

Entscheidungsunterstützung/ Künstliche Intelligenz

Teil 4

**BiTS, Sommersemester 2005
Dr. Stefan Kooths**

Gliederung

1. Einführung
2. Entscheidungslehre
3. Entscheidungsunterstützungssysteme
4. Künstliche Intelligenz im Überblick
5. AHP und ANP
- 6. GENEFER-Technologien**
7. Prognosen (Finanzzeitreihen)
8. Simulation (Spielkonsolenmarkt)
9. MSS-Produktpräsentationen

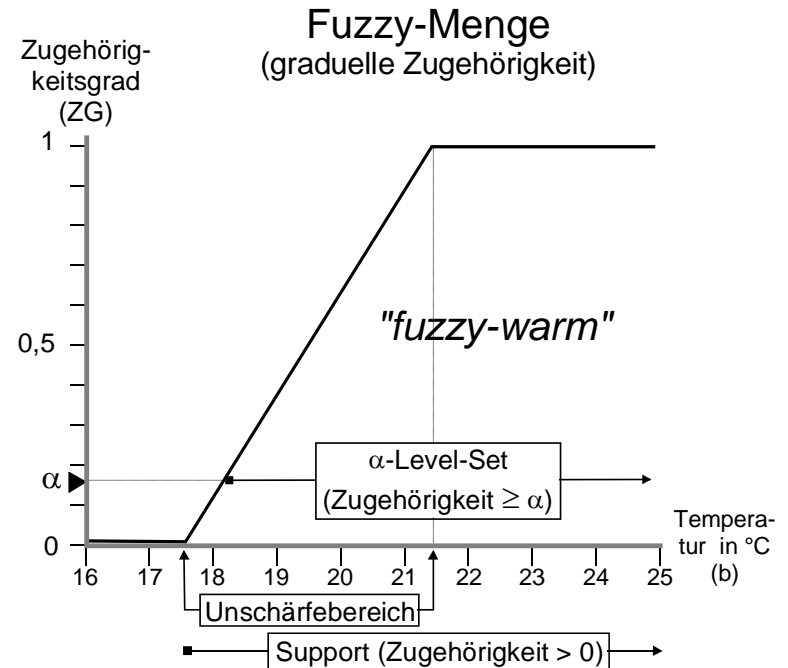
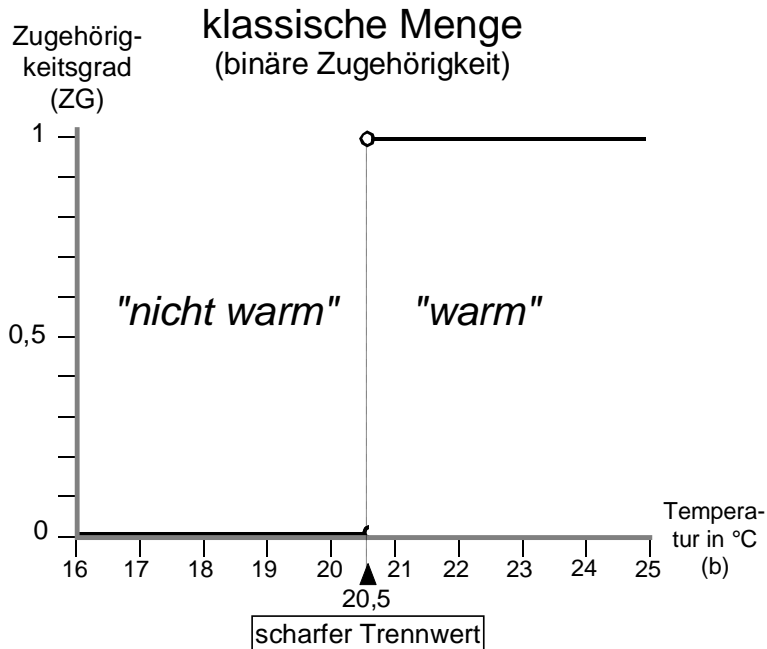
GENEFER-Technologien

- Basistechnologie: Fuzzy-Systeme
- Grundkonzept: Hybridisierung
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- Tuning
 - genetisch
 - neuronal (Backprop)

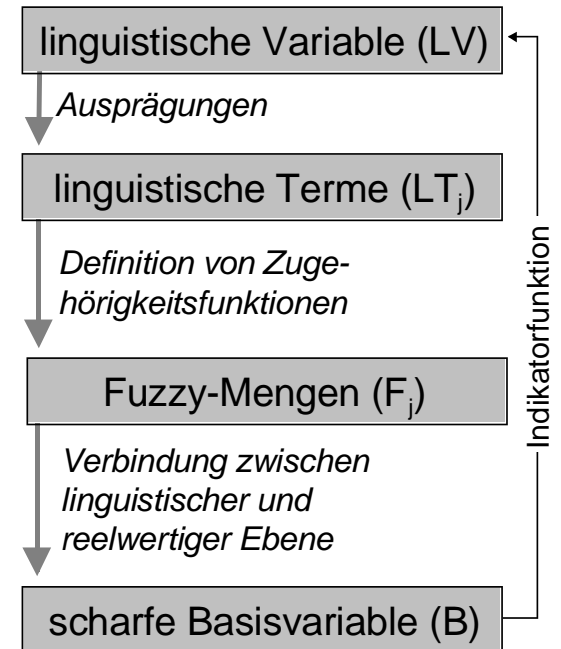
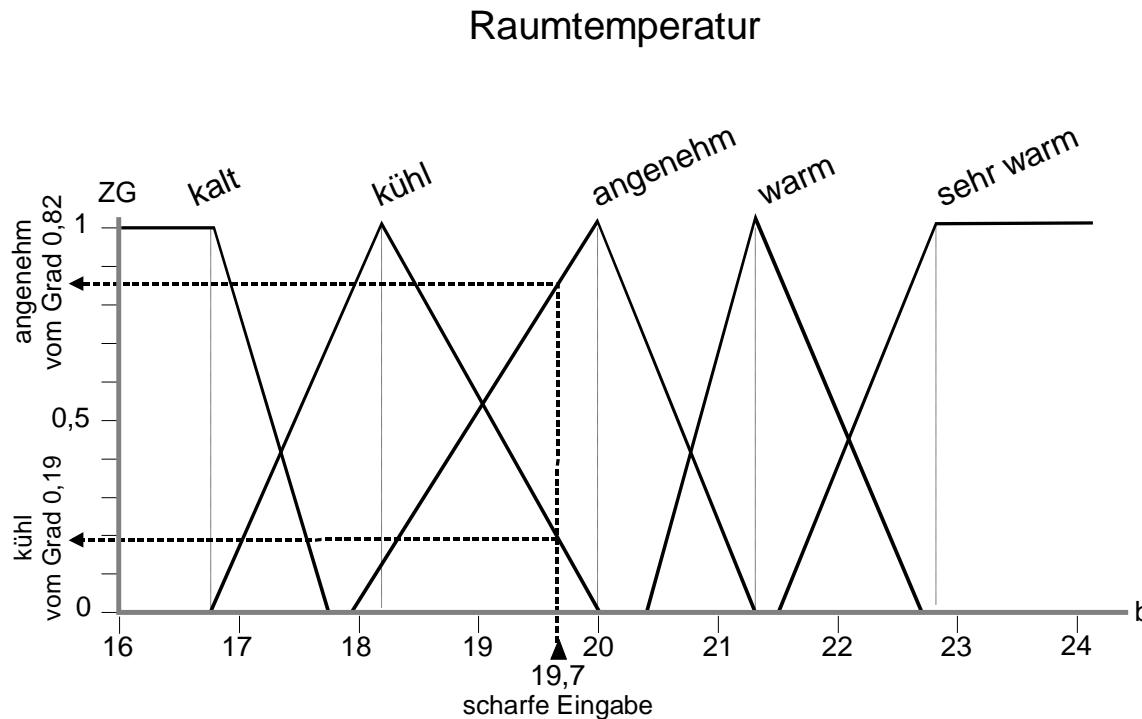
GENEFER-Technologien

- **Basistechnologie: Fuzzy-Systeme**
- Grundkonzept: Hybridisierung
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- Tuning
 - genetisch
 - neuronal (Backprop)

