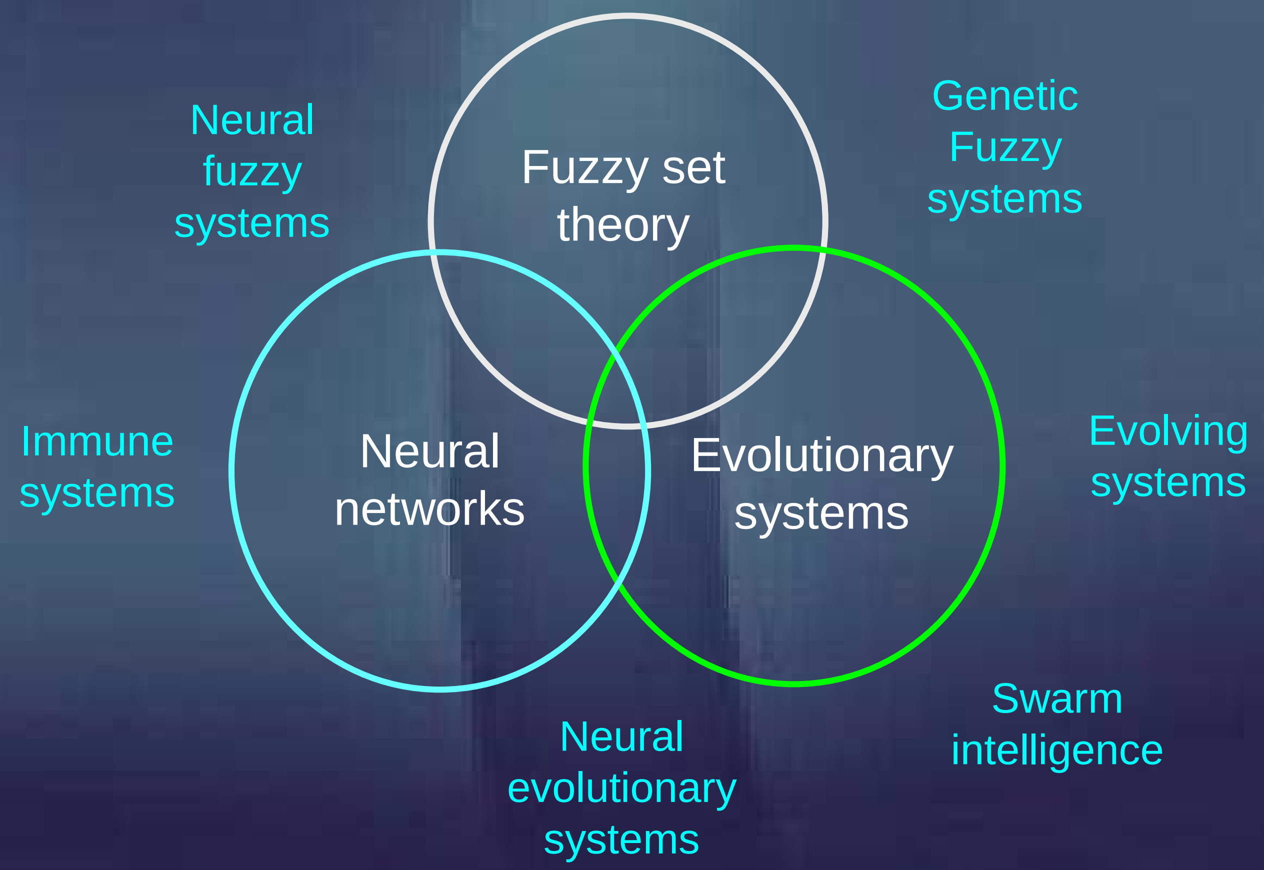
13 Fuzzy Systems and Computational Intelligence

Fuzzy Systems Engineering
Toward Human-Centric Computing

Contents

13.1 Computational intelligence

Computational Intelligence

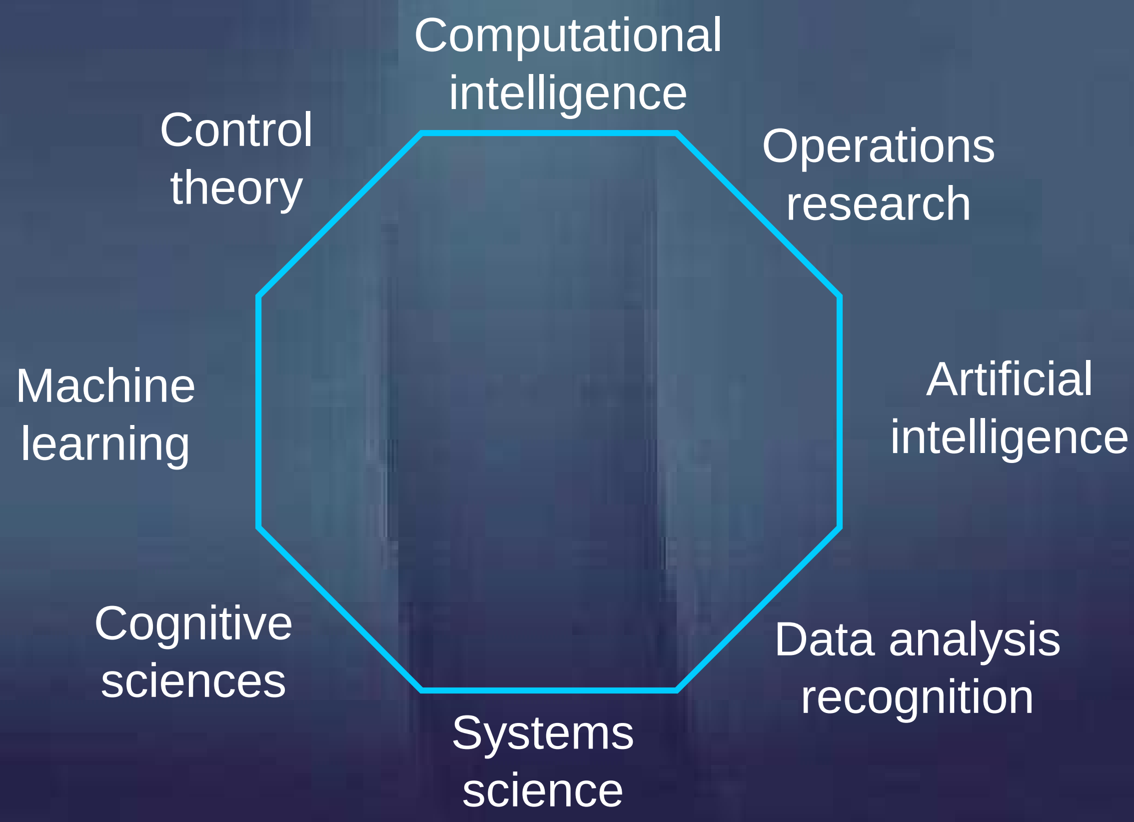


Computational intelligence

- Data processing systems with capabilities of (Bezdek, 1992/1994)
  - pattern recognition
  - adaptive
  - fault tolerance
  - performance approximates human performance
  - no use of explicit knowledge
- Framework to design and analyze intelligent systems (Duch, 2007)
  - autonomy
  - learning
  - reasoning

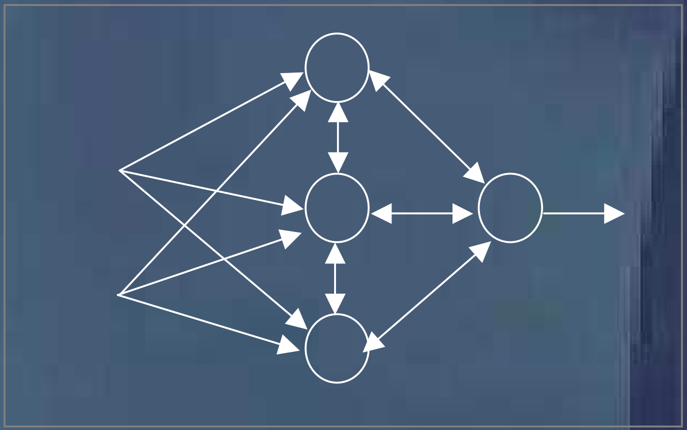
- Computing systems able to (Eberhart, 1996)
  - learn
  - deal with new situations using
    - reasoning
    - generalization
    - association
    - abstraction
    - discovering
- Computational intelligence
  - largely human-centered
  - forms of artificial and synthetic intelligence
  - collaboration man-machine

Intelligent Systems

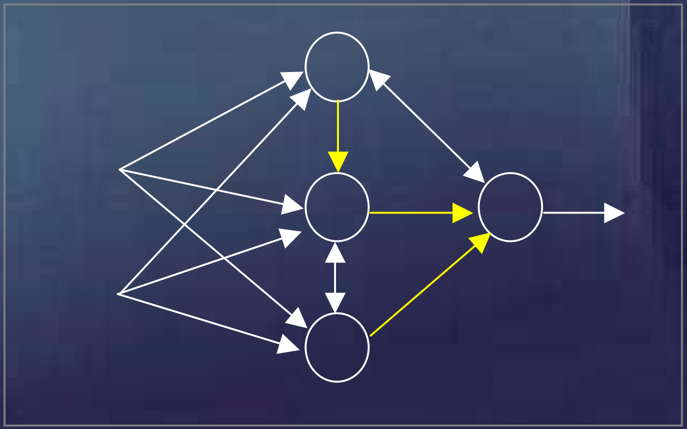


13.2 Recurrent neurofuzzy systems

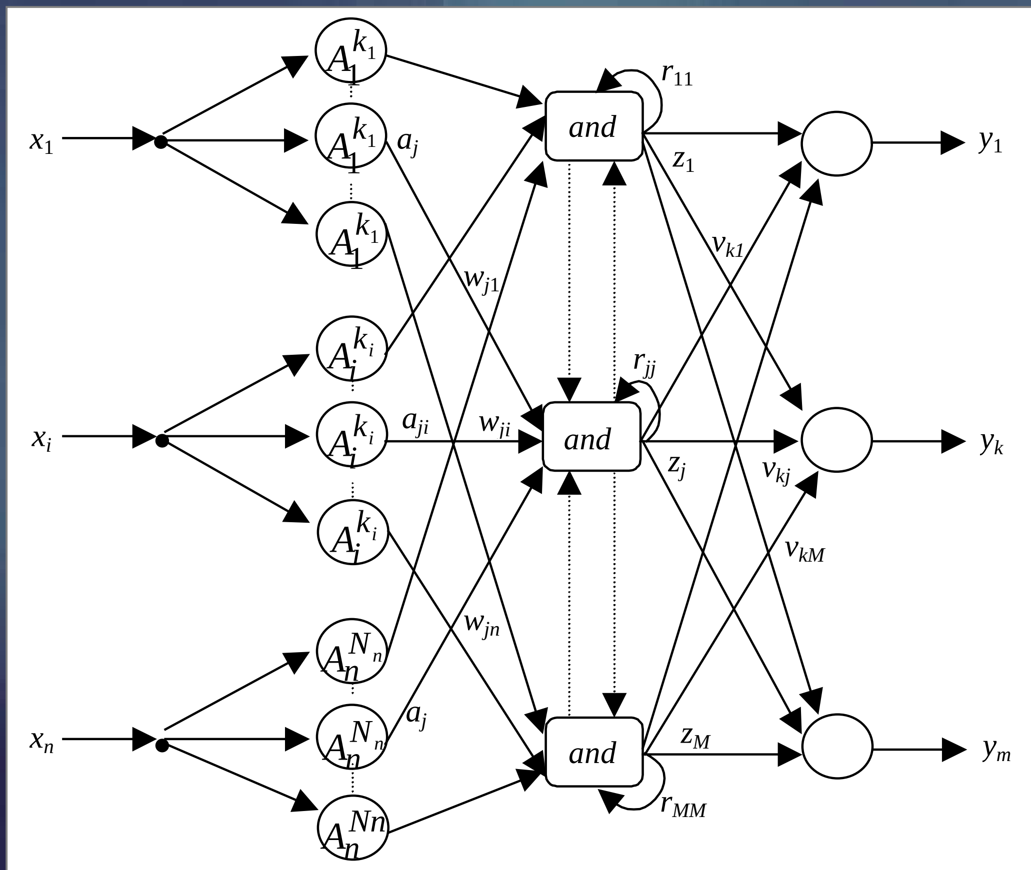
- Globally recurrent
  - – full feedback connections



