



64-424 Intelligente Roboter

[http://tams.informatik.uni-hamburg.de/
lectures/2011ws/vorlesung/ir](http://tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/ir)

Jianwei Zhang



Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

Wintersemester 2011/2012



Gliederung

1. Grundlagen der Sensorik
2. Winkel und Bewegungen
3. Kräfte und Druck
4. Abstandssensoren
5. Scandaten verarbeiten
6. Rekursive Zustandsschätzung
7. Sichtsysteme
8. **Fuzzy-Logik**

Adaptive Methoden zur Regelung

Fuzzy-Regelung

Charakteristische Funktion / Zugehörigkeitsfunktion

Fuzzy-Menge

Linguistische Variablen und Terme



Gliederung (cont.)

Darstellungen der Zugehörigkeitsfunktionen
Fuzzy-Regelung
Literatur

9. Steuerungsarchitekturen





Einführung in die Fuzzy-Regelung

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

Richard P. Feynman

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

Albert Einstein



Adaptive Methoden zur Regelung

- ▶ Regelung kann aufgefasst werden als Abbildung von einem Sensorraum auf Aktionen
- ▶ 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
- ▶ Oft sind die Sensordaten ungenau, verrauscht und/oder hochdimensional



Adaptive Methoden zur Regelung (cont.)

- ▶ Das Erstellen einer optimalen Abbildung zwischen Sensorraum und Aktionen ist dann mit klassischen regelungstechnischen Methoden sehr schwierig
- ▶ Um solche Systeme regeln zu können, muss eine einfachere Methode zur Beschreibung benutzt werden oder die Regelung muss sich an die Bedingungen anpassen



Adaptive Methoden zur Regelung (cont.)

Modelle für Adaptive Regler:

- ▶ Neuronale Netze
 - ▶ *Adaptive Linear Combinator (ALC)*
 - ▶ *Multi-Layer Perceptron (MLP)*
 - ▶ *Functional Link Network*
 - ▶ *Radial Basis Functions (RBF)*
 - ▶ *Associative Memory Networks (AMNs)*
 - ▶ ...
- ▶ Fuzzy-Controller
 - ▶ *Conventional Controller (Mamdani-Typ)*
 - ▶ *Funktionsapproximator (TSK-Modell oder B-Spline-Modell)*



Lernverfahren

- ▶ Überwachtes Lernen
- ▶ Reinforcement-Lernen
- ▶ Unüberwachtes-Lernen

**Mehr zu Lernverfahren in der Vorlesung
„Algorithmisches 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 natürsprachliche Abstufungen von Begriffen wie „**groß**“, „**schön**“, „**stark**“ ...
- ▶ Menschliche Denk- und Verhaltensmodelle mit der einstufigen Logik
 - ▶ **Autofahren:** *wenn-dann*-Regeln
 - ▶ **Autoparken:** Genau auf den Millimeter?
- ▶ Unscharfe Sprache statt numerischer Beschreibung
 - ▶ „Bremse 2.52 m vor der Kurve!“ → nur in Maschinensystemen
 - ▶ „Bremse kurz vor der Kurve!“ → in natürlicher Sprache



Fuzzy-Regelung (cont.)

L.A. Zadeh's **principle of incompatibility**

„Stated informally, the essence of this principle is that as the complexity of a system increases, our ability to make precise and yet significant statements about its behaviour diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics.“



Fuzzy-Regelung (cont.)

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



Charakteristische Funktion

Scharfe Mengen lassen sich definieren durch Angabe ihrer charakteristischen Funktion:

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

wobei $\mu_{\mathcal{A}} : X \rightarrow \{0, 1\}$ ist



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:

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

- ▶ 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 $\mu_A : X \rightarrow [0, 1]$. Für alle $x \in X$ bezeichnet $\mu_A(x)$ den Grad der Zugehörigkeit (des Enthaltenseins) von x in A

Als **Trägermenge** einer Fuzzy-Menge bezeichnet man die Menge aller Elemente aus X mit positiver Zugehörigkeit zu A :

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



Fuzzy-Menge (cont.)

Notation:

$$\begin{aligned}
 X \text{ endlich: } \quad A &= \mu_A(x_1)/x_1 + \cdots + \mu_A(x_n)/x_n \\
 &= \sum_{i=1}^n \mu_A(x_i)/x_i
 \end{aligned}$$

$$X \text{ unendlich: } \quad A = \int_X \mu_A(x)/x$$



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:

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



Linguistische Variablen und Linguistische Terme

- ▶ In **Anwendungen** werden Fuzzy-Mengen zumeist eingesetzt zur Modellierung **linguistischer Terme**
- ▶ 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



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 Variable

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

$$(v, T(v), X, G, M)$$

Dabei ist:

- v** : der Name der Variable
- $T(v)$** : eine Menge von linguistischen Termen von v , wobei jeder Wert eine Fuzzy-Menge in dem Universum X ist
- G** : eine Syntaxregel, die $T(v)$ aus einer Menge von Grundtermen erzeugt
- M** : eine semantische Regel, die jedem Wert von $T(v)$ seine Bedeutung zuordnet



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 (Antezedenzen) vom IF-Teil:

linguistische Variablen aus der Domäne der Prozesszustände

In den Konklusionen (Konsequenzen) 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



Vergleich von Fuzzy-Controller Modellen

IF-Teil:

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



Vergleich von Fuzzy-Controller Modellen (cont.)

