Computational Intelligence in Transportation Applications

Měření a zpracování dat (MDS)

Ondřej Přibyl

Ústav aplikované matematiky
ČVUT v Praze, Fakulta dopravní

Organization

- Introduction, definition, history
- Theoretical Basics
  - Artificial Neural Networks
  - Fuzzy System
  - ANFIS - Adaptive Neuro-Fuzzy Inference System
  - Genetic algorithms
- Some Real World Applications – overview
- Discussion and Conclusions

What is intelligence?

- The ability to *learn* or *understand* from experience
- The ability to *acquire* and *retain* knowledge
- The ability to *respond* quickly and successfully to a new situation
- The ability to *use reason* to solve problems

Is a system really intelligent?

- **Turing test**

  a proposal for a test of a machine's capability to perform human-like conversation (Turing 1950)
- **Principle:**

  Place both a human and a machine mimicking human responses outside the field of direct observation and use an unbiased interface to interrogate them.

  If the responses are distinguishable, the machine is not displaying intelligence.

Terminology

- Artificial Intelligence – "Subject dealing with computational models that can think and act rationally"
  - Strong symbolic manipulation (Expert systems)
- Soft Computing – "Emerging approach to computing which parallels the ability of human mind to reason and learn in an environment of uncertainty and imprecision" (L. Zadeh)

Brief History (Jang et al. 1997)

1943 Invention of a computer

	Conventional AI	Neural Networks	Fuzzy Systems	Other
1940s	Cybernetics (Norbert Wiener)	McCulloch - Pitts neuron model		
1950s	Artificial Intelligence	Perceptron		
1960s	Lisp Language	Adaline, Madaline	Fuzzy sets (Zadeh)	
1970s	Knowledge engineering (expert systems)	Back-propagation alg., Cognitron	Fuzzy controller	Genetic algorithm
1980s		Self-organizing map Hopfield Net. Boltzman machine	Fuzzy modeling (TSK model)	Artificial life Immune modelling
1990s			Neuro-fuzzy modeling ANFIS	Genetic programming

Characteristics of soft computing

- Biologically inspired computing models
- Uses human expertise: IF-THEN rules or conventional knowledge representation
- New optimization techniques
- Numerical computation
- New application domains
  - adaptive control, non-linear system identification, pattern recognition, …

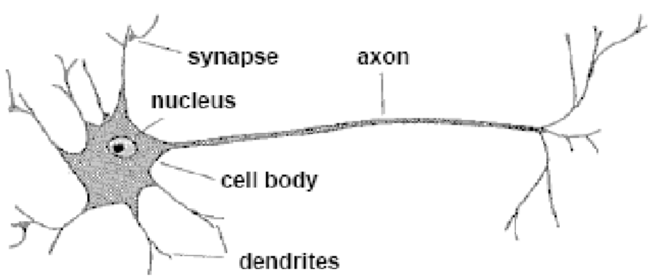
Characteristics of soft computing (cont.)

- Model-free learning
- Intensive computation
- Fault tolerance
- Goal driven characteristics
- Real world applications

Organization

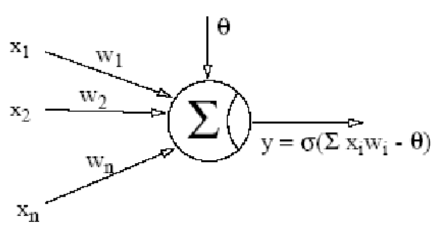
- Introduction, definition, history
- Theoretical Basics
  - **Artificial Neural Networks**
  - Fuzzy System
  - ANFIS - Adaptive Neuro-Fuzzy Inference System
  - Genetic algorithms
- Some Real World Applications – overview
- Discussion and Conclusions