- Partially recurrent
  - – partial feedback connections

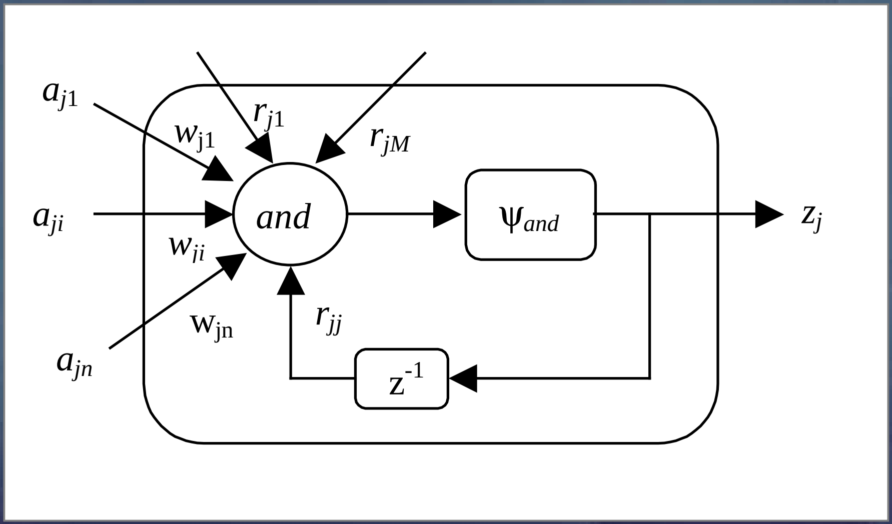


Recurrent neural fuzzy network model



Pedrycz and Gomide, FSE 2007

Recurrent ***and*** fuzzy neuron



$$z_j = \mathop{T}_{i=1}^{n+M} (w_{ji} \, s \, a_{ji})$$

z_j = AND($\mathbf{a}_j$;$\mathbf{w}_j$)

- Ni number of fuzzy sets that granulate the ith input
- j indexes *and* neurons; given k_i , j is found using

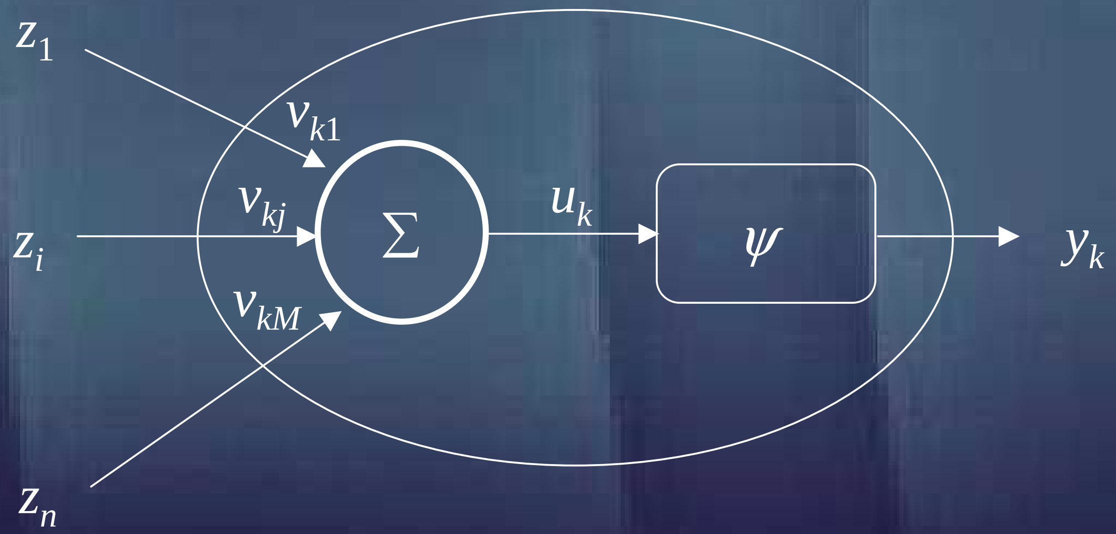
$$j = k_n + \sum_{i=2}^{M} (k_{(n-i+1)} - 1) \left(\prod_{r=1}^{i-1} N_{(n+1-r)} \right)$$

- $x_1, \ldots, x_i, \ldots, x_n$ inputs

- $a_{ji} = A_i^{k_i}(x_i)$

- w_{ji} weight between ith input and jth *and* neuron
- z_j output of the jth *and* neuron
- v_{kj} weight jth input of the kth output neuron
- r_{jl} feedback connection of the lth input of the jth *and* neuron
- $\mathrm{y}_{\mathrm{k}} = \psi(u_k)$ output kth neuron of the output layer

Output layer neuron



$$y = \psi(u_k) = \psi\left(\sum_{j=1}^{M} v_{kj} z_j\right)$$

Learning algorithm

procedure NET-LEARNING (**x**, **y**) **returns** a network

input:data **x**, **y**
local: fuzzy sets
t, *s*: triangular norms
ε: threshold

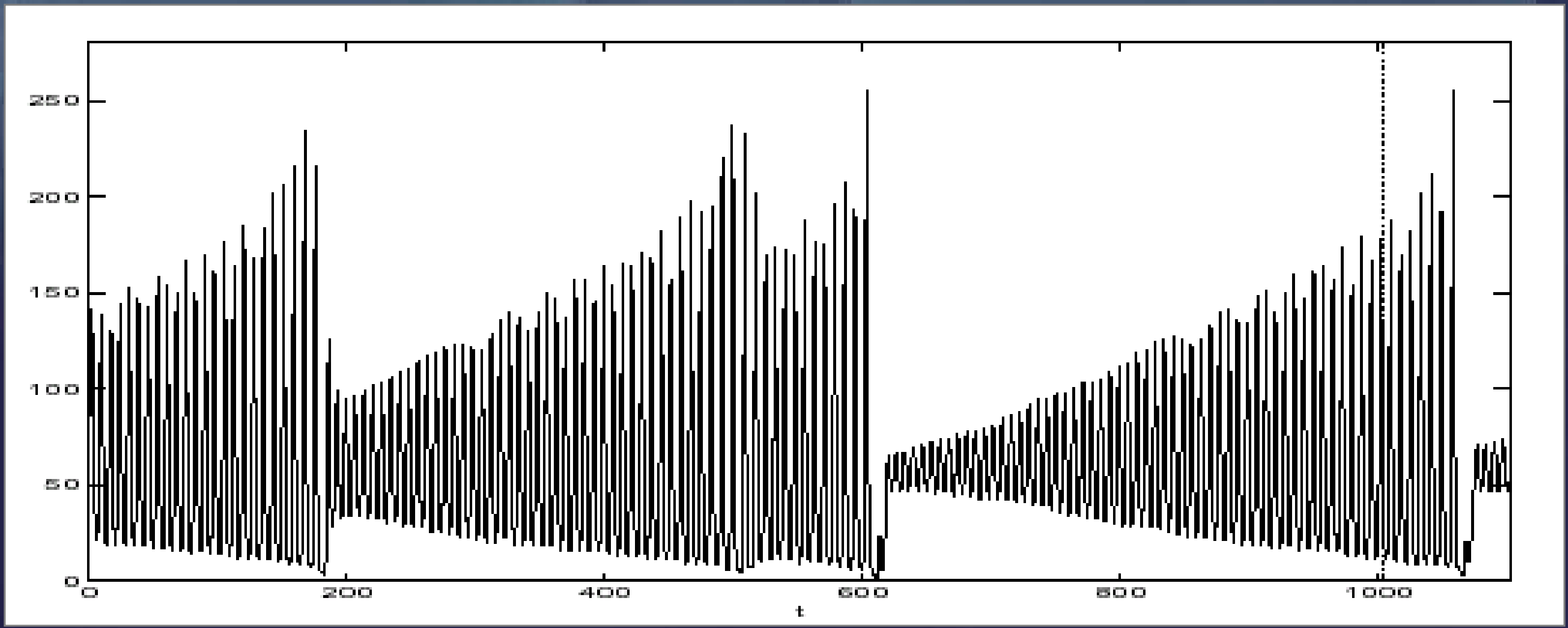
GENERATE-MEMBERSHIP-FUNCTIONS
INITIALIZE-NETWORK-WEIGHTS

until stop criteria $\leq \varepsilon$ **do**
choose and input-output pair *x* and *y* of the data set
ACTIVE-AND-NEURONS
ENCODING
UPDATE-WEIGHTS
return a network

