Contents

13.1 Computational intelligence
13.2 Recurrent neurofuzzy systems
13.3 Genetic fuzzy systems
13.4 Coevolutionary hierarchical genetic fuzzy system
13.5 Hierarchical collaborative relations
13.6 Evolving fuzzy systems
13.1 Computational intelligence
Computational Intelligence

- Fuzzy set theory
- Neural networks
- Evolutionary systems
- Neural fuzzy systems
- Genetic fuzzy systems
- Immune systems
- Neural evolutionary systems
- Evolving systems
- Swarm 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

- Computational intelligence
- Control theory
- Machine learning
- Cognitive sciences
- Systems science
- Operations research
- Artificial intelligence
- Data analysis recognition

Pedrycz and Gomide, FSE 2007
13.2 Recurrent neurofuzzy systems
- Globally recurrent
  - full feedback connections

- Partially recurrent
  - partial feedback connections
Recurrent neural fuzzy network model
Recurrent \textit{and} fuzzy neuron

\[
\begin{align*}
    z_j &= \text{AND}(a_j, w_j) \\
    z_j &= \sum_{i=1}^{n+M} \psi_{\text{and}}(w_{ji} \cdot a_{ji})
\end{align*}
\]
- $N_i$ number of fuzzy sets that granulate the $i$th 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 $i$th input and $j$th neuron
- $z_j$ output of the $j$th neuron
- $v_{kj}$ weight $j$th input of the $k$th output neuron
- $r_{jl}$ feedback connection of the $l$th input of the $j$th neuron
- $y_k = \psi(u_k)$ output $k$th neuron of the output layer
Output layer neuron

\[ y = \psi(u_k) = \psi(\sum_{j=1}^{M} v_{kj} z_j) \]

Pedrycz and Gomide, FSE 2007
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 ≤ ε do
    choose and input-output pair x and y of the data set
    ACTIVE-AND-NEURONS
    ENCODING
    UPDATE-WEIGHTS
return a network

Pedrycz and Gomide, FSE 2007
Example

Chaotic NH3 laser time series data

• first 1000 samples for learning
• predict next 100 steps
100 steps ahead prediction

Pedrycz and Gomide, FSE 2007
Normalized squared forecasting errors (NSE) NH3 laser time series

<table>
<thead>
<tr>
<th>Model</th>
<th>1 step ahead</th>
<th>100 steps ahead</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIR</td>
<td>0.0230</td>
<td>0.0551</td>
</tr>
<tr>
<td>NFN</td>
<td>0.0139</td>
<td>0.0306</td>
</tr>
</tbody>
</table>

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

Pedrycz and Gomide, FSE 2007
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

Pedrycz and Gomide, FSE 2007
Genetic Fuzzy Systems

Genetic algorithms

Genetic design

Learning

Knowledge base

Encoder

Decoder

Processing

Fuzzy processing

Pedrycz and Gomide, FSE 2007
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

Fuzzy Systems
Level IV

Rule-Bases
Level III

Partition Sets
Level I

Individual Rules
Level II

(a)
Collaboration between individuals

\[ R_j: \text{If } x_1 \text{ is } A_1^j \text{ and } \ldots \text{and } x_n \text{ is } A_n^j \text{ then } y = g(w_j, x) \]
Fitness evaluation in hierarchical collaborative evolution
Example: function approximation

\[ R_j: \text{If } x_1 \text{ is } A_1^j \text{ and } \ldots \text{and } x_n \text{ is } A_n^j \text{ then } y = g(w_j,x) \]

*and* = t-norm

\[
at = \frac{ab}{p_t + (1 - p_t)(a + b - ab)}
\]

\(p_t\) : obtained by coevolution

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

Pedrycz and Gomide, FSE 2007
$F_1 : \Omega \rightarrow R$

$F_1(x_1, x_2) = f_1(x_1, x_2) + N(m, \sigma)$

$f_1(x_1, x_2) = 1.9(1.35 + \exp(x_1) \sin[13(x_1 - 0.6)^2 \exp(-x_2) \sin(7x_2)])$