Neuron



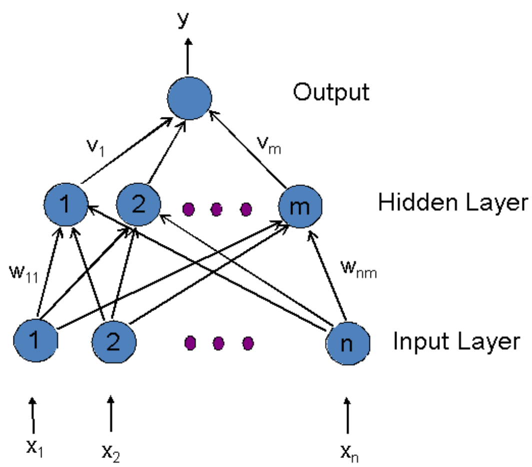
Biological neuron

Model of a neuron



Multilayer feedforward network

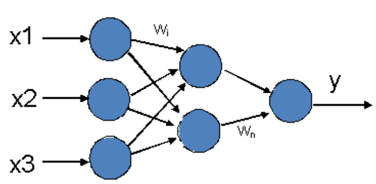
- A two-layered feedforward NN



- The strength is in interconnectivity !

$$y = \frac{1}{1 + \exp\left(-v' f(w, x)\right)}$$

Learning



Training data set

Input			desired output	network output
x1	x2	x3	d	y
12	2	6	3	3
15	4	4	1	2
4	3	4	5	.
13	5	8	3	.

- Mathematically:
  - Initialisation: W_i = rand
  - Updating: $W_{new} = W_{old} + \Delta W$
  - ΔW =? learning rule
  - Performance measure

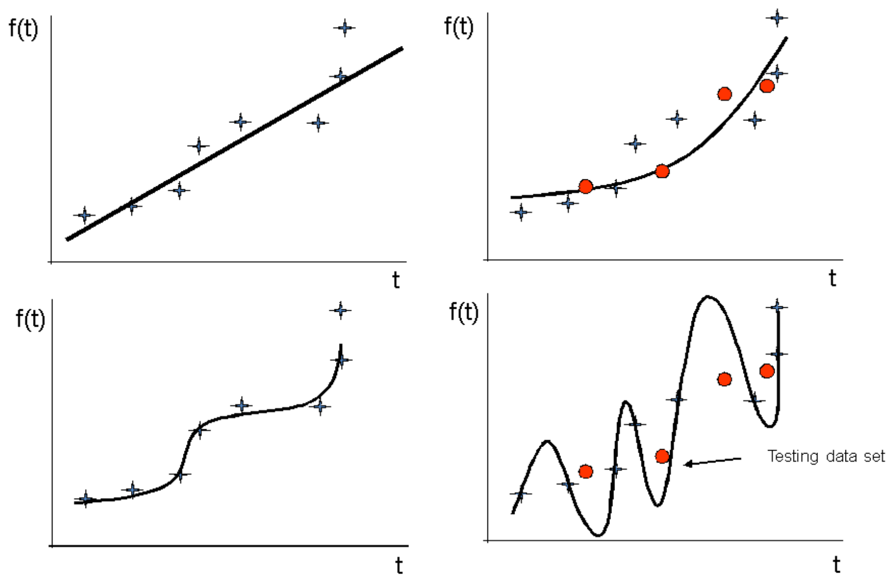
$$\varepsilon(t) = \frac{1}{2}\sum_{k=1}^{l} e_k^2(t), \qquad e_k(t) = d_k(t) - y_k(t)$$

  - Learning rule – gradient descent

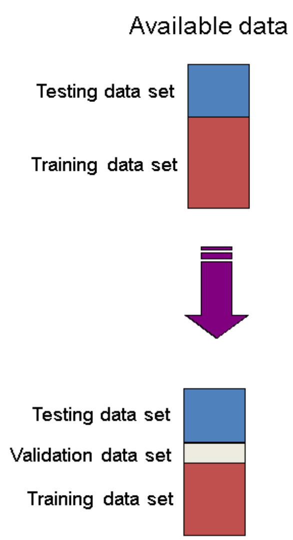
$$\Delta w_{ki}(t) = -\eta \frac{\partial \varepsilon(t)}{\partial w_{ki}(t)}, \qquad \Delta w_{ij}^{h}(t) = -\eta \frac{\partial \varepsilon(t)}{\partial w_{ij}^{h}(t)}$$