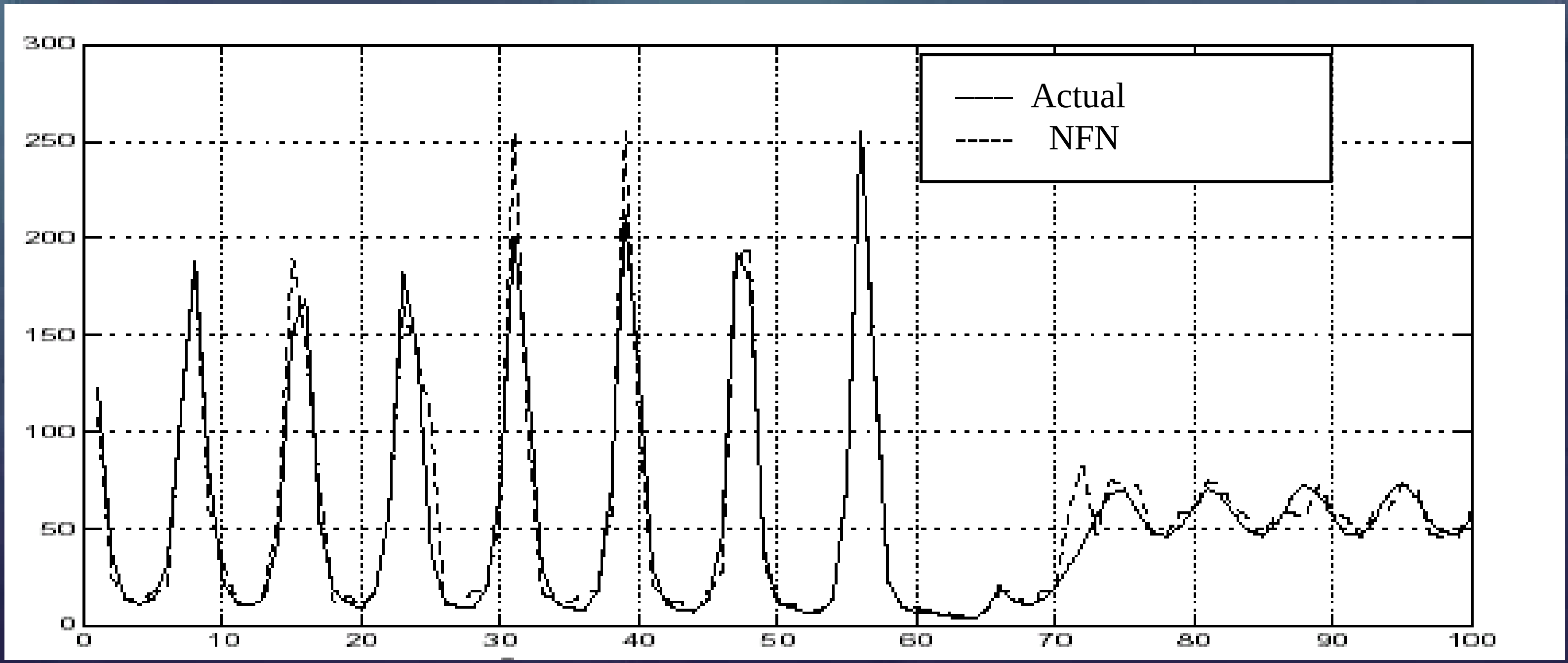
Example

Chaotic NH3 laser time series data

- first 1000 samples for learning
- predict next 100 steps



100 steps ahead prediction



Normalized squared forecasting errors (NSE) NH3 laser time series

Model	1 step ahead	100 steps ahead
FIR	0.0230	0.0551
NFN	0.0139	0.0306

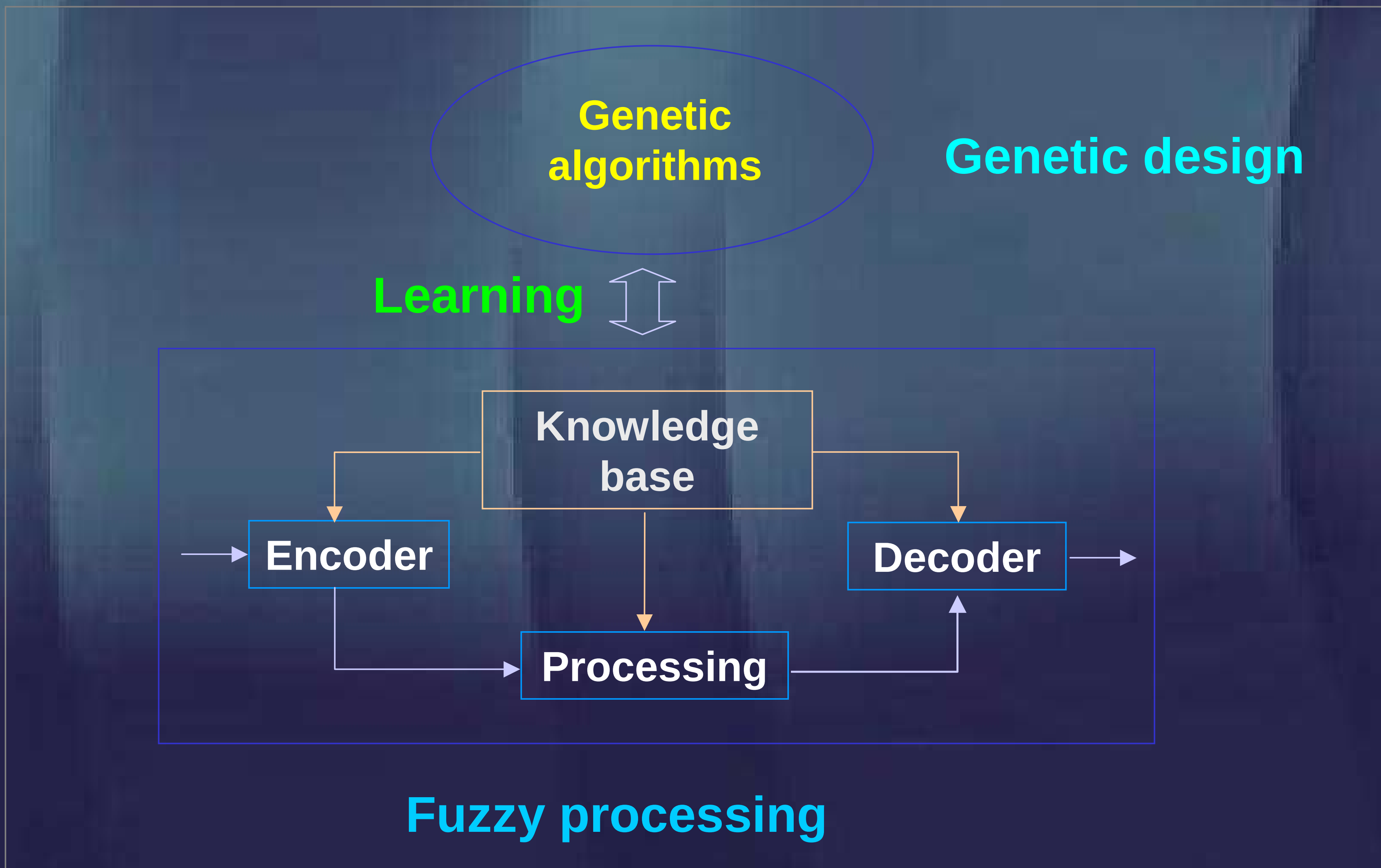
$$NSE = \frac{1}{\sigma^2 N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

13.3 Genetic fuzzy systems

Genetic fuzzy systems

- GFS is an approach to design fuzzy models and systems
- GFS = fuzzy system + learning using genetic algorithm
- Learning of models structure and parameters
  - rule base
  - fuzzy rules
  - membership functions
  - operators
  - inference procedures

Genetic Fuzzy Systems

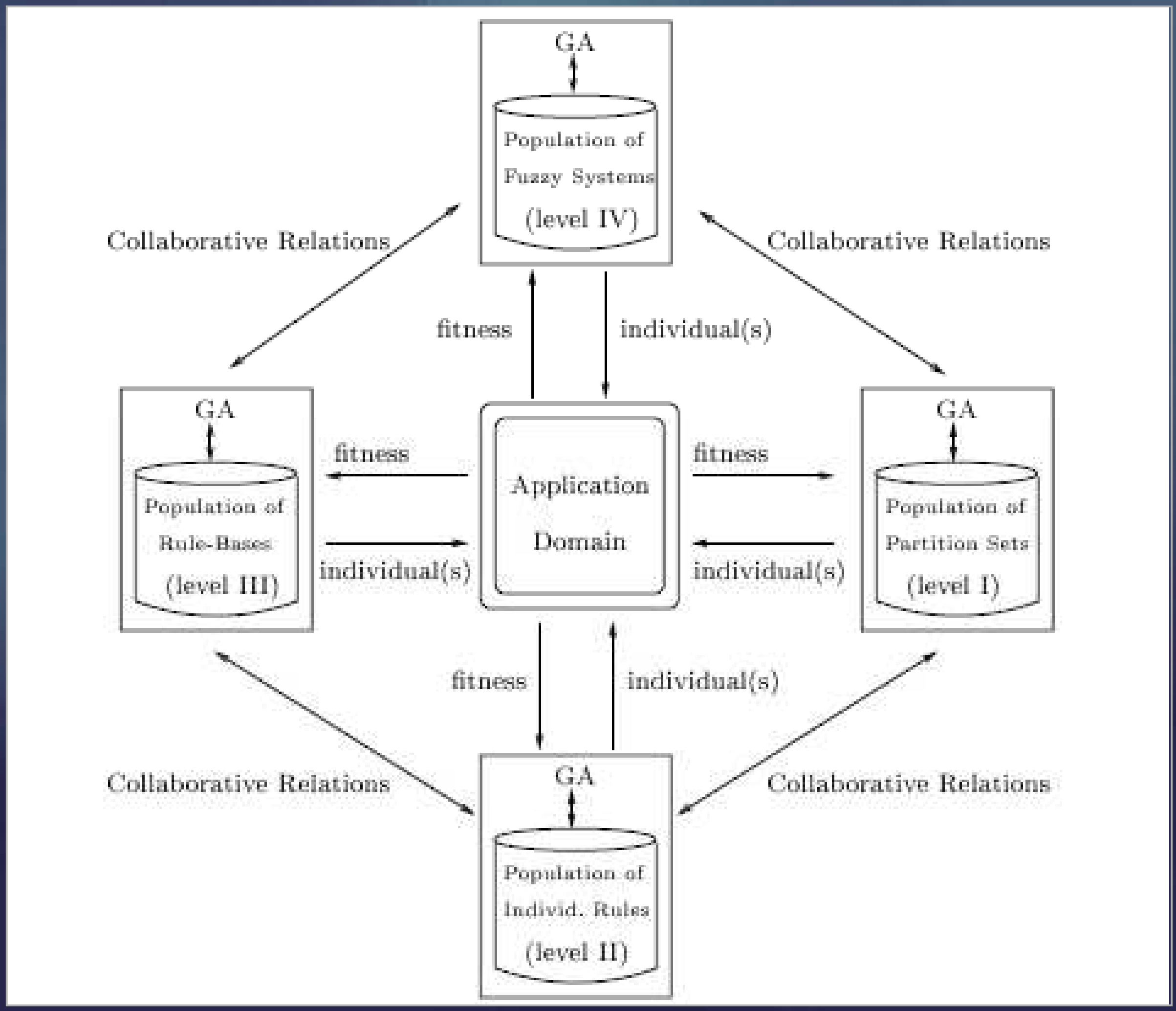


13.4 Coevolutionary hierarchical genetic fuzzy system

Coevolution

- Considers interactions between population members
- Populations hierarchically structured
- Hierarchy levels associated with partial solutions of the problem
  - individuals of different populations keep collaborative relations
  - collaboration depends on the fitness of the individuals
  - hierarchical levels:
    - I : membership functions
    - II : fuzzy rules
    - III: rule bases
    - IV: fuzzy systems (models)