$\Omega = [0,1], m = 0, \sigma = 0.3$
Result

\begin{figure}[h]
\centering
\subfloat[\textit{CoevolGFS}]{
  \includegraphics[width=0.45\textwidth]{coevalgfs.png}
}
\hfill
\subfloat[\textit{ANFIS}]{
  \includegraphics[width=0.45\textwidth]{anfis.png}
}
\caption{Comparison of \textit{CoevolGFS} and \textit{ANFIS} for function \( f_1 \) with \( x_1 \) and \( x_2 \) as input variables.}
\end{figure}

Pedrycz and Gomide, FSE 2007
Partitions

CoevolGFS

ANFIS

Pedrycz and Gomide, FSE 2007
RME for function approximation example

<table>
<thead>
<tr>
<th>Approach</th>
<th>Training RME</th>
<th>Test RME</th>
<th>Number of Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoevolGFS</td>
<td>0.25</td>
<td>0.13</td>
<td>8</td>
</tr>
<tr>
<td>ANFIS</td>
<td>0.32</td>
<td>0.21</td>
<td>9</td>
</tr>
</tbody>
</table>

Pedrycz and Gomide, FSE 2007
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 \( y = -17.2 - 2.2x_1 + 7.6x_2 - 0.08x_1^2 - 0.78x_2^2 + 0.45x_1x_2 \)

\( R_4 \): If \( x_1 \) is high and \( x_2 \) is high then \( y = 1.14 + 2.0x_1 + 1.24x_2 - 0.25x_1^2 - 0.28x_2^2 - 0.34x_1x_2 \)
## Classification performance: Intertwined spirals

<table>
<thead>
<tr>
<th>Approach</th>
<th>Cycles</th>
<th>Misclassification</th>
<th>Number of Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoevolGFS</td>
<td>529</td>
<td>18</td>
<td>9</td>
</tr>
<tr>
<td>ANFIS</td>
<td>1000</td>
<td>0.21</td>
<td>9</td>
</tr>
</tbody>
</table>

---

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

Pedrycz and Gomide, FSE 2007
Rule base evolution
Recursive clustering

Pedrycz and Gomide, FSE 2007
Participatory learning

(Arousal mechanism (interpretation))

(Data) → Learning process (modeling) → Model (update)

(ρ) → (a)

(Details in Chapter 14)
Functional fuzzy models

\[ R_i: \text{if } x \text{ is } A_i \text{ then } y_i = a_{i0} + \sum_{j=1}^{n} a_{ij} x_j \]

\[ A^i_j(x_j) = \exp[-k_{ij}(x_j - v_{ij})^2] \]

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

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

\[ \lambda_i = A^i_1(x_1) \times A^i_2(x_2) \times \cdots \times A^i_n(x_n) \]
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)
    COMPUTE-RULE-ACTIVATION
    COMPUTE-OUTPUT
  return y
Example

Time series forecasting

Average weekly inflows of a power plant
Estimated partial correlation
Performance measures

\[ RMSE = \sqrt{\frac{1}{P} \sum_{k=1}^{P} (x^k - x^k_d)^2} \]

\[ MRE = \frac{100}{P} \sum_{k=1}^{P} \frac{|x^k - x^k_d|}{x^k_d} \]

\[ MAD = \frac{1}{P} \sum_{k=1}^{P} |x^k - x^k_d| \]

\[ RE_{\text{max}} = 100 \max \left( \frac{|x^k - x^k_d|}{x^k_d} \right) \]

\[ \rho = \frac{\sum_{k=1}^{P} (x^k_d - \bar{x}_d)(x^k - \bar{x})}{\sqrt{\sum_{k=1}^{P} (x^k_d - \bar{x}_d)^2 \sum_{k=1}^{P} (x^k - \bar{x})^2}} \]
Result
### Forecasting performance average weekly inflow

<table>
<thead>
<tr>
<th>Error</th>
<th>Models</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ePL</td>
</tr>
<tr>
<td>RMSE (m³/s)</td>
<td>378.71</td>
</tr>
<tr>
<td>MAD (%)</td>
<td>240.55</td>
</tr>
<tr>
<td>MRE (%)</td>
<td>12.54</td>
</tr>
<tr>
<td>$\text{RE}_{\text{max}}$ (%)</td>
<td>75.51</td>
</tr>
<tr>
<td>$\rho$</td>
<td>0.95</td>
</tr>
<tr>
<td>Number of rules</td>
<td>2</td>
</tr>
</tbody>
</table>