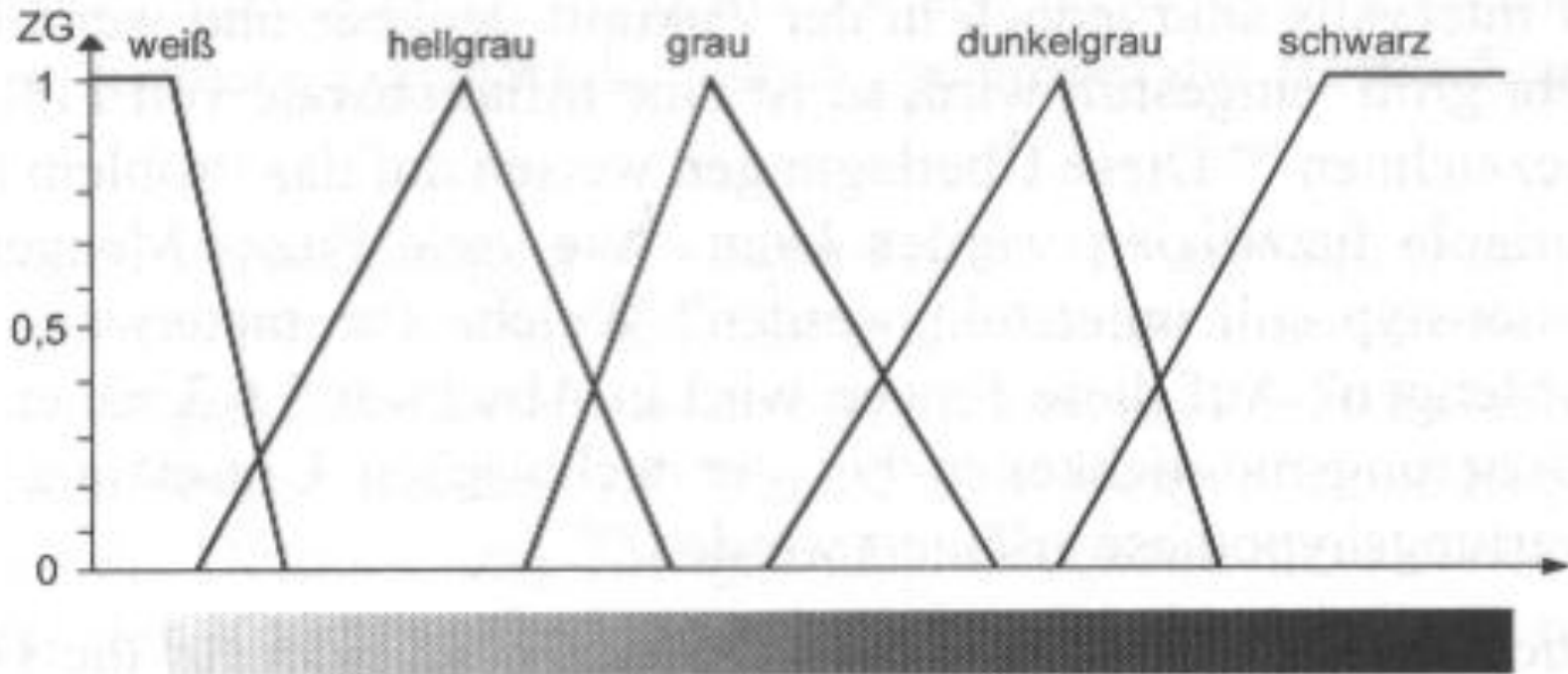
Klassische Mengen und Fuzzy Mengen



Fuzzifizierung

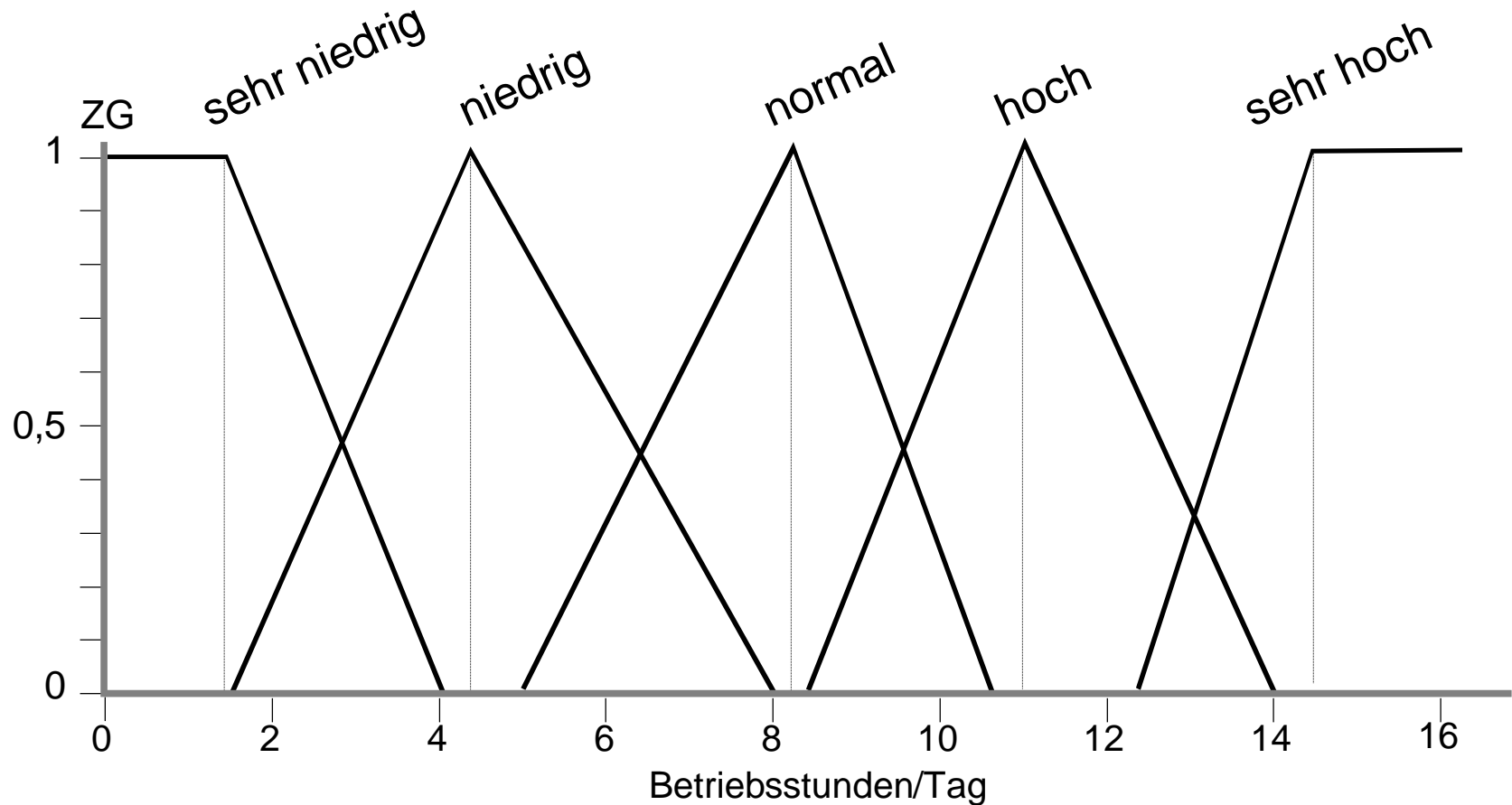


Fuzzifizierung: Schwarz-Weiß-Grau-Beispiel



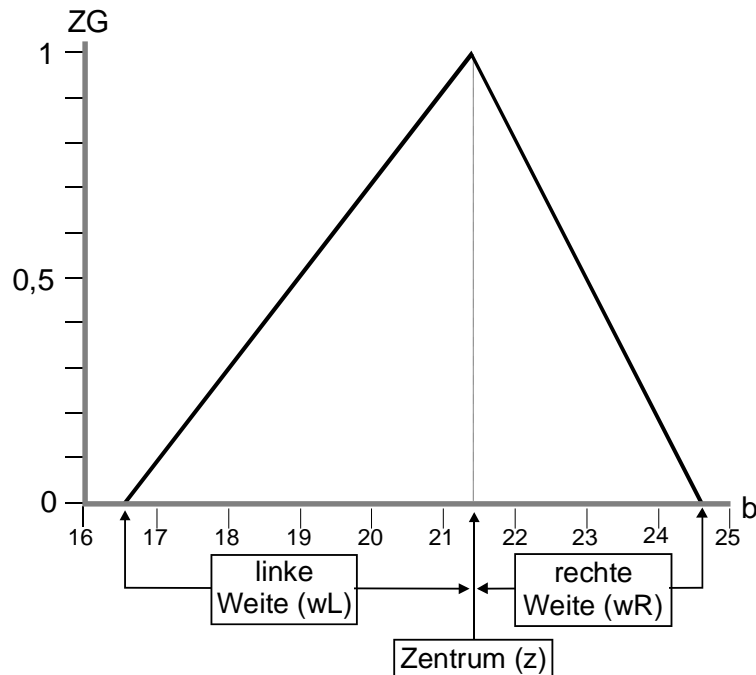
Fuzzifizierung: Ökonomisches Beispiel

Auslastungsgrad einer Maschine

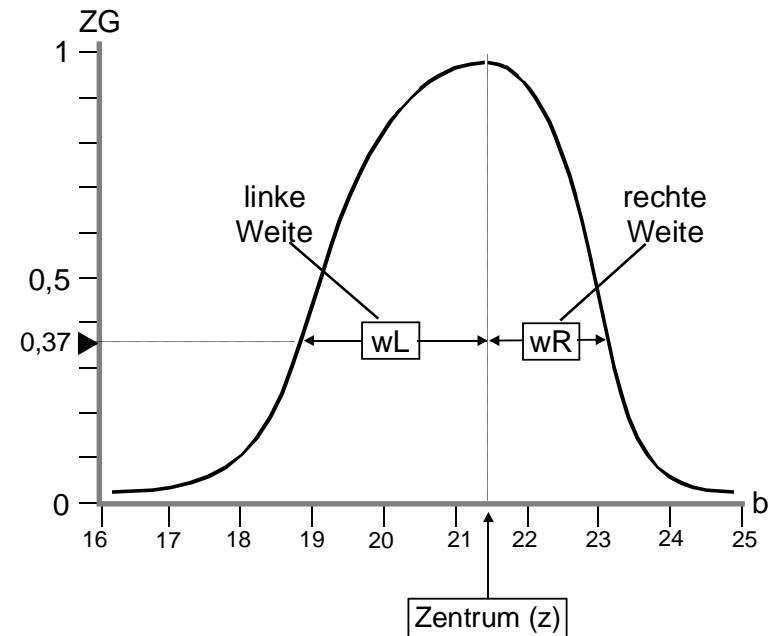


Fuzzy-Mengen: Grundformen

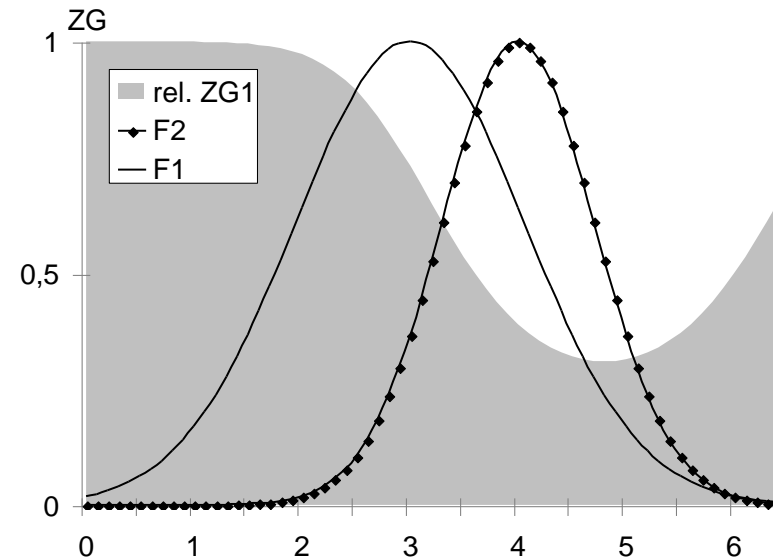
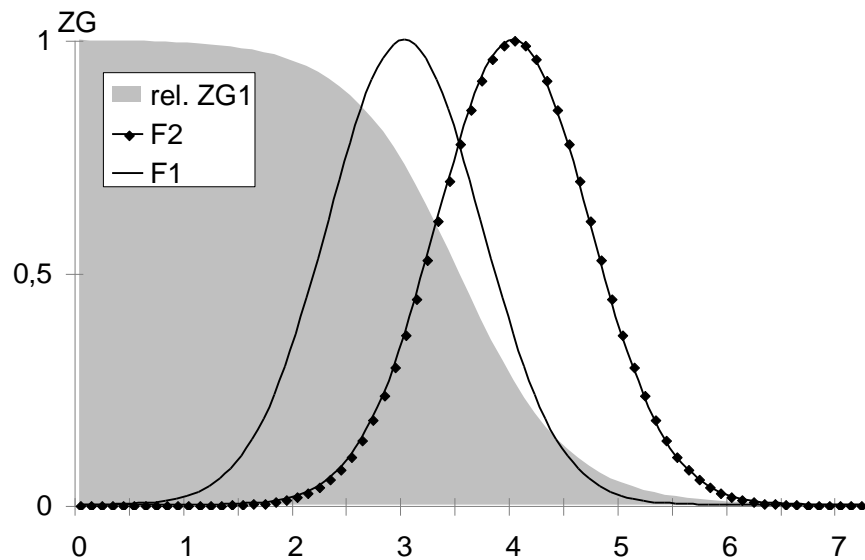
trianguläre Form