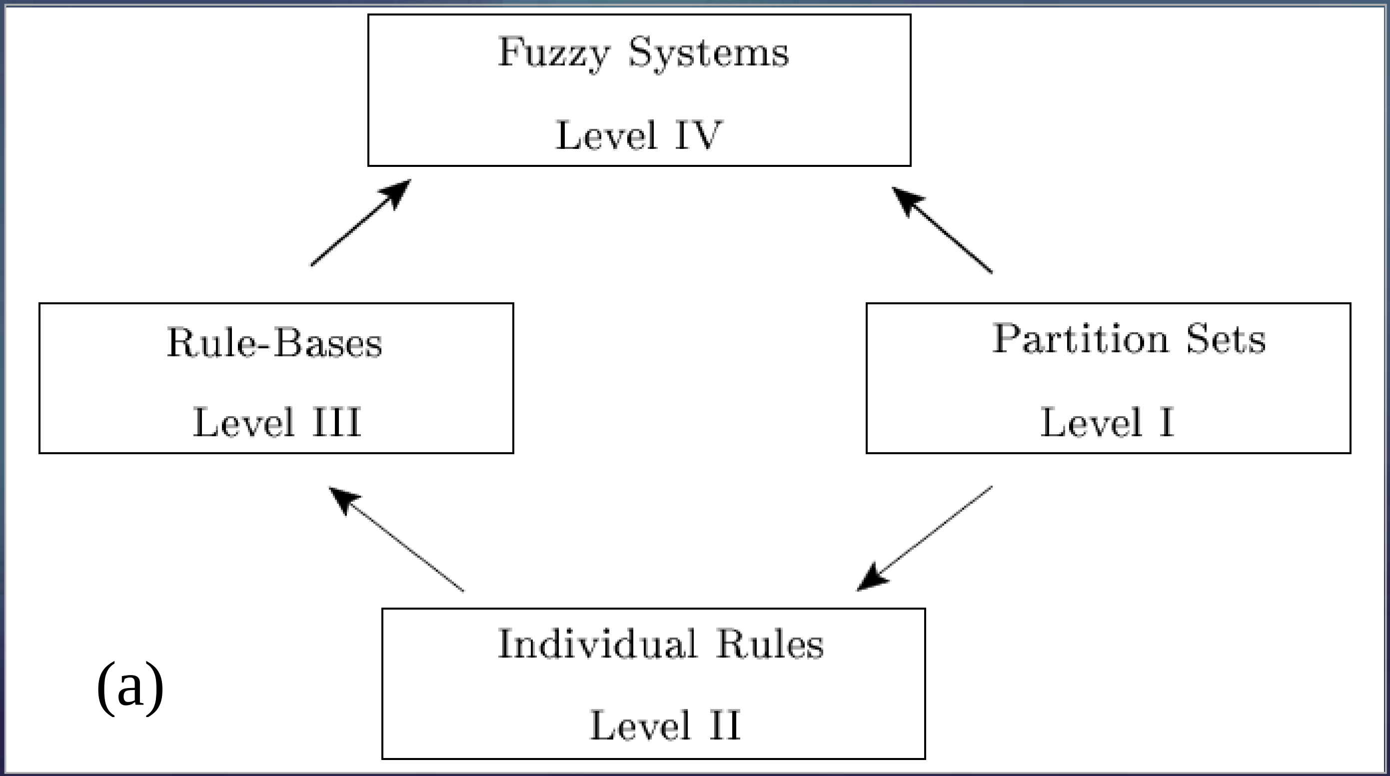
Coevolutionary GFS approach



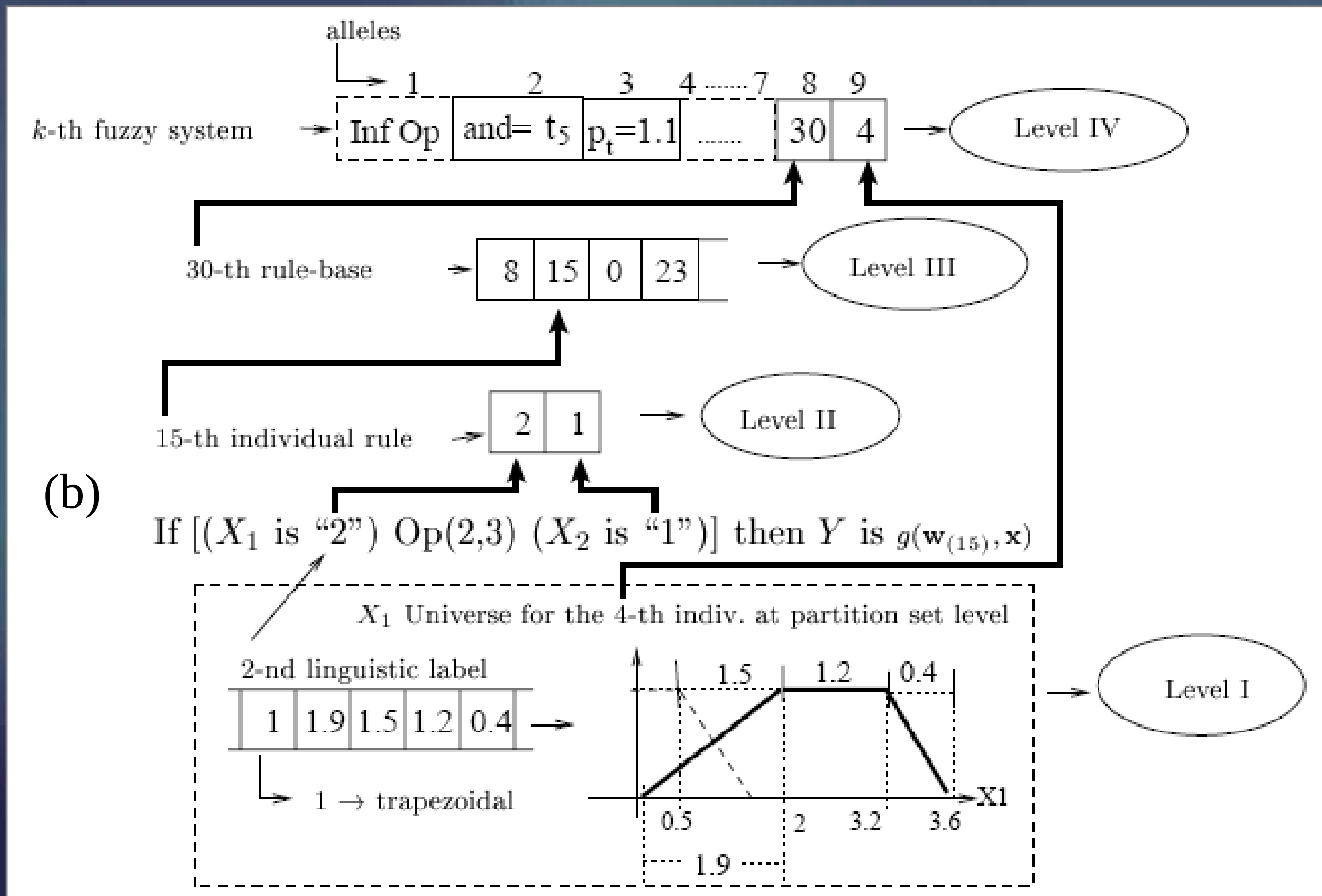
Pedrycz and Gomide, FSE 2007

13.5 Hierarchical collaborative relations

Collaboration between species



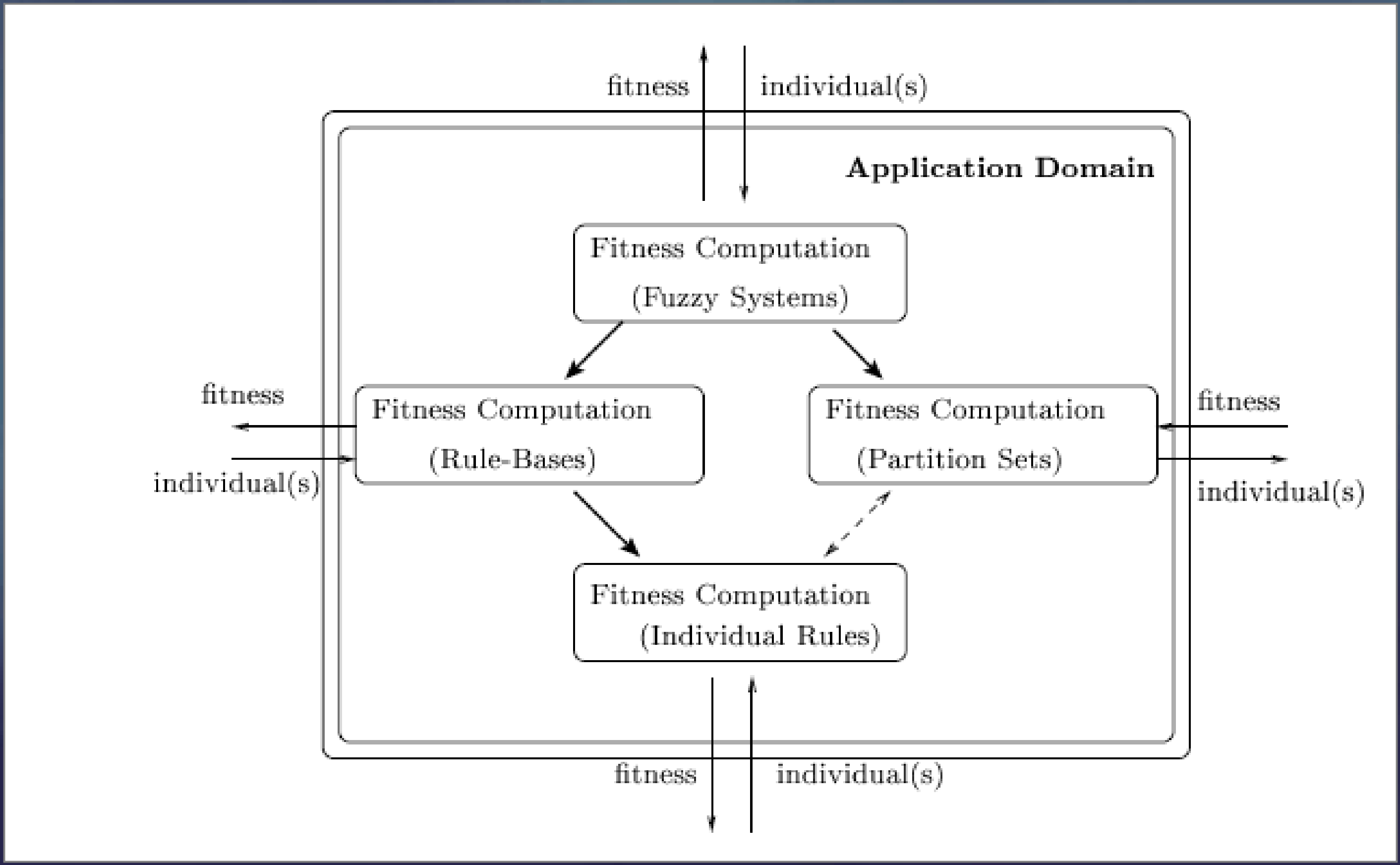
Collaboration between individuals



R_j: **If** x_1 is A_1^j ***and*** ...***and*** x_n is A_n^j **then** y = g(w_j,**x**)

Pedrycz and Gomide, FSE 2007

Fitness evaluation in hierarchical collaborative evolution



Example: function approximation

R_j: **If** x_1 is A_1^j ***and*** ...***and*** x_n is A_n^j **then** y = g(w_j,**x**)