IF-Teil:

- ▶ Der IF-Teil einer Regel wird folgendermaßen modelliert:

$$(x_1 \text{ is } A_{i_1}^1) \text{ and } (x_2 \text{ is } A_{i_2}^2) \text{ and } \dots (x_n \text{ is } A_{i_n}^n)$$

wobei x_j die j -te Eingabe ($j = 1, \dots, n$) und $A_{i_j}^j$ der i -te linguistische Term definiert auf x_j ist

- ▶ Die „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.)

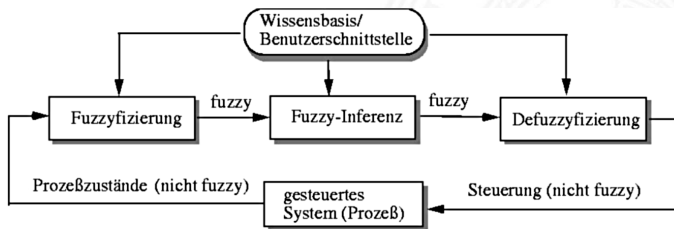
Zugehörigkeitsfunktion:

- ▶ typisch: trianguläre oder trapezoide Zugehörigkeitsfunktionen
- ▶ modernere Systeme: „Gaussian“, „Cauchy“, „*sinc*“, „Laplace“, „Logistic“, „Hyperbolic Tangent“
- ▶ Problem: Alle Funktionen brauchen neben den Partitionspositionen (*Knoten*) weitere Parameter
- ▶ Weil die Knoten das intrinsische Ergebnis einer 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

Komponenten der Fuzzy-Regelung

Ein kompletter Fuzzy-Controller besteht aus insgesamt vier Komponenten

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





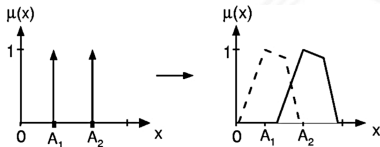
Wissensbasis

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

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

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



Inferenz-Maschine

- ▶ Die Inferenz-Maschine vergleicht die fuzzyfizzierten Eingangswerte mit den Zugehörigkeitsfunktionen der Antezedenzen für jede Regel
- ▶ Daraus erschließt sie durch geeignete Kombination die Fuzzy-Mengen der Ausgangsvariablen (Konsequenzen)
- ▶ Für die mathematische Modellierung des Vergleichs und des Schlußfolgerns existieren viele Lösungsvorschläge
- ▶ Eine einfache Methode ist die Inferenz mit Hilfe der Min-Max-Operatoren



Inferenz-Maschine (cont.)

Beispiel: Gegeben sei ein Regelsystem mit zwei Antezedenzen A und B und einer Konsequenz 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 (cont.)

- ▶ 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* a_{A_i} und a_{B_i}

$$a_{A_i} = \max(\min(A', A_i))$$

$$a_{B_i} = \max(\min(B', B_i))$$

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

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



Inferenz-Maschine (cont.)

- ▶ 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 der Konsequenz C_i errechnet sich aus

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



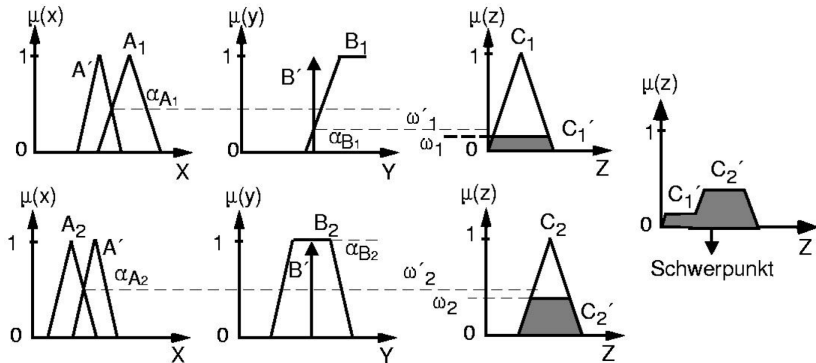
Inferenz-Maschine (cont.)

- ▶ Zuletzt faßt man alle Schlußfolgerungen C'_i zusammen und erhält die Ausgangsfunktion C_A

$$C_A = \max(C'_1, C'_2, \dots, C'_k)$$

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

Inferenz-Maschine (cont.)