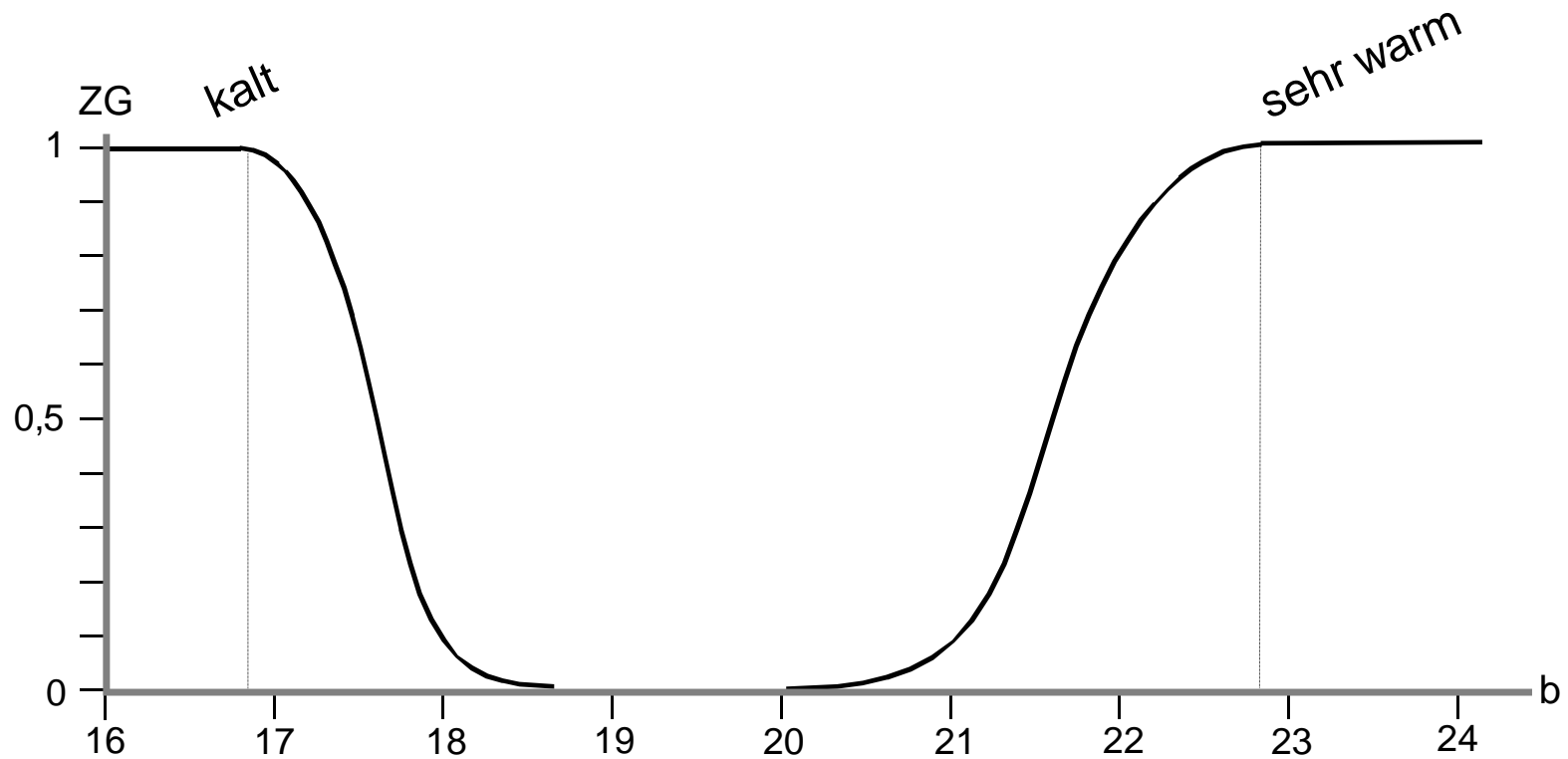
Gauß-Form



Gaußproblem: Dominanzeffekt der großen Weite



Randsättigung als Lösung des Gaußproblems

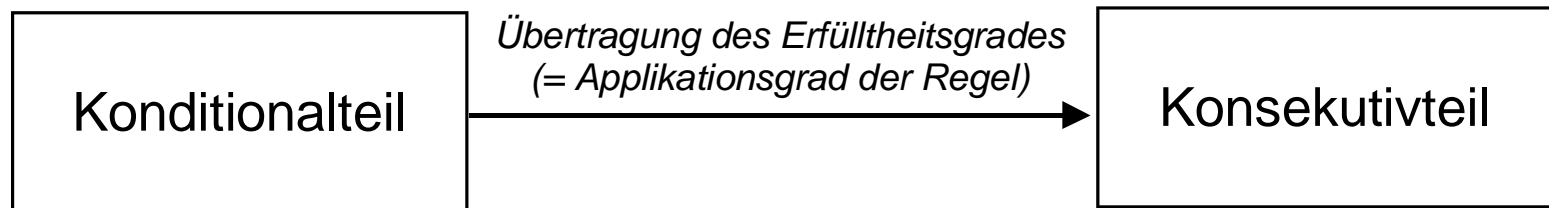


Fuzzy-Regeln: Aufbau und Inferenzprinzip

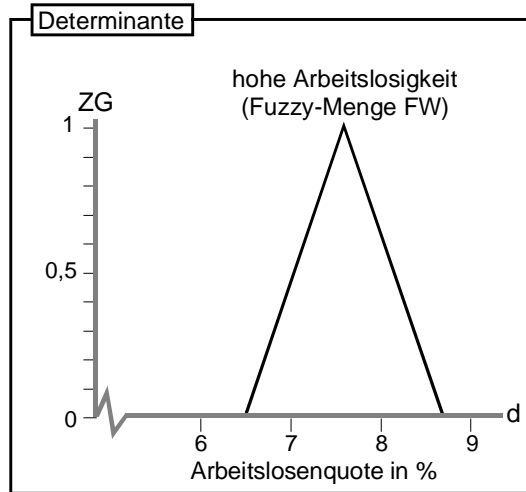
- Stilisierte Form einer Fuzzy-Regel



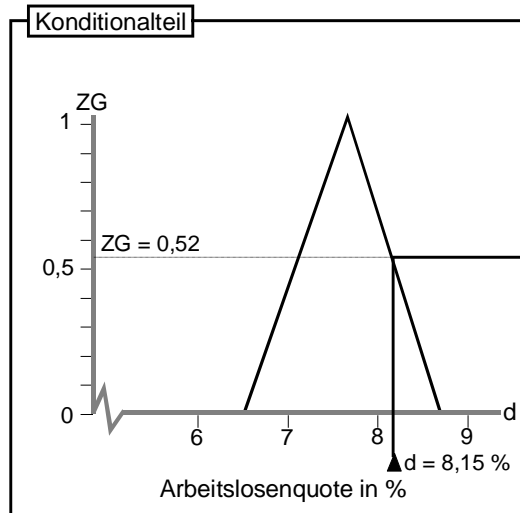
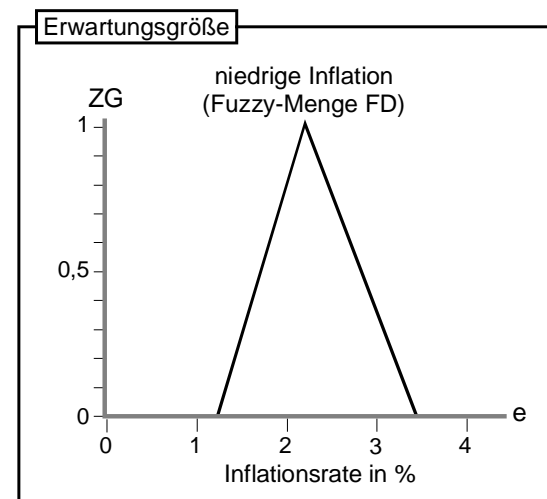
- Inferenzprinzip



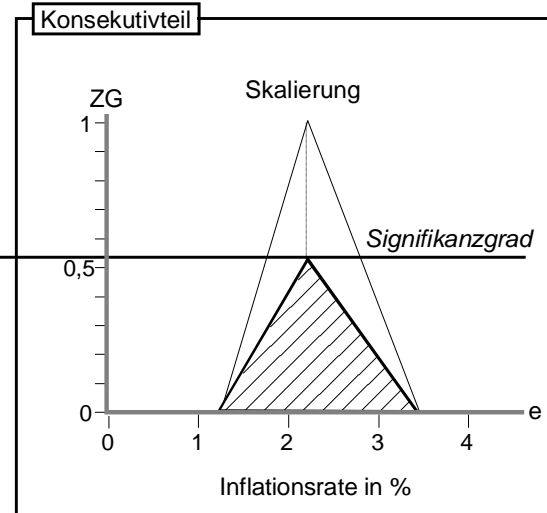
Fuzzy-Inferenz: Beispiel Inflationsprognosen



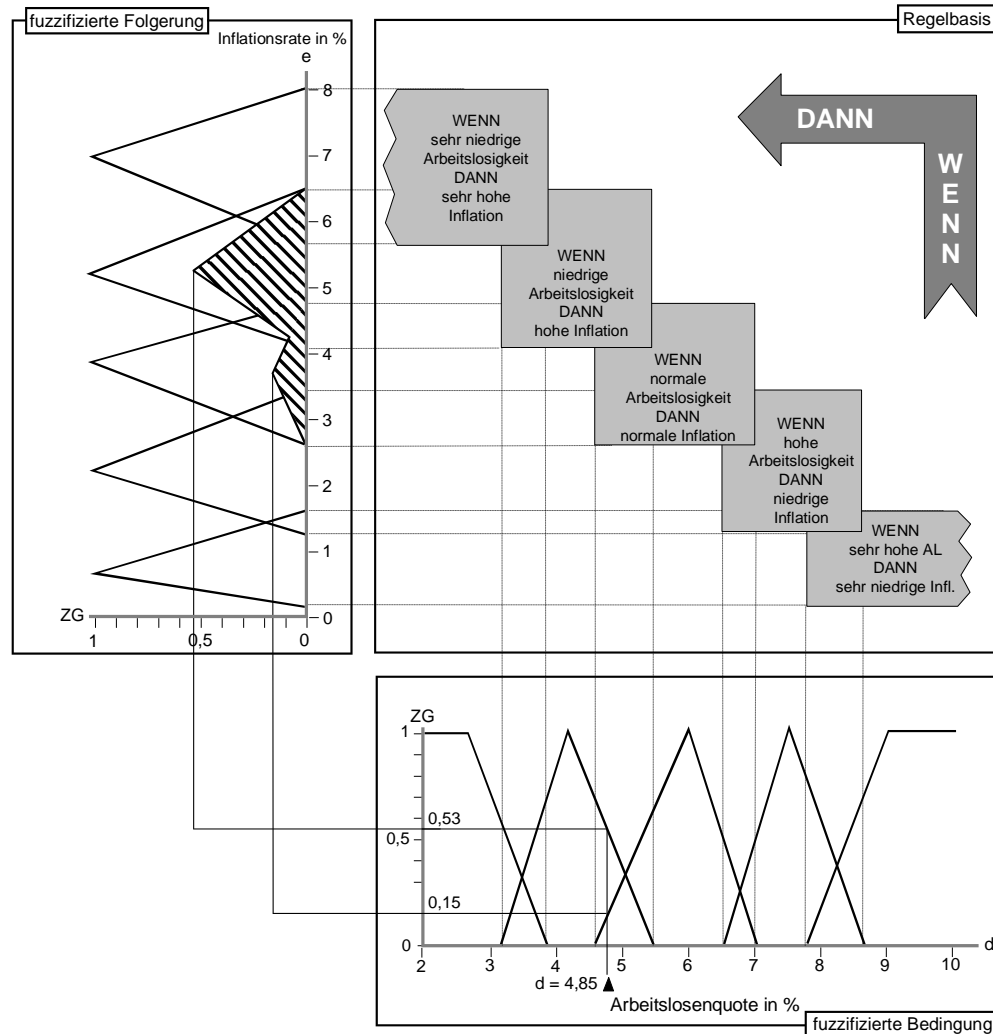
hat Einfluß auf ...