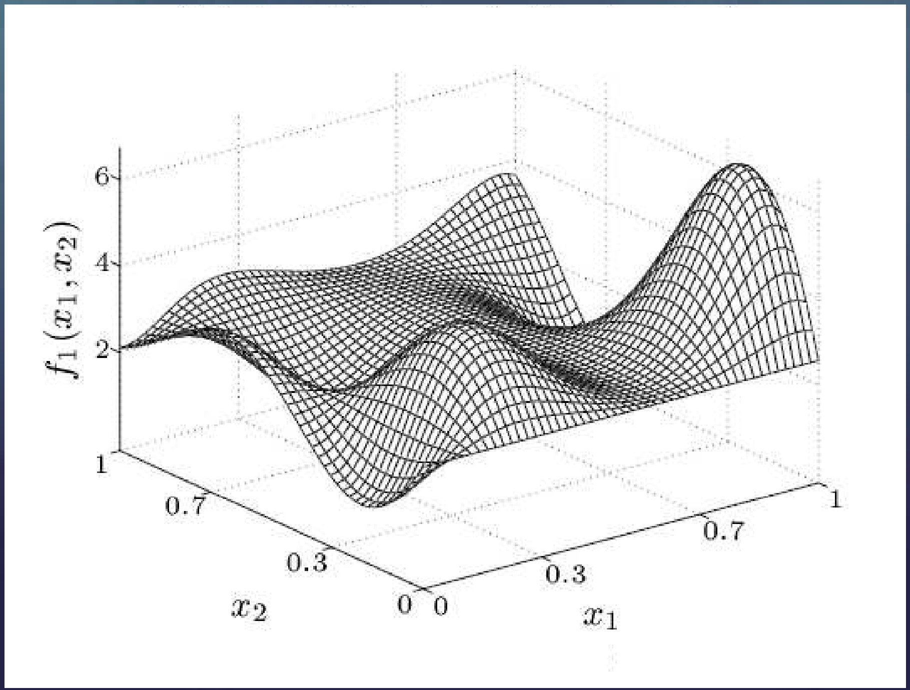
and = t-norm

$$a\,t\,b = \frac{ab}{p_t + (1 - p_t)(a + b - ab)}$$

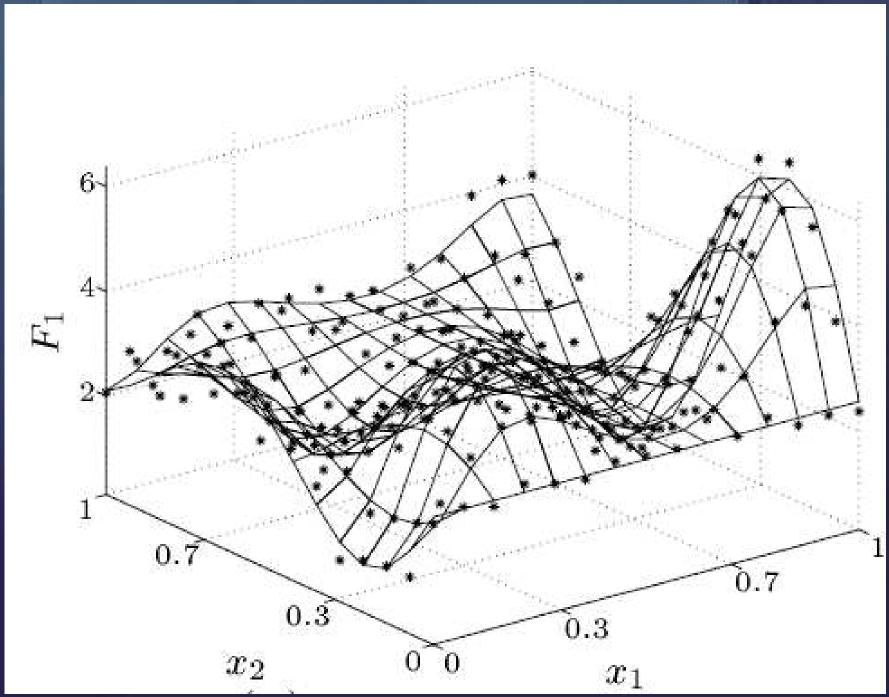
p_t : obtained by coevolution

g(w_j,**x**) : least squares + pruning

$$F_1 : \Omega \rightarrow R$$
$$F_1(x_1, x_2) = f_1(x_1, x_2) + N(m, \sigma)$$
$$f_1(x_1, x_2) = 1.9(1.35 + \exp(x_1) sin[13(x_1 - 0.6)^2 \exp(-x_2) sin(7x_2)])$$
$$\Omega = [0,1],\ m = 0,\ \sigma = 0.3$$



