigkeitsfunktion:

- typisch: trianguläre oder trapezoide Zugehörigkeitsfunktionen
- modernere Systeme: „Gaussian“, „Cauchy“, „*sinc*“, „Hyperbolic Tangent“,...
- Problem: Alle Funktionen brauchen neben den Partitionspositionen (*Knoten*) weitere Parameter
- weil die Knoten ggf. das Ergebnis einer intrinsischen Partitionierung sind, ist die Wahl der übrigen Parameter weder natürlich noch intuitiv
- Linguistische Terme die auf B-Spline Basisfunktionen beruhen, können allein auf Grundlage der Knoten gebildet werden und brauchen *keine* weiteren Parameter

[Bie97] Benno Biewer:
Fuzzy-Methoden.
Springer-Verlag; Berlin, 1997

[BZ70] R.E. Bellman, L.A. Zadeh:
Decision-Making in a Fuzzy Environment.
In: *Management Science*
17 (1970)

[Lip06] Wolfram-Manfred Lippe:
Soft-Computing mit Neuronalen Netzen, Fuzzy-Logic und Evolutionären Algorithmen.
Springer-Verlag; Berlin, 2006

[Zad65] L.A. Zadeh:
Fuzzy Sets.
In: *Information Control*
8 (1965), S. 338–353