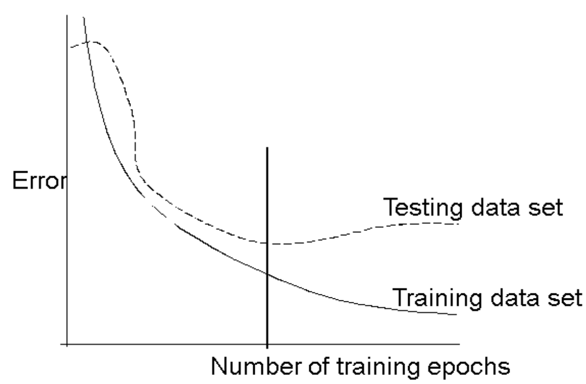
Learning rate

Overfitting - problem of ANN



How to solve problem of overfitting?





Features of ANN

- Fault Tolerance: Neural networks are robust
  - Information in the form of weights is distributed all over the network
  - They can survive the failure of some nodes and their performance degrades gracefully under faults
- Flexibility and Adaptability
  - They can deal with information that is fuzzy, probabilistic, inconsistent and noisy
  - They can adapt intelligently to previously unseen situations
  - They can learn from examples presented to them and do not need to be programmed
- Parallelism: They embody parallel computing
  - Make it possible to build parallel processing hardware for implementing them
  - No logical operations are used after training
  - Extremely fast computation can be achieved
- Learning delay
  - Training neural networks is often time consuming but after training they can operate in real time

Features of ANN (cont.)

- Model free
  - No rules are required to be given in advance
  - There is no need to assume an underlying data distribution such as usually is done in statistical modeling
- Size and complexity
  - For large scale implementations of ANN we need massive arrays of neurons
- "Black box"
  - Individual relations between the input variables and the output variables are not developed by engineering judgment
  - The knowledge extraction is difficult

When to consider using ANN

- Input is high-dimensional (raw sensor data)
- Possibly noisy data
- Training time is unimportant
- Form of target function is unknown
- Human readability of result is unimportant
- When facing multivariate non-linear problems
- Common application fields
  - Pattern recognition
  - Function approximation
  - Prediction, forecasting

Application of ANN to transportation

- Travel Behavior
  - Modelling drivers behavior in signalised urban Intersection
  - Driver decision making model
- Traffic Flow
  - Intersection control
  - Estimation of Speed-Flow relationship
- Traffic management
  - Trip generation model
  - Urban public transport equilibrium
  - Incident detection
  - Prediction parking characteristics
  - Travel time prediction

Organization

- Introduction, definition, history
- Theoretical Basics
  - Artificial Neural Networks
  - **Fuzzy System**
  - ANFIS - Adaptive Neuro-Fuzzy Inference System
  - Genetic algorithms
- Some Real World Applications – overview
- Discussion and Conclusions

"It's fuzzy math."

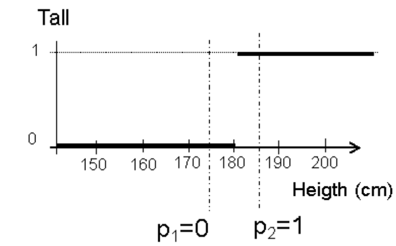
Why Fuzzy Systems?

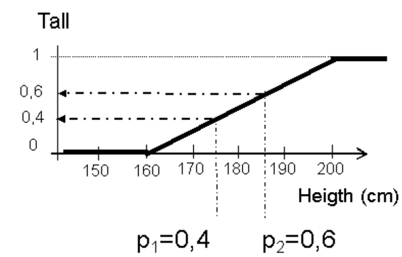
- The knowledge in computers is usually binary coded – using Aristotelian's logic
- It is difficult to understand the representation of data inside computers
- How to teach computers to understand human expressions?
- Is it possible to design a model using direct expert rules

Fuzzy Set Theory