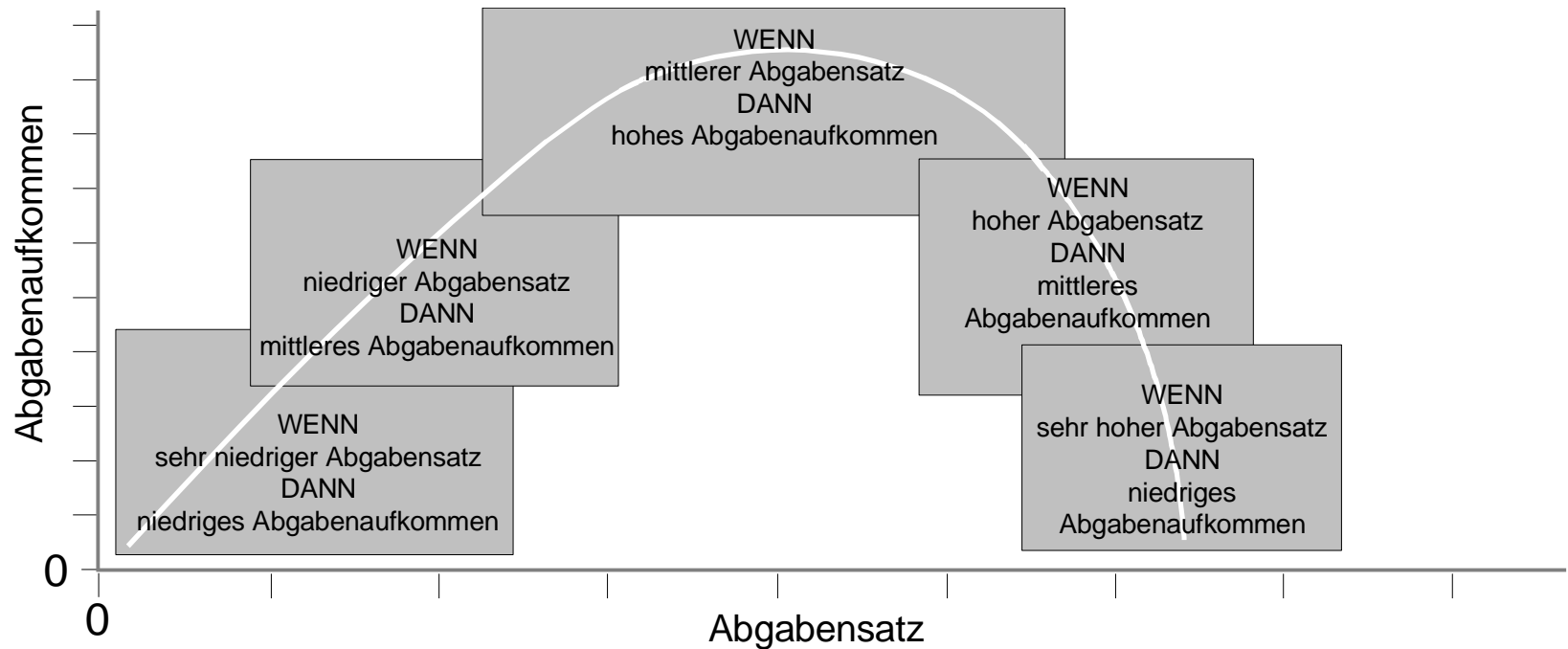
Übertragung des Applikationsgrades



Fuzzy-Regelbasis: Inflationsprognose



Fuzzy-Regelbasis: Laffer-Kurve (Abschätzung des Abgabenaufkommens)



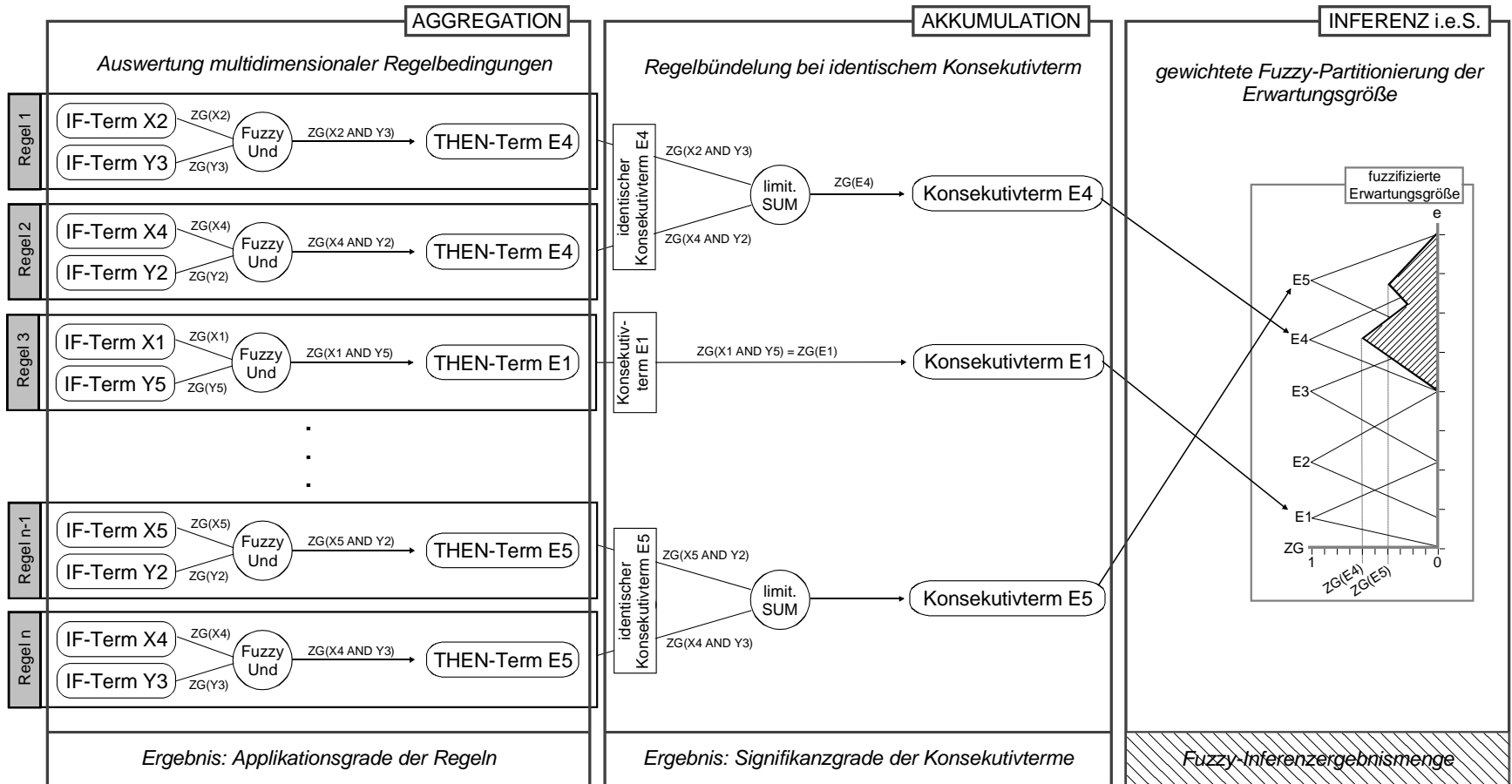
Fuzzy-Inferenz: Akkumulation

- verschiedene Regeln mit identischen Konsektivteil
- Signifikanzgrad = $\min(1, \text{Summe der Applikationsgrade})$

Fuzzy-Inferenz: Aggregation

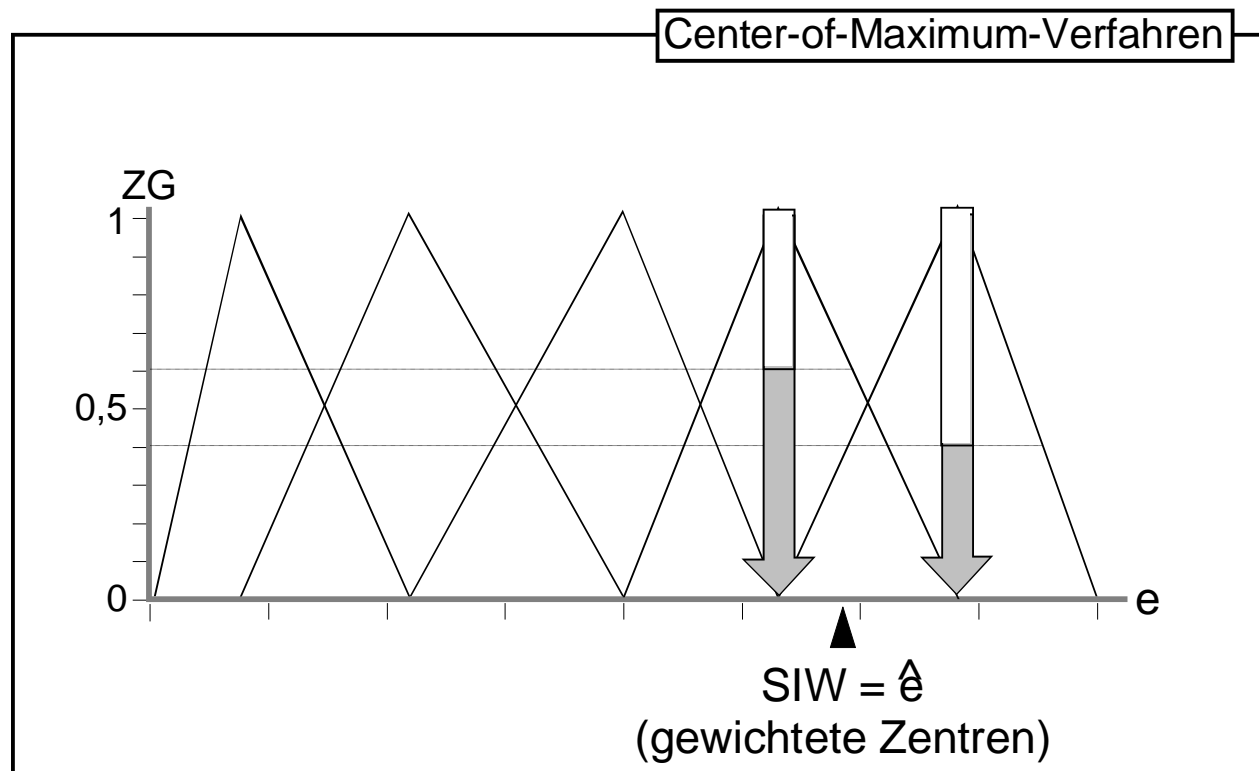
- Auswertung mehrdimensionaler Fuzzy-Regeln
- Fuzzy-Und
 - Minimum-Operator
 - Produkt-Operator