Defuzzifikation

- ▶ Um in einem Regelungsprozeß konkrete Stellgrößen an die Aktuatoren senden zu können, müssen aus den durch die Inferenz gewonnenen Ausgangsfunktionen „scharfe“ Ausgangswerte gebildet werden
- ▶ Eine vernünftige Vorgehensweise ist die Schwerpunktmethode
- ▶ Der Ausgangswert wird hierbei als Schwerpunkt der Ausgangsfunktion bezüglich ihrer Abszisse berechnet



Fuzzy-Regler: Mamdani-Typ

- ▶ Der klassische Fuzzy-Regler des Mamdani-Typs basiert auf der direkten Nutzung symbolischer Regeln

IF $(x_1 \text{ is } A_{i_1}^1) \text{ and } (x_2 \text{ is } A_{i_2}^2) \text{ and } \dots \text{ and } (x_n \text{ is } A_{i_n}^n)$
 THEN $y \text{ is } B_k$

wobei B_k eine Fuzzy-Menge mit den gleichen Eigenschaften wie im IF-Teil ist und $k = 1, \dots, t$, und t die Anzahl der linguistischen Terme ist, die y modellieren

- ▶ Das Zusammenfassen der Ausgabewerte aller aktiven Regeln geschieht mit dem *max*-Operator
- ▶ Eine einfacher zu berechnende Variation des Zusammenfassens ist die einfache Addition der Ausgabewerte



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

- ▶ **IF** x_1 is A_1^i and x_2 is A_2^i and ... and x_n is A_n^i
THEN $y = p_0^i + p_1^i x_1 + p_2^i x_2 + \dots + p_n^i x_n$

wobei die Parameter von A_1^i bis A_n^i , sowie $p_0^i, p_1^i, p_2^i, \dots + p_n^i$ adaptiv gewonnen werden können

- ▶ Erfolgreich eingesetzt bei Funktionsapproximation und überwachtem Lernen

▶ B-Spline-Typ

- ▶ Nachbildung der B-Spline-Interpolation mit Hilfe von *a priori* Wissen
- ▶ Ein spezieller Sugeno-Typ, aber effektiver, schneller, geeignet für überwachtes Lernen und unüberwachtes Lernen



B-Spline Fuzzy-Regler

Eine *Regel* (i_1, i_2, \dots, i_n) mit n konjunktiven Termen in den IF-Teilen ist gegeben durch

IF $(x_1 \text{ is } X_{i_1, k_1}^1)$ and $(x_2 \text{ is } X_{i_2, k_2}^2)$ and \dots and $(x_n \text{ is } X_{i_n, k_n}^n)$
THEN $y \text{ is } Y_{i_1 i_2 \dots i_n}$

- ▶ x_j : die j -te Eingabe ($j = 1, \dots, n$)
- ▶ k_j : die Ordnung der B-spline Basisfunktion für x_j
- ▶ X_{i_j, k_j}^j : der i -te linguistische Term von x_j definiert durch eine B-spline Basisfunktion ist
- ▶ $i_j = 1, \dots, m_j$, repräsentiert wie fein die j -te Eingabe fuzzy-partitioniert ist
- ▶ $Y_{i_1 i_2 \dots i_n}$: die Kontrollpunkte (*deBoor-Punkte*) der *Regel* (i_1, i_2, \dots, i_n) sind



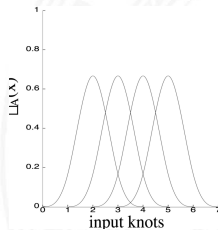
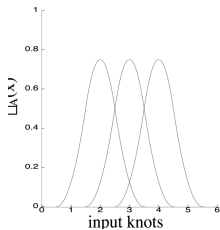
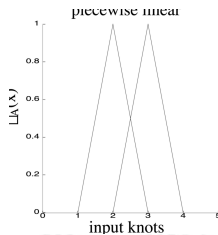
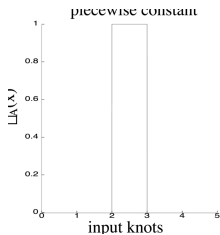
B-Spline Basisfunktion

- ▶ Sei x die Eingabe des zu regelnden Systems definiert auf dem Intervall $[x_1, x_m]$
- ▶ Mit einer Sequenz geordneter Parameter (Knoten) x_1, x_2, \dots ist der i -te B-spline $N_{i,k}$ der Ordnung k (Grad $k - 1$) rekursiv wie folgt definiert:

$$N_{i,k}(x) = \begin{cases} \begin{cases} 1 & \text{for } x \in [x_i, x_{i+1}) \\ 0 & \text{otherwise} \end{cases} & \text{if } k = 1 \\ \frac{x - x_i}{x_{i+k-1} - x_i} N_{i,k-1}(x) + \frac{x_{i+k} - x}{x_{i+k} - x_{i+1}} N_{i+1,k-1}(x) & \text{if } k > 1 \end{cases}$$

mit $i = 1, \dots, m - k$

B-Spline Basisfunktion (cont.)





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