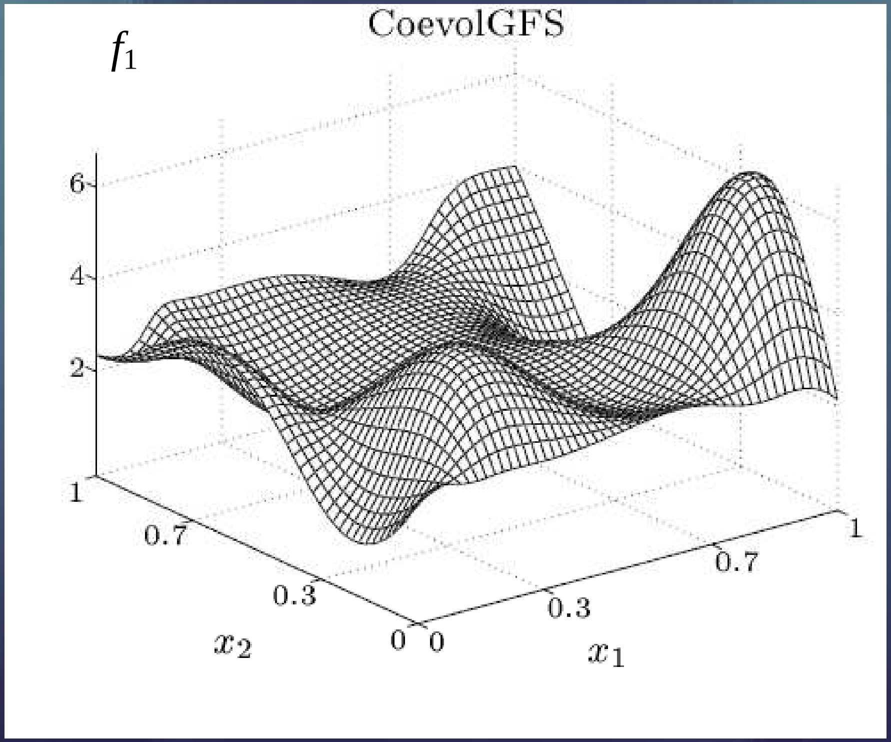
Original function

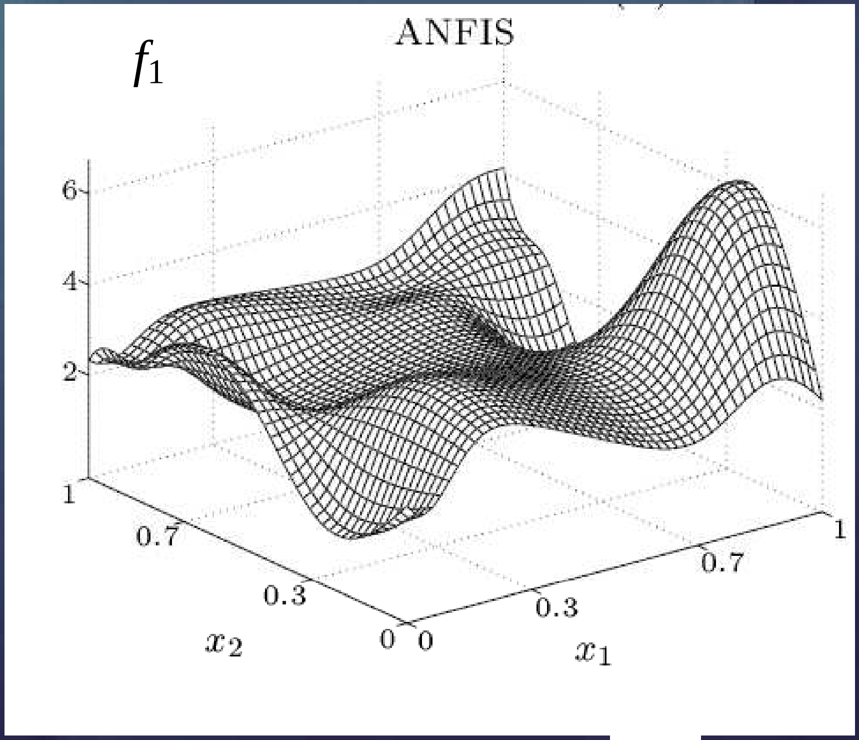


Training data

Pedrycz and Gomide, FSE 2007

Result

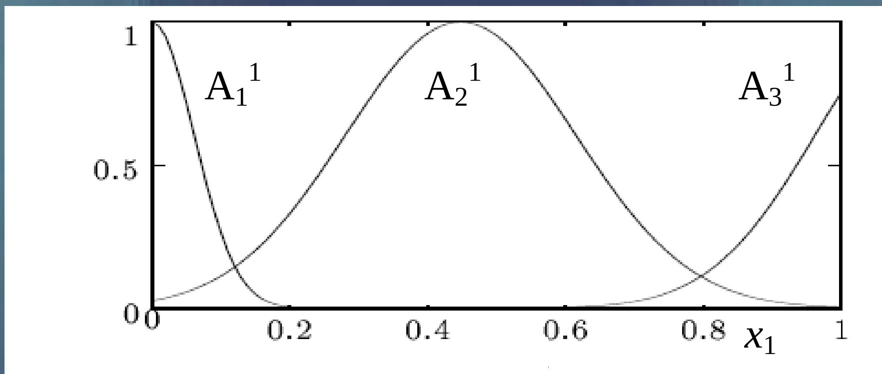


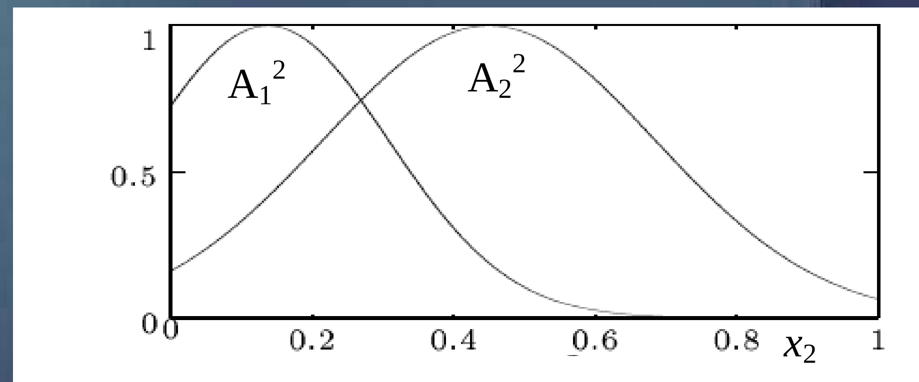


Pedrycz and Gomide, FSE 2007

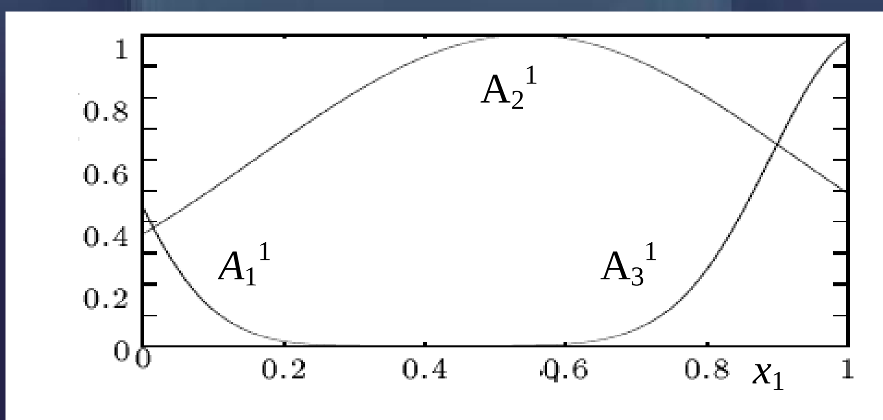
Partitions

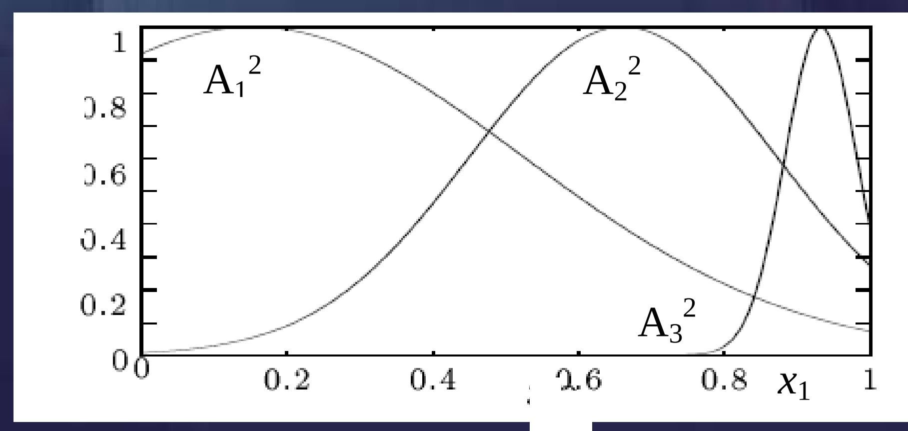
CoevolGFS





ANFIS





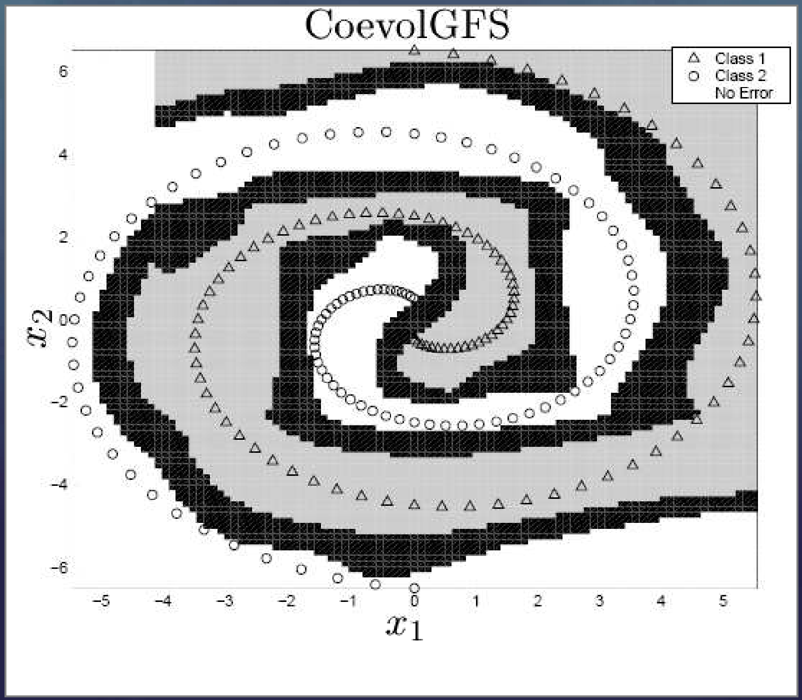
Pedrycz and Gomide, FSE 2007

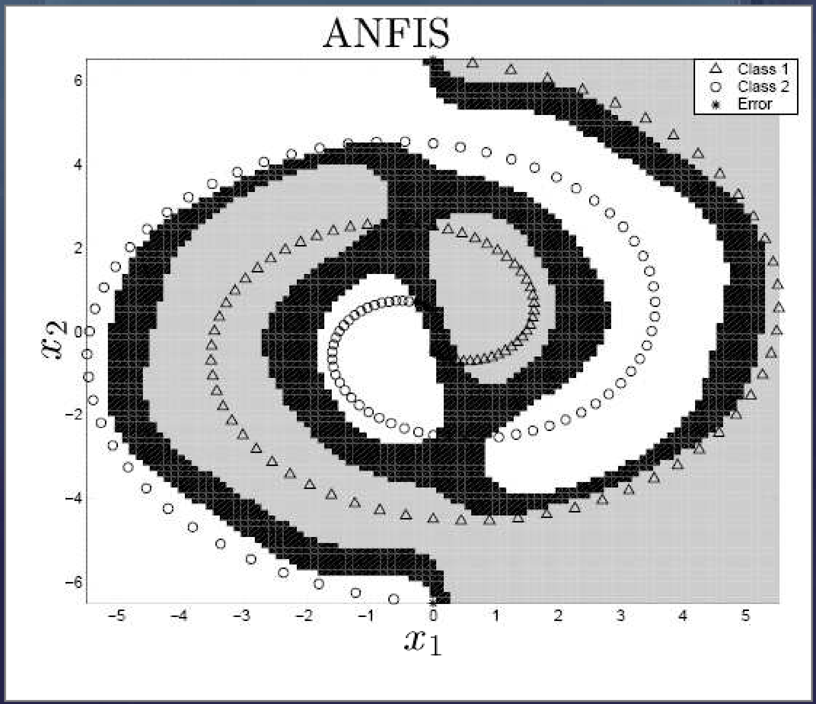
RME for function approximation example

Approach	Training RME	Test RME	Number of Rules
CoevolGFS	0.25	0.13	8
ANFIS	0.32	0.21	9

Example: classification

- Intertwined spirals





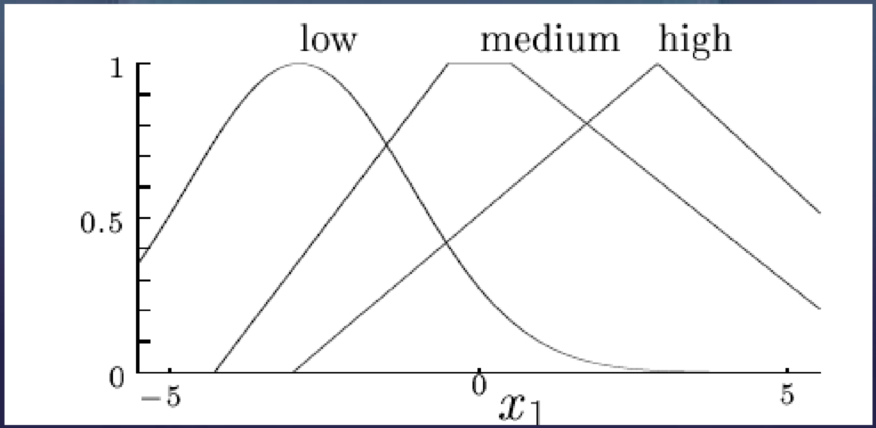
Classification rules

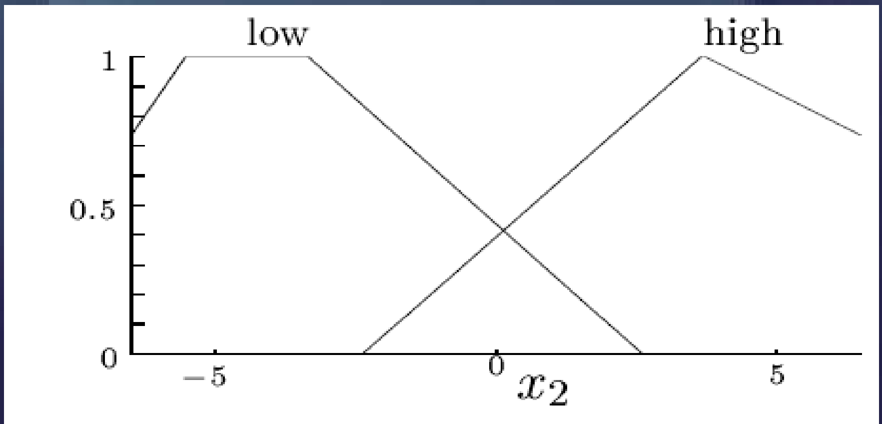
R_1 : If x_1 is low *and* x_2 is low then $y = -0.31 + 1.6x_1 - 0.26x_2 + 0.34x_1^2 + 0.17x_2^2 - 0.1x_1x_2$

R_2 : If x_1 is medium *and* x_2 is low then $y = 15.3 - 1.3x_1 + 7.7x_2 - 0.05x_1^2 + 0.84x_2^2 - 0.46x_1x_2$

R_3 : If x_1 is medium *and* x_2 *is* high then $y = -17.2 - 2.2x_1 + 7.6x_2 - 0.08x_1^2 - 0.78x_2^2 + 0.45x_1x_2$

R_2 : If x_1 is high *and* x_2 is high then $y = 1.14 + 2.0x_1 + 1.24x_2 - 0.25x_1^2 - 0.28x_2^2 - 0.34x_1x_2$





Pedrycz and Gomide, FSE 2007

Classification performance: Intertwined spirals

Approach	Cycles	Misclassification	Number of Rules
CoevolGFS	529	18	9
ANFIS	1000	0.21	9