Fuzzy-Regelauswertung im Überblick



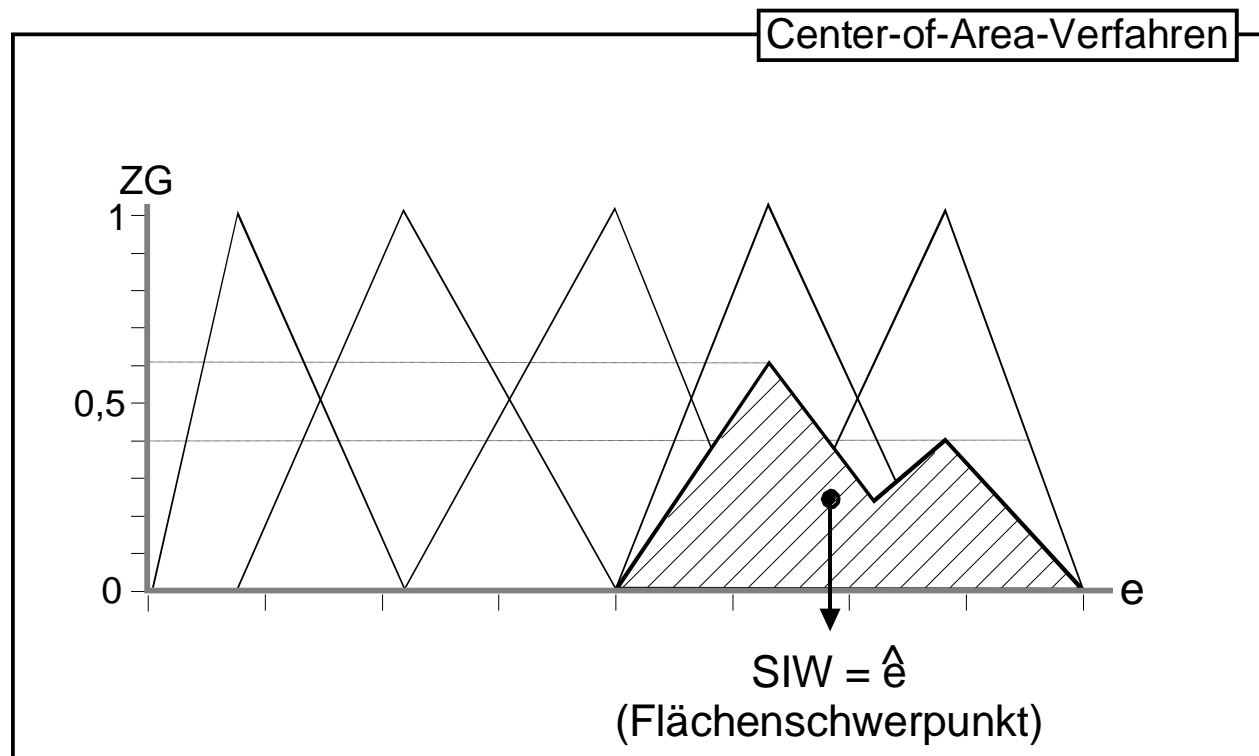
Defuzzifizierung

- Suche nach dem repräsentativen scharfen Wert der Outputbasisvariablen
- Center-of-Maximum-Verfahren

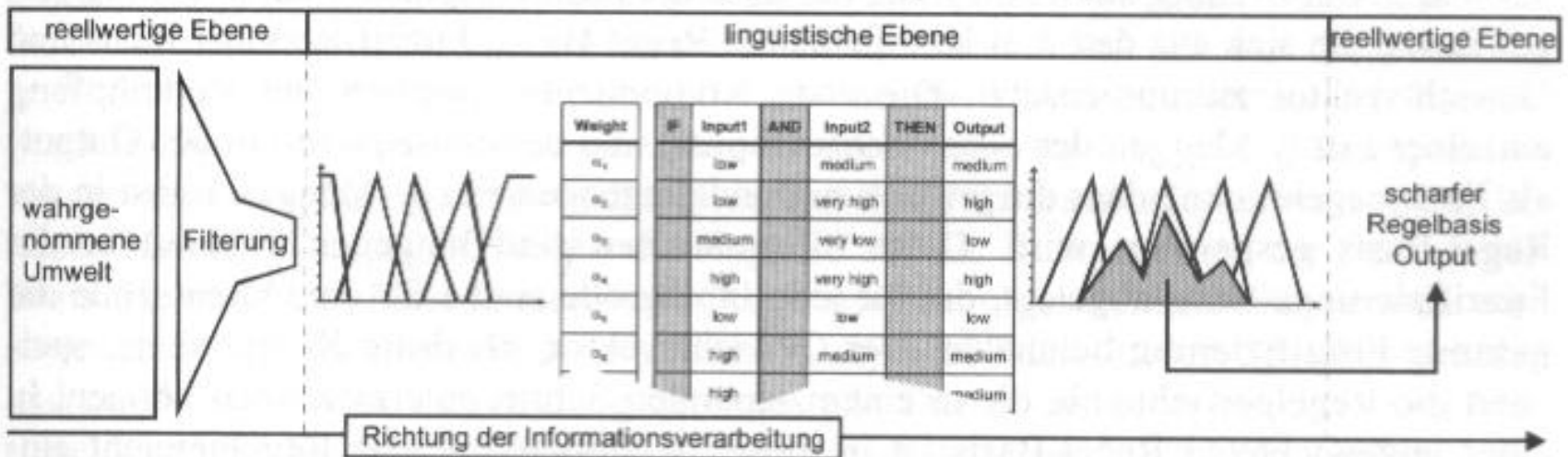


Defuzzifizierung

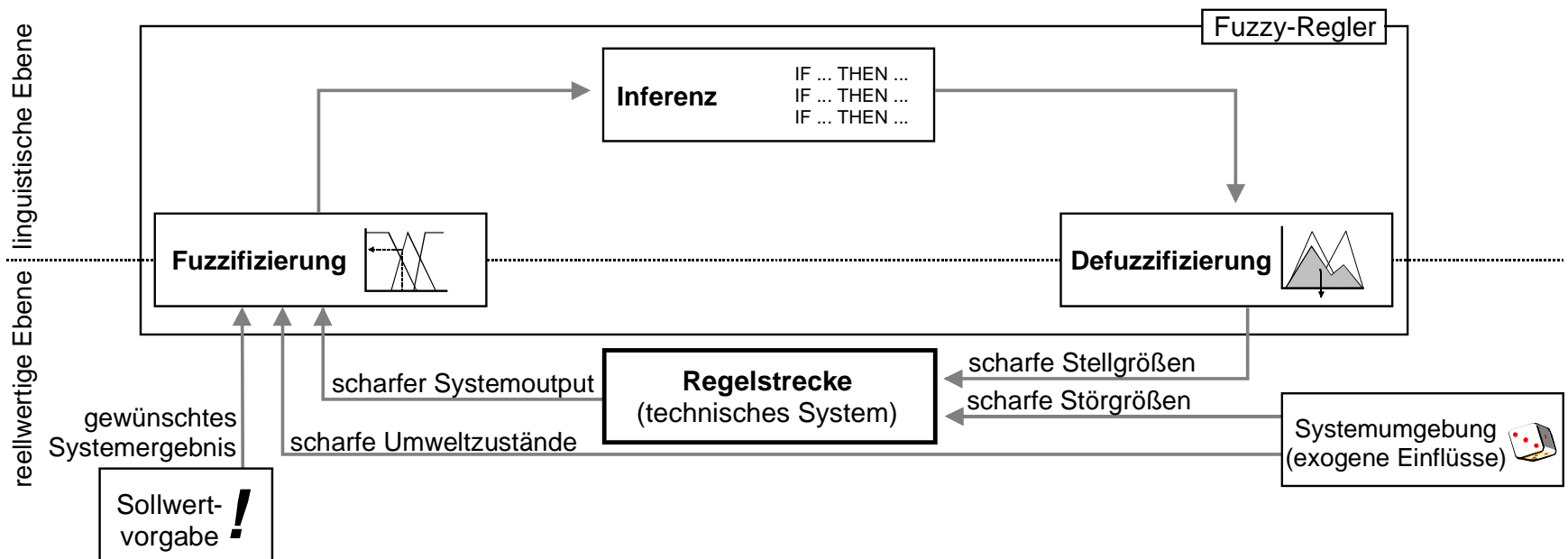
- Suche nach dem repräsentativem scharfen Wert der Outputbasisvariablen
- Center-of-Area-Verfahren



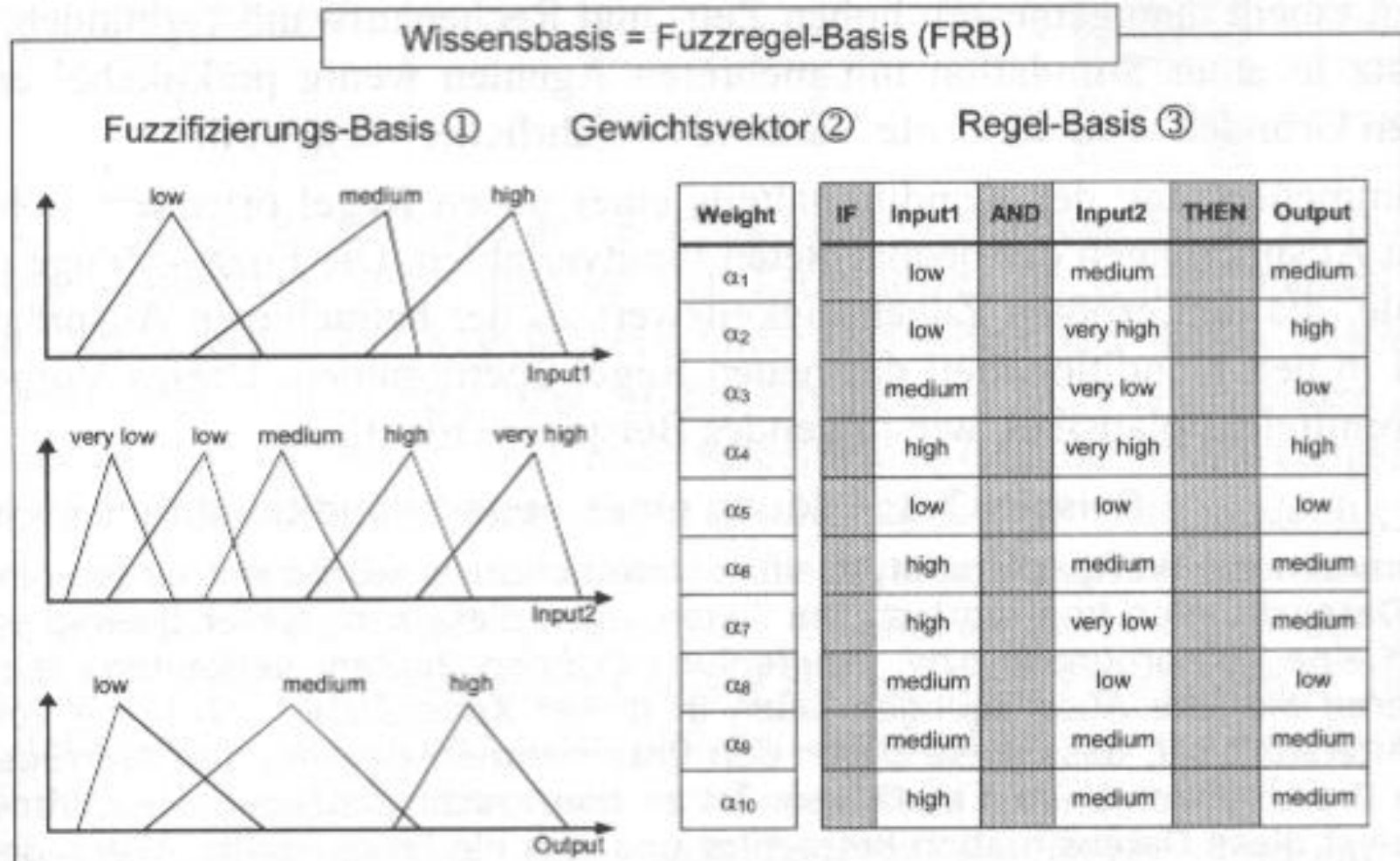
Fuzzy-Inferenzprozess



Fuzzy-Regelung (Fuzzy-Control)



Wissensbasis eines Regelsystems



GENEFER-Technologien

- Basistechnologie: Fuzzy-Systeme
- **Grundkonzept: Hybridisierung**
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- Tuning
 - genetisch
 - neuronal (Backprop)

Hybridisierung

Fuzzy-Systeme	KNN	GA	Fuzzy-GA	Neuro-Fuzzy
Vorteile				
<ul style="list-style-type: none"> • Umgang mit Unschärfe • explizite Wissensrepräsentation • Expertenwissen nutzbar • klare, einfache Architektur 	<ul style="list-style-type: none"> • kein Expertenwissen notwendig • verschiedene Lernalgorithmen • Umgang mit Unschärfe 	<ul style="list-style-type: none"> • robuster Such- und Optimierungsalgorithmus 	<ul style="list-style-type: none"> • Umgang mit Unschärfe • explizite Wissensrepräsentation • lernfähig • Expertenwissen nutzbar • klare Architektur 	
Nachteile				
<ul style="list-style-type: none"> • nicht lernfähig • Expertenwissen unbedingt notwendig 	<ul style="list-style-type: none"> • Black-Box Charakter • Expertenwissen nicht nutzbar • Netztopologie unklar • Konvergenz des Lernprozesses und Optimalität des Lernergebnisses nicht gegeben 	<ul style="list-style-type: none"> • kodierte und damit schwer interpretierbare Lösungsform • rechenintensiv, da populationsbasiert 	<ul style="list-style-type: none"> • rechenintensiv 	<ul style="list-style-type: none"> • Konvergenz des Lernprozesses und Optimalität des Lernergebnisses nicht gegeben

GENEFER-Technologien

- Basistechnologie: Fuzzy-Systeme
- Grundkonzept: Hybridisierung
- **Inputidentifizierung: Fuzzy-Curves/-Surfaces**
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- Tuning
 - genetisch
 - neuronal (Backprop)

Fuzzy-Curves/Fuzzy-Surfaces

Output y	Input x_1	Input x_2	Input x_3	...	Input x_{n_j}
y_1	$x_{1,1}$	$x_{2,1}$	$x_{3,1}$...	$x_{n_j,1}$
y_2	$x_{1,2}$	$x_{2,2}$	$x_{3,2}$...	$x_{n_j,2}$
...
y_{nk}	$x_{1,nk}$	$x_{2,nk}$	$x_{3,nk}$...	$x_{n_j,nk}$

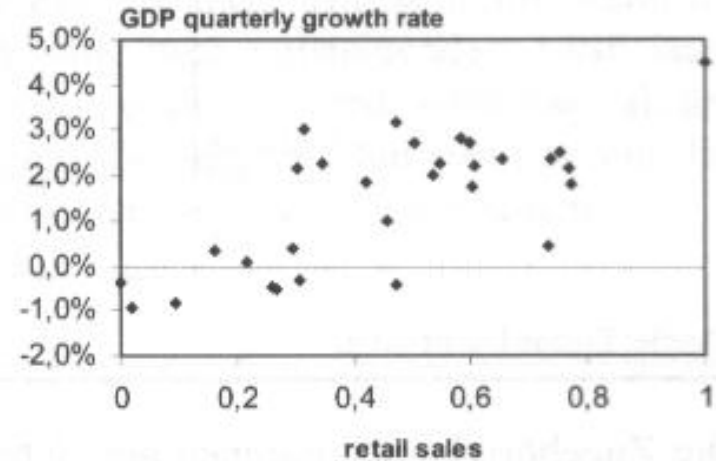
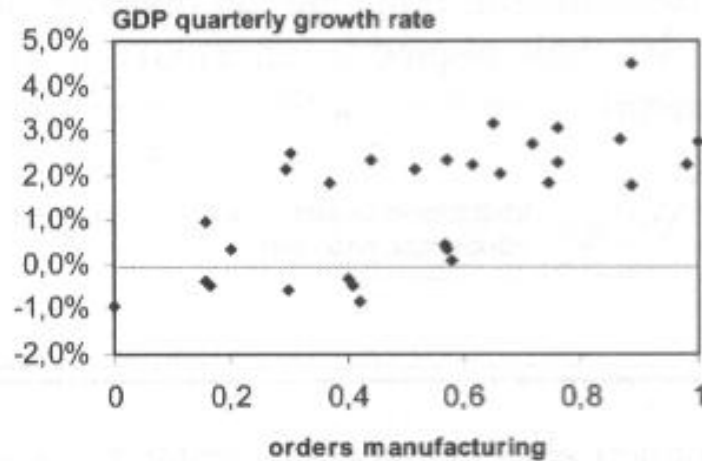
- Skalierung (Einheitsinterval)
- „Fuzzy-Menge“ zu jedem Datenpunkt
 - skaliert mit Outputwert
 - Interpretation als Fuzzy Regelbasis

Wenn $\bar{x}_j = \mu_{j,k}(\bar{x}_j)$ ist, dann ist der Output $y = y_k$.

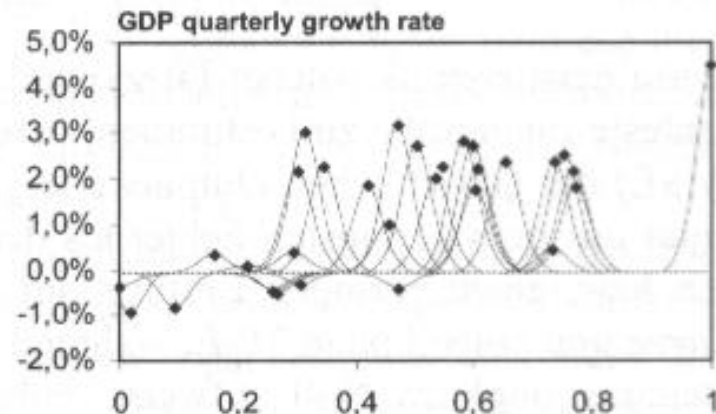
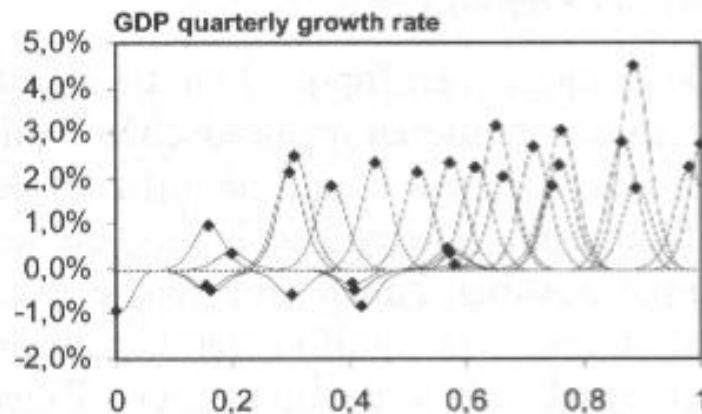
- Defuzzifizierung über alle Beobachtungen
- Ordnung der Input-Größen gemäß MSE

Fuzzy-Curves: Datenpunkt-Fuzzy-Mengen

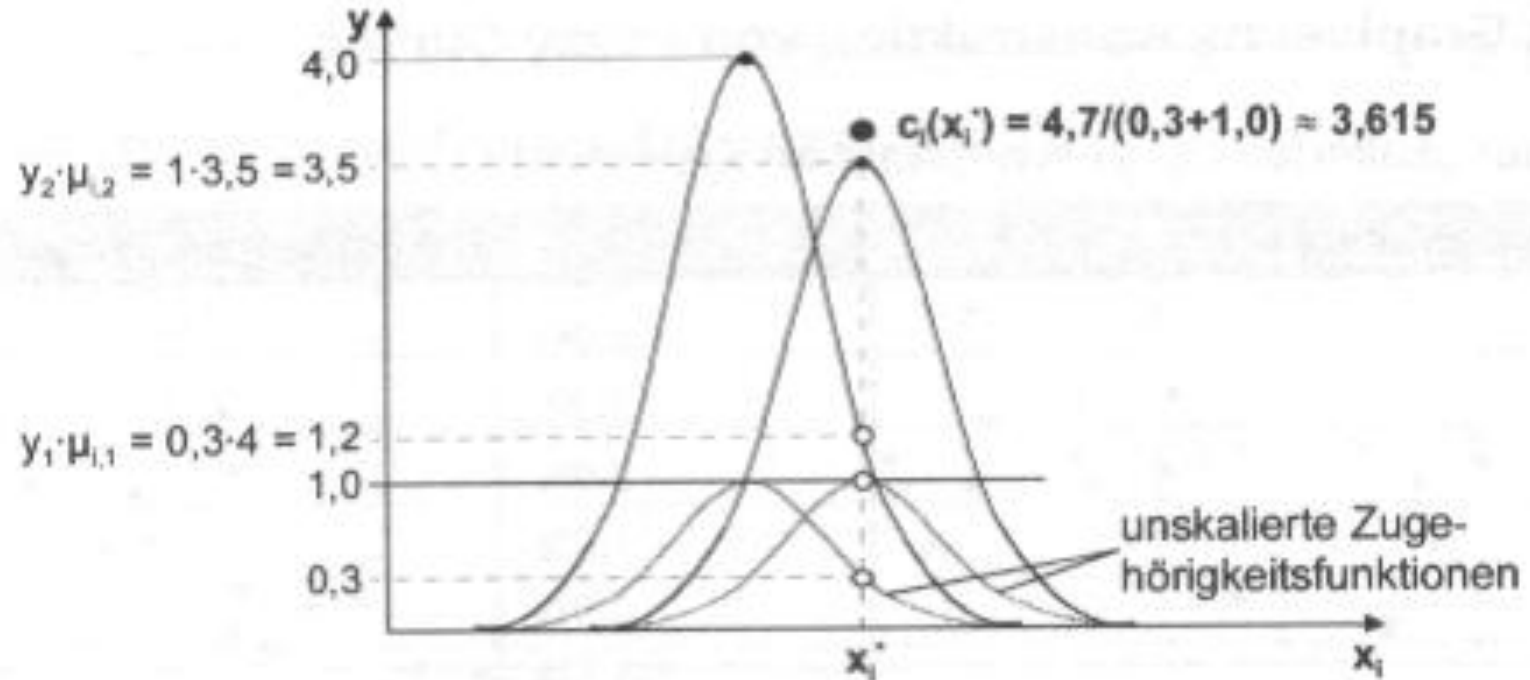
a) Punktwolken der skalierten (\bar{x}_i, y) -Wertpaare im \bar{x}_i/y -Raum