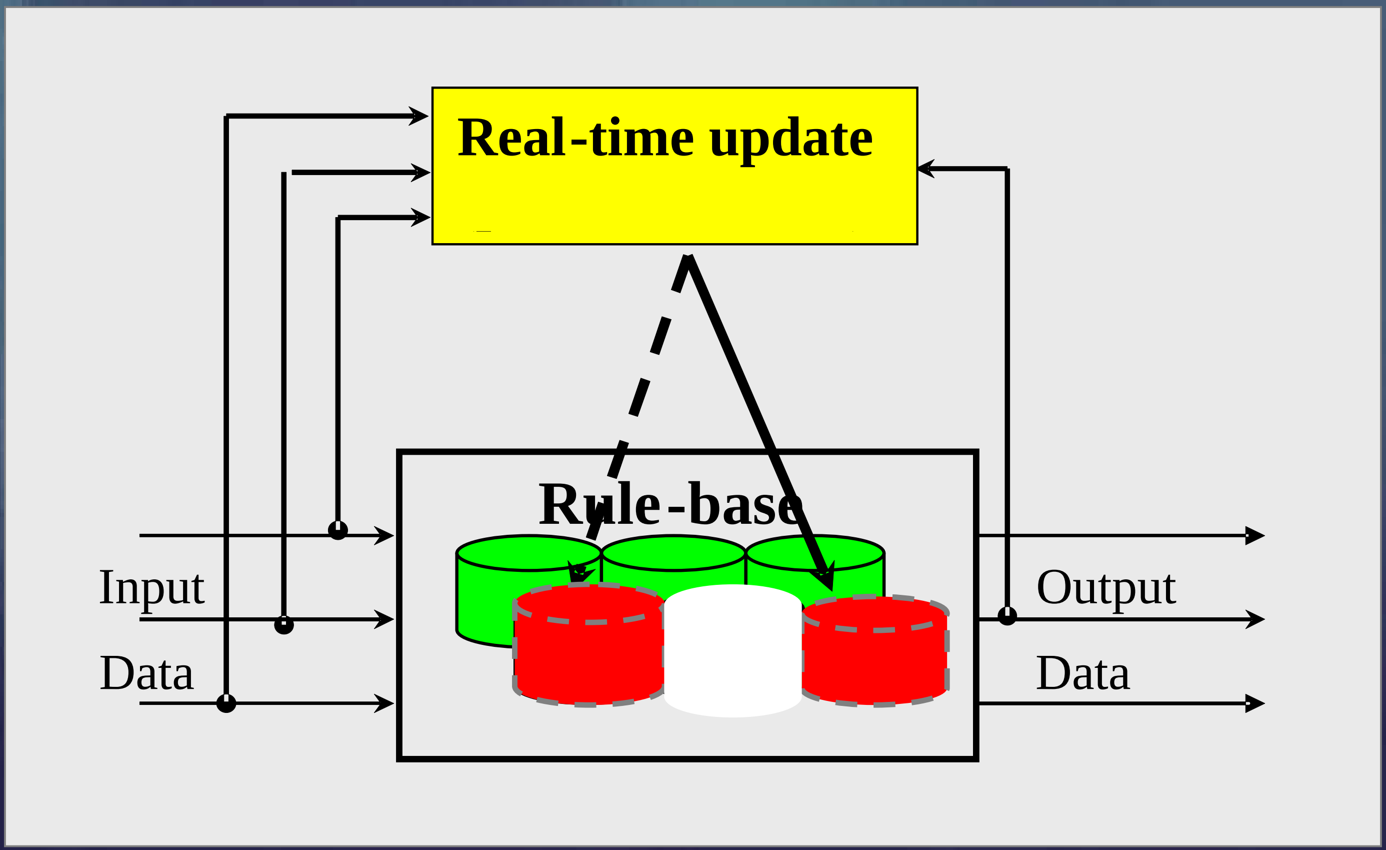
Pedrycz and Gomide, FSE 2007

13.6 Evolving fuzzy systems

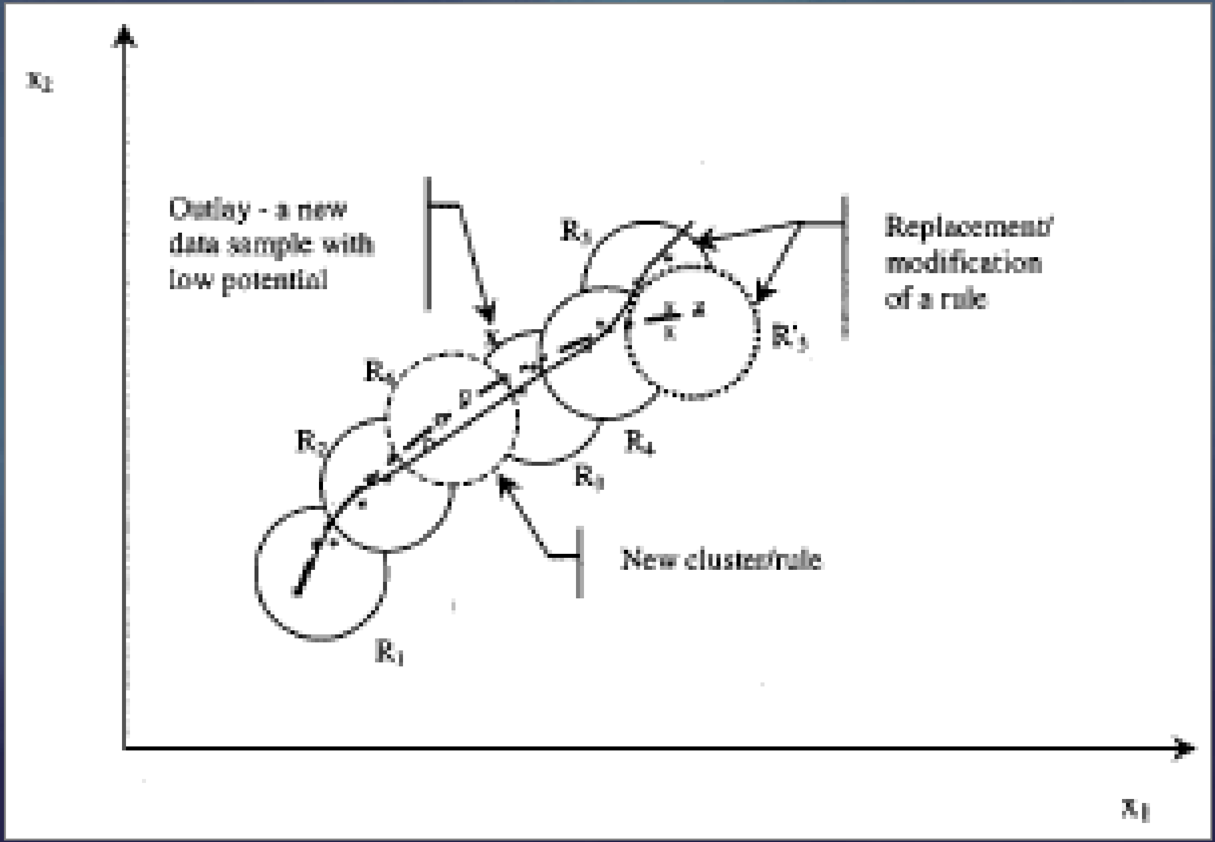
Evolving fuzzy systems

- Evolving systems: an approach to develop adaptive fuzzy models
- Evolving modeling targets nonstationary process and systems
- Main properties
  - inherit new knowledge
  - gradual changes
  - life-long learning
  - self organization of the system structure
  - complements GFS approach
  - may act online

Rule base evolution

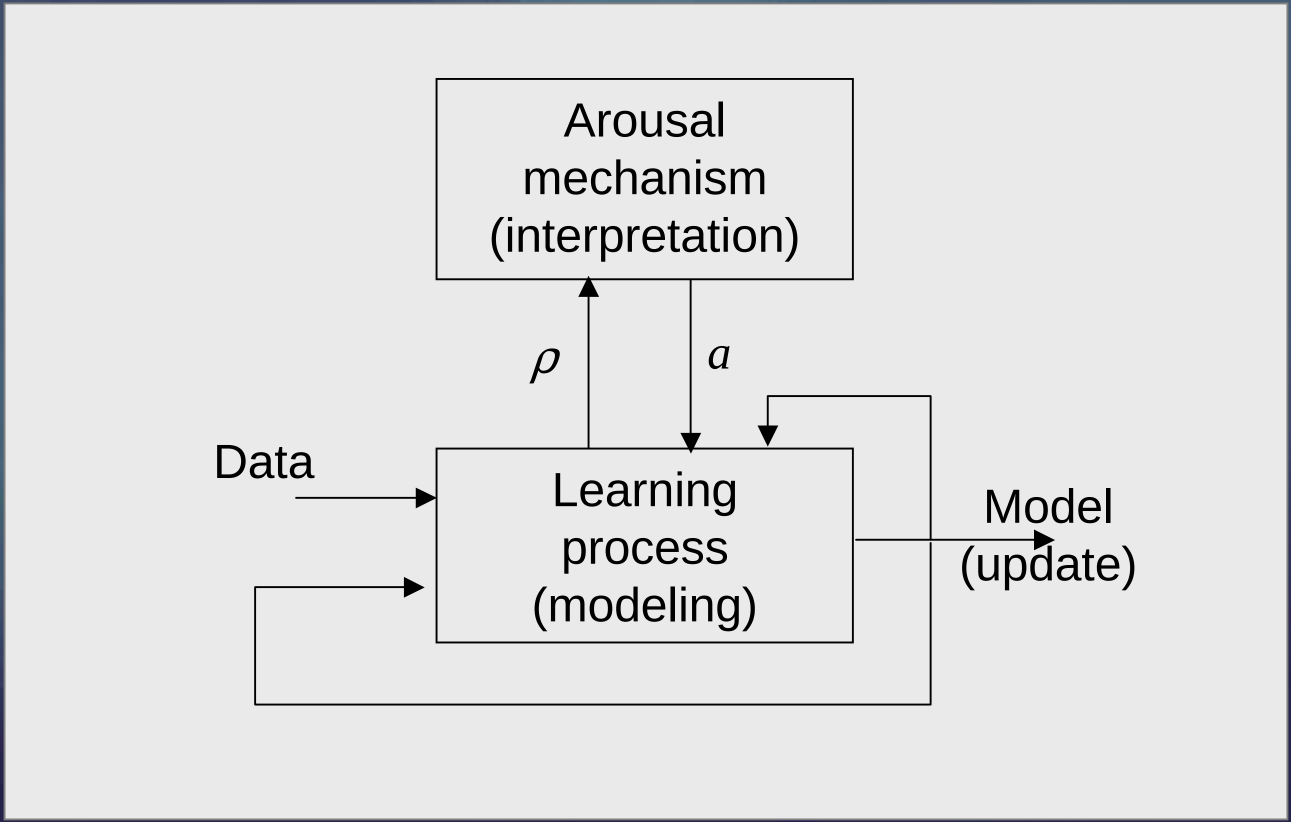


Recursive clustering



Pedrycz and Gomide, FSE 2007

Participatory learning



(Details in Chapter 14)