b) Fuzzifizierung der (\bar{x}_i/y) -Wertpaare



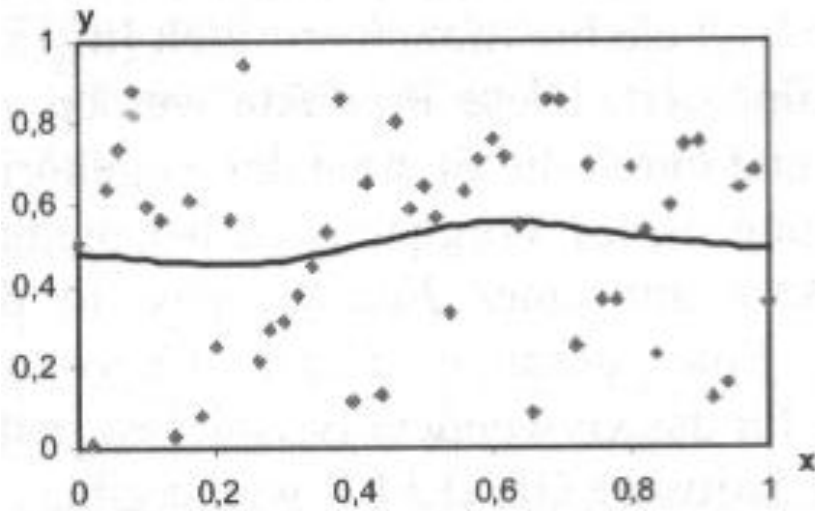
Fuzzy-Curves: Punktbestimmung



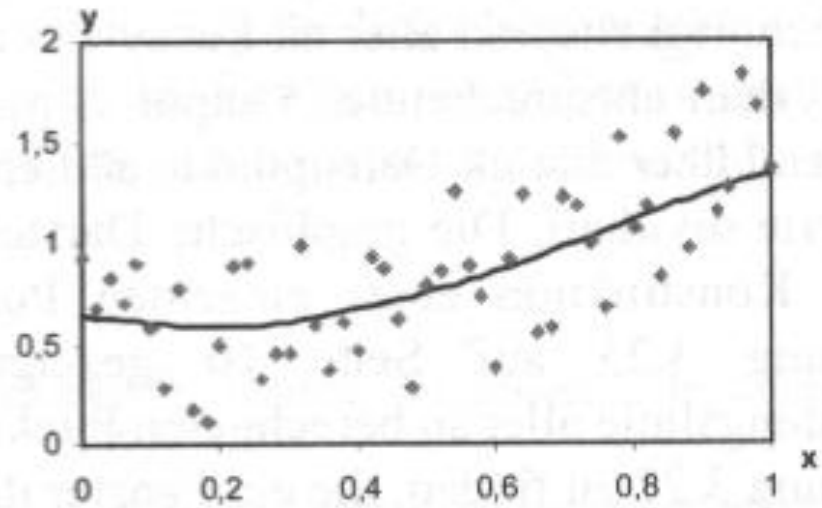
$$c_j(\bar{X}_j) = \frac{\sum_{k=1}^{nk} y_k \cdot \mu_{j,k}(\bar{X}_j)}{\sum_{k=1}^{nk} \mu_{j,k}(\bar{X}_j)}$$

Fuzzy-Curves: Synthetisches Beispiel

$$y = \text{random}(x)$$



$$y = x^2 + \text{random}(x),$$



Fuzzy-Surfaces

- Problem abhängiger Inputs ("Kolinearität")
- Ausdehnung auf zweidimensionale Betrachtung (Inputpaare)
- Fuzzy-Regel:

Wenn $x_j = \mu_{j,k}(x_j)$ und $x_h = \mu_{h,k}(x_h)$ ist, dann ist der Output $y = y_k$

- defuzzifizierter Schätzwert (Punkt auf Fuzzy-Surface)

$$S_{j,h}(\bar{x}_j, \bar{x}_h) = \frac{\sum_{k=1}^{nk} y_k \cdot \mu_{j,k}(\bar{x}_j) \cdot \mu_{h,k}(\bar{x}_h)}{\sum_{k=1}^{nk} \mu_{j,k}(\bar{x}_j) \cdot \mu_{h,k}(\bar{x}_h)}$$

FC-FS-Algorithmus: Pseudocode

```
01      Generate Fuzzy Curves for all possible inputs
02      Rank inputs according to ascending MSE
03      Identify first input as the one with the lowest MSE
04      Eliminate  $\alpha\%$  of the inputs with the largest MSE
05      while there are still remaining inputs do
06          begin
07              Create Fuzzy Surfaces using the last identified input with all remaining ones
08              Rank according to MSE
09              Identify next input as the one with lowest MSE
10              Eliminate  $\alpha\%$  of the inputs with the largest MSE
11          end;
```


GENEFER-Technologien

- Basistechnologie: Fuzzy-Systeme
- Grundkonzept: Hybridisierung
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- **Regelgenerierung**
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- Tuning
 - genetisch
 - neuronal (Backprop)

Evolutionäre Regelgenerierung: Kriterien

- Completeness: Covering-Value $CV(d_t) \geq \varepsilon > 0$

$$A_i(in_t) = \min(A_{i,1}(in_{1,t}), \dots, A_{nf,nj}(in_{nj,t})) \quad \text{Compatibility Degree: antecedents}$$

$$FR_i(d_t) = \min(A_i(in_t), B_i(out_t)) \quad \text{Compatibility Degree: rule}$$

$$CV(d_t) = \sum_{i=1}^{mi} FR_i(d_t) \quad \text{Covering Value}$$

- Consistency: Kompatibilitätsgrad
 - Konzept der positiven und negativen Beobachtungspunkte

$$TDS_{p,\xi}^+(FR_i) = \{ d_p \in TDS_p \mid FR_i(d_p) \geq \xi \}$$

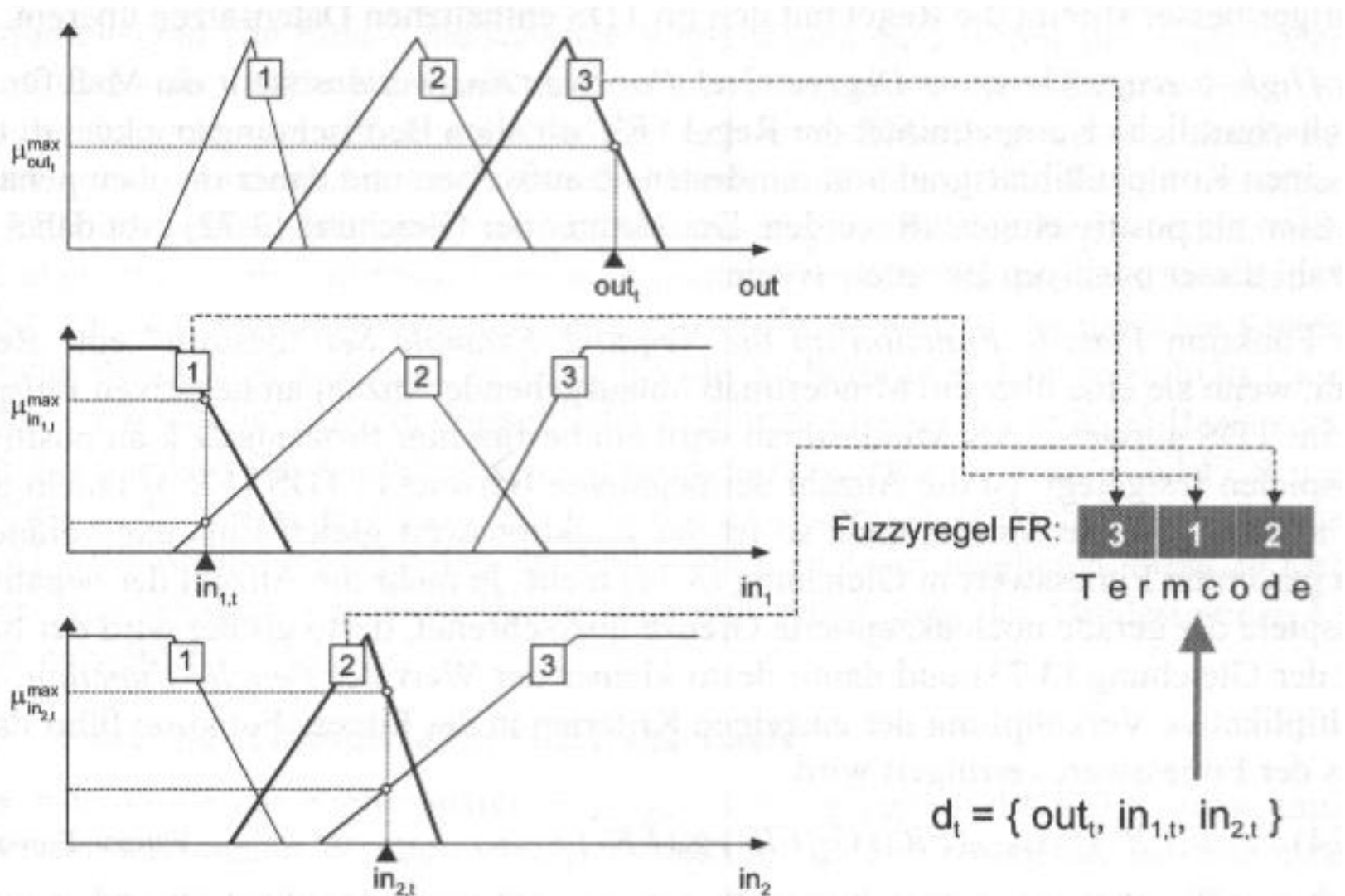
$$TDS_p^-(FR_i) = \{ d_p \in TDS_p \mid FR_i(d_p) = 0 \text{ und } A_i(in_p) > 0 \}$$

- $k = |TDS^-| / |TDS^+|$
- Gauss-ZG: Mindestzugehörigkeitswert $FR_i(d_p) \leq \hat{\rho}$

Evolutionäre Regelgenerierung: Pseudocode

```
01 begin
02   Initialize:
    (a) Set  $k$ ,  $\varepsilon$  and  $\xi$ 
    (b) Set training data set TDS with  $p = 1, \dots, n_p$  as index
    (c) Set Covering Value to zero for all examples in the TDS:  $CV(d_p) = 0$ 
    (d) Initialize generated Fuzzy Rule Set  $G\_FRS$  to empty
03   while not TDS =  $\emptyset$  do
04     {Rule Generating Loop}
05     begin
06       Initialize the candidate fuzzy rule set  $C\_FRS$  to empty
07       Create the best covering Fuzzy Rule  $FR^C$  for every example in the TDS and add it
        to  $C\_FRS$ , if  $FR^C \notin C\_FRS$ .
08       Evaluate all  $FR^C$  in  $C\_FRS$  and select the one with the highest fitness value
09       Copy the selected best Fuzzy Rule  $FR^{best}$  to the generated Fuzzy Rule Set  $G\_FRS$ 
10       for every  $d_p \in TDS$  do
11         {Covering Loop}
12         begin
13            $CV(d_p) = CV(d_p) + FR^{best}(d_p)$ 
14           if  $CV(d_p) \geq \varepsilon$  then remove  $d_p$  from TDS
15         end;
16     end;
17 end;
```

Evolutionäre Regelgenerierung: Beispiel



Evolutionäre Regelgenerierung: Fitness-Bewertung

$$F(FR^C) = \psi_{TDS}(FR^C) \cdot G_{\xi}(FR^C) \cdot g_n(FR^C)$$

 Fitness-Funktion

$$\psi_{TDS}(FR^C) = \frac{1}{np} \cdot \sum_{p=1}^{np} FR^C(d_p)$$

High Frequency Value

$$G_{\xi}(FR^C) = \sum_{d_p \in TDS_{\xi}^+(FR^C)} \frac{FR^C(d_p)}{|TDS_{\xi}^+(FR^C)|}$$

High Average Covering Degree
Over Positive Observations

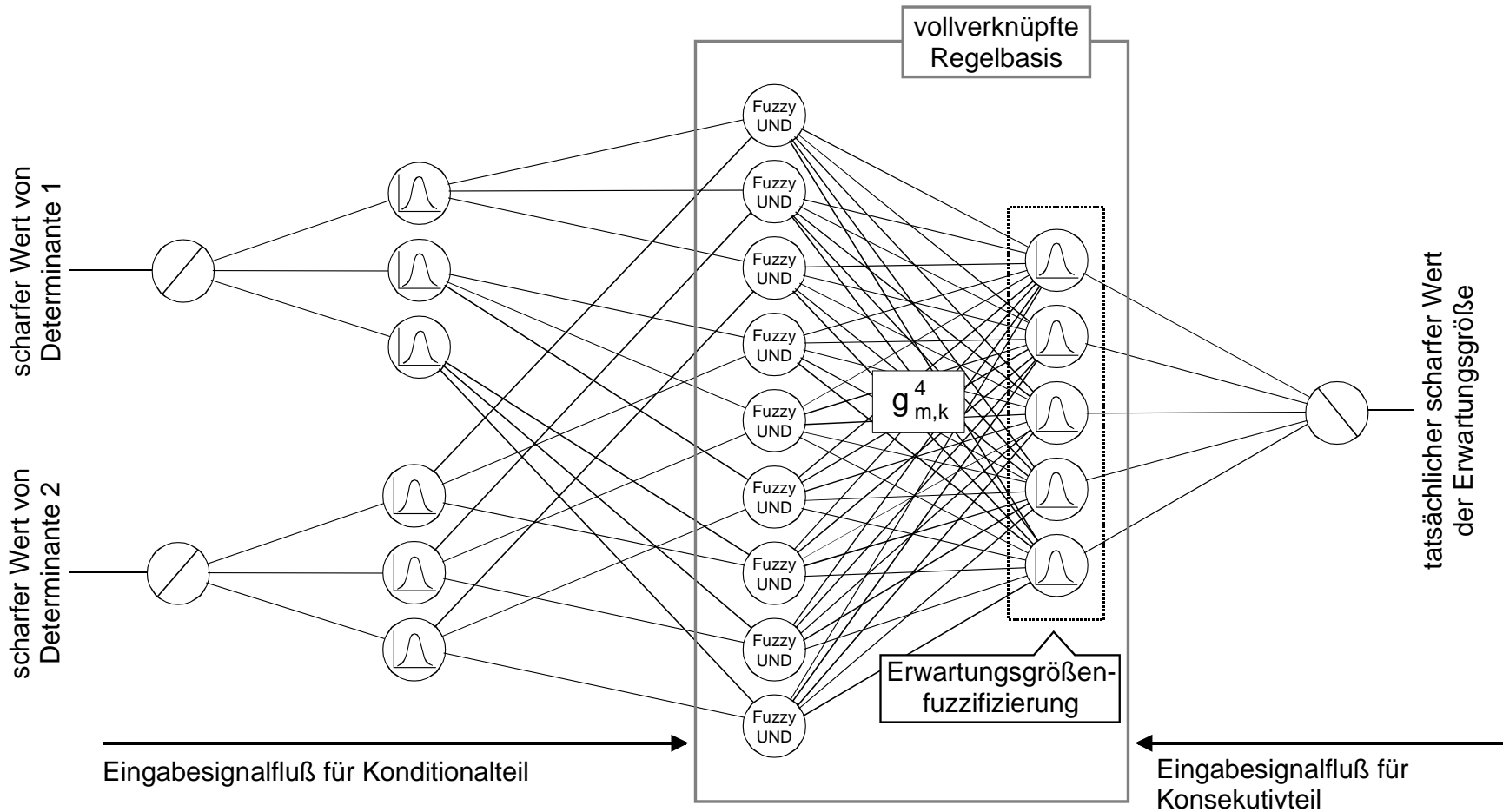
$$g_n(FR^C) = \begin{cases} 1 & \text{falls } |TDS^-(FR^C)| \leq k \cdot |TDS_{\xi}^+(FR^C)| \\ \frac{1}{|TDS^-(FR^C)| - k \cdot |TDS_{\xi}^+(FR^C)|} & \text{sonst} \end{cases}$$

Penalty Function
on the Negative
Example Set

Evolutionäre Regelgenerierung: Regelanzahl

- Ähnlichkeit Beobachtungspunkte
- Granularität der Fuzzifizierung
- Mindestabdeckung (ε)

Neuronale Regelgenerierung (Wettbewerbslernen)



GENEFER-Technologien

- Basistechnologie: Fuzzy-Systeme
- Grundkonzept: Hybridisierung
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- **Simplifizierung: genetisch**
- Tuning
 - genetisch
 - neuronal (Backprop)

Simplifizierung der Rule Base: Ist weniger mehr?

- Nebenbedingung: Completeness!

- CV einer RB in Bezug auf Trainingsbeispiel

$$CV(d_t) = \bigcup_{i=\{1,\dots,ni\}} FR_i(d_t) \geq \tau$$

- CVs einer RB in Bezug auf Trainingsmenge
TSCD = Training-Set Completeness Degree

$$TSCD(C_{pi}, TDS) = \bigcap_{d_t \in TDS} CV_{C_{pi}}(d_t)$$

- Fitnessfunktion

$$F(C_{pi}) = \begin{cases} MSE(C_{pi}) & \text{falls } TSCD(C_{pi}, TDS) \geq \tau \\ \frac{1}{2} \cdot \sum_{d_t \in TDS} (out_t)^2 & \text{sonst.} \end{cases}$$

Simplifizierungsalgorithmus: Pseudocode

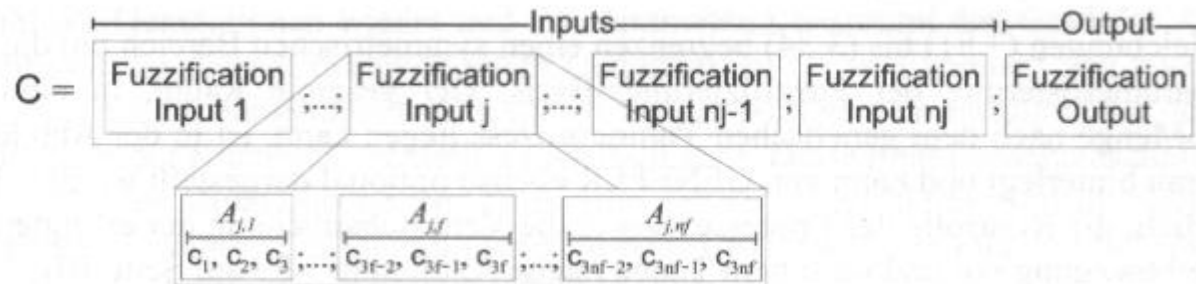
```
01      Create Initial encoded Rule-Base Population
02      Evaluate all individuals
03      for GenerationCounter = 1 to Termination Generation do
04          begin
05              Create Mating Pool
06              for Individual = 1 to PopulationSize/2 do
07                  begin
08                      Randomly choose 2 individuals
09                      Apply Crossover and Mutation operation and create 2 offspring
10                      Copy both offspring in the next generation population
11                  end;
12              Evaluate all individuals
13      end;
```

GENEFER-Technologien

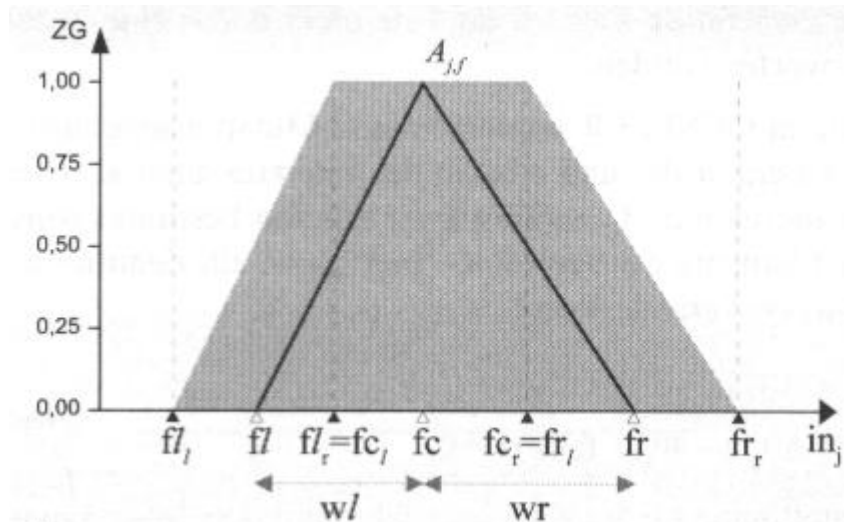
- Basistechnologie: Fuzzy-Systeme
- Grundkonzept: Hybridisierung
- Inputidentifizierung: Fuzzy-Curves/-Surfaces
- Fuzzifizierung: Wettbewerbslernen
- Regelgenerierung
 - evolutionär/induktiv
 - neuronal (Wettbewerbslernen)
- Simplifizierung: genetisch
- **Tuning**
 - genetisch
 - neuronal (Backprop)

Tuning der Fuzzification Base: Genetisch 1

- Reellwertige Kodierung



- Beschränkung des Suchraums



$$fl_l = fl - 0,5 \cdot (fc - fl)$$

$$fl_r = fc_l = fc - 0,5 \cdot (fc - fl)$$

$$fc_r = fr_l = 0,5 \cdot (fr - fc)$$

$$fr_r = fr + 0,5 \cdot (fr - fc)$$

Tuning der Fuzzification Base: Genetisch 2

- Crossover Operatoren

$$C_1^{t+1} = a \cdot C_{P1}^t + (1 - a) \cdot C_{P2}^t$$

$$C_2^{t+1} = a \cdot C_{P2}^t + (1 - a) \cdot C_{P1}^t$$

$$C_3^{t+1} \text{ mit } c_3^{t+1} = \min(c_{P1}^t, c_{P2}^t) \quad \forall c_{P1}^t \in C_{P1}^t \wedge c_{P2}^t \in C_{P2}^t$$

$$C_4^{t+1} \text{ mit } c_4^{t+1} = \max(c_{P1}^t, c_{P2}^t) \quad \forall c_{P1}^t \in C_{P1}^t \wedge c_{P2}^t \in C_{P2}^t$$

- Mutationsoperator

$$c' = \begin{cases} c + \Delta(t, fc_r - c) & \text{falls } \alpha = 1 \\ c - \Delta(t, c - fc_l) & \text{falls } \alpha = 0 \end{cases}$$

mit:

$$\Delta(t, y) = y \cdot (1 - r^{(1-vT)^z})$$

Tuning der Fuzzification Base: Neuronal (MBP)