Functional fuzzy models

$$R_i : if\ \mathbf{x}\ is\ \mathbf{A}_i\ then\ y_i = a_{i0} + \sum_{j=1}^{n} a_{ij} x_j$$

$$A_j^i(x_j) = \exp[-k_{ij}(x_j - v_{ij})^2]$$

$$y = \sum_{i=1}^{c} w_i y_i$$

$$w_i(x) = \frac{\lambda_i(x)}{\sum_{i=1}^{c} \lambda_i(x)}$$

$$\lambda_i = A_1^i(x_1)\ t\ A_2^i(x_2)\ t \cdots t\ A_n^i(x_n)$$

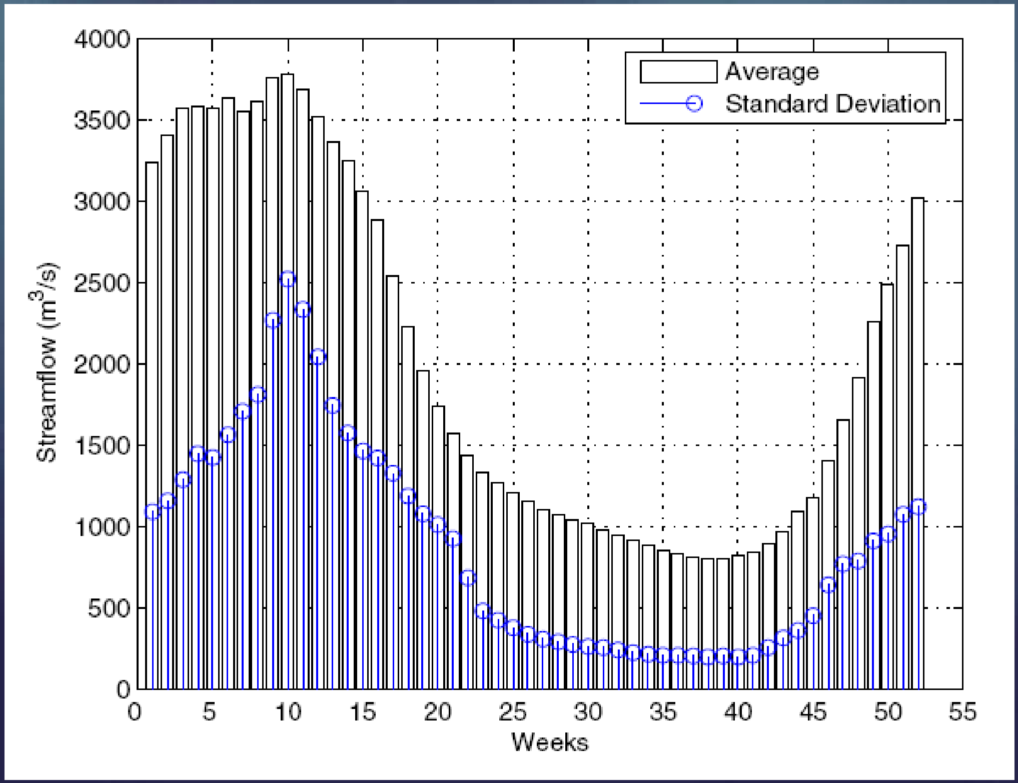
Evolving participatory learning algorithm

```
procedure EVOLVE-PARTICPATORY- LEARNING (x,y) returns an output
input : data x,y
local:    antecedent parameters
          consequent parameters

 INITIALIZE-RULES-PARAMETERS
       do forever
            read x
            PL-CLUSTERING
            UPDATE-RULE-BASE
            RUN-LEAST-SQUARES(x,y)
            COMUTE-RULE-ACTIVATION
            COMPUTE-OUTPUT
 return y
```

Example

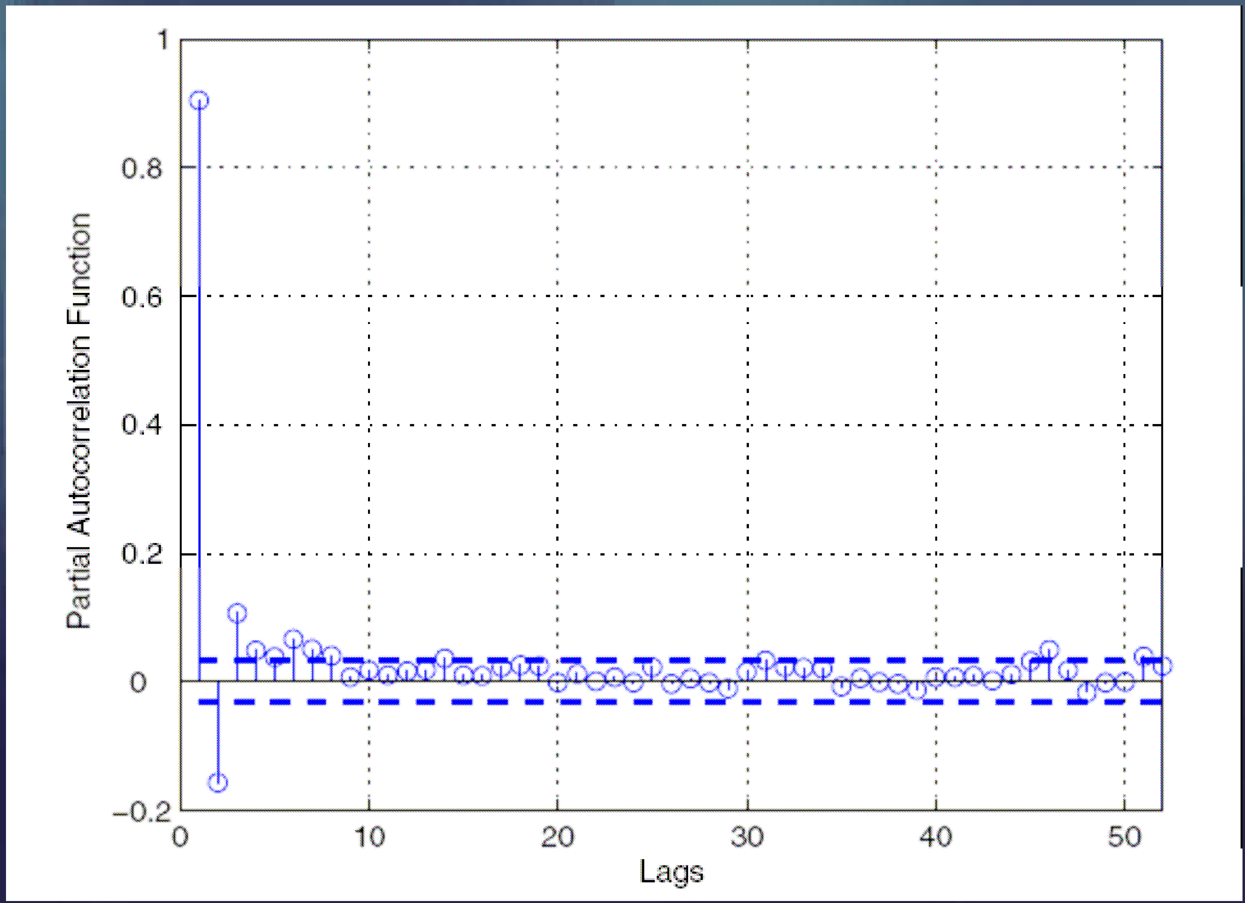
Time series forecasting



Average weekly inflows of a power plant

Pedrycz and Gomide, FSE 2007

Estimated partial correlation



Performance measures

$$RMSE = \sqrt{\frac{1}{P}\sum_{k=1}^{P}(x^k - x_d^k)^2}$$

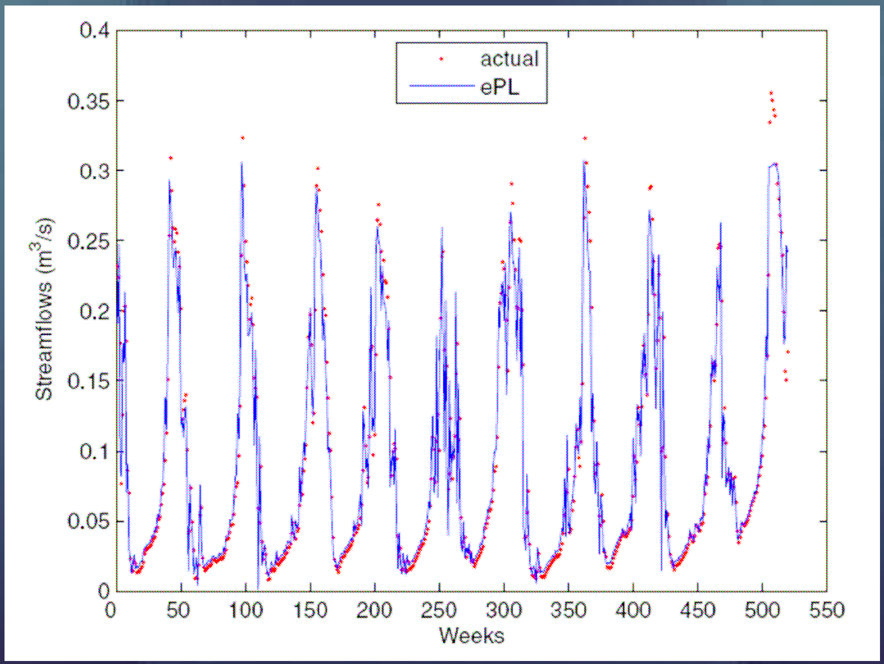
$$MRE = \frac{100}{P}\sum_{k=1}^{P}\frac{|x^k - x_d^k|}{x_d^k}$$

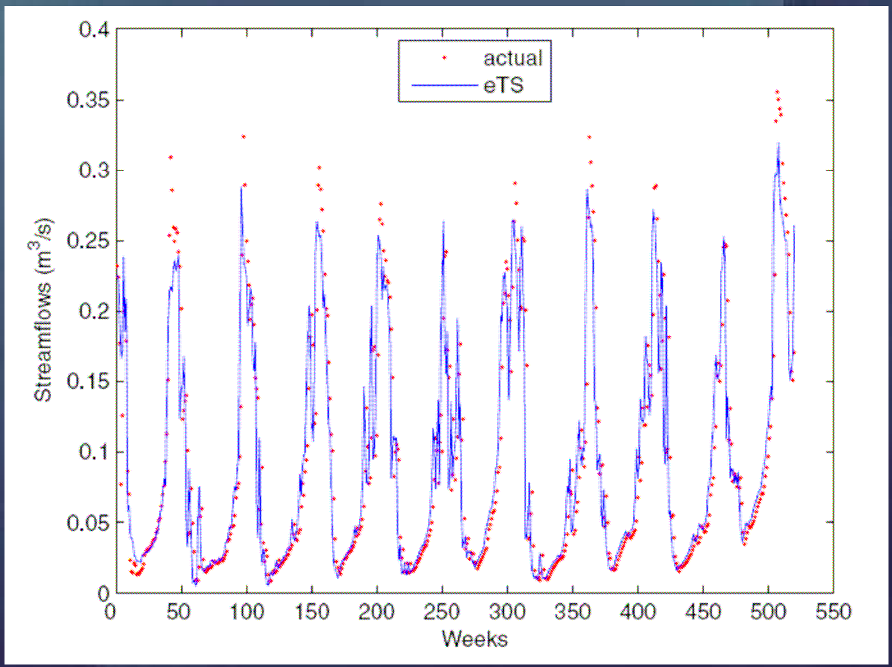
$$MAD = \frac{1}{P}\sum_{k=1}^{P}|x^k - x_d^k|$$

$$RE_{\max} = 100\max\left(\frac{|x^k - x_d^k}{x_d^k}\right)$$

$$\rho = \frac{\sum_{k=1}^{P}(x_d^k - \bar{x}_d)(x^k - \bar{x})}{\sqrt{\sum_{k=1}^{P}(x_d^k - \bar{x}_d)^2 \sum_{k=1}^{P}(x^k - \bar{x})^2}}$$

Result





Pedrycz and Gomide, FSE 2007

Forecasting performance average weekly inflow

Error	Models	
	ePL	eTS
RMSE (m^3/s)	378.71	545.28
MAD (%)	240.55	356.85
MRE (%)	12.54	18.42
RE_{max} (%)	75.51	111.22
ρ	0.95	0.89
Number of rules	2	2

Pedrycz and Gomide, FSE 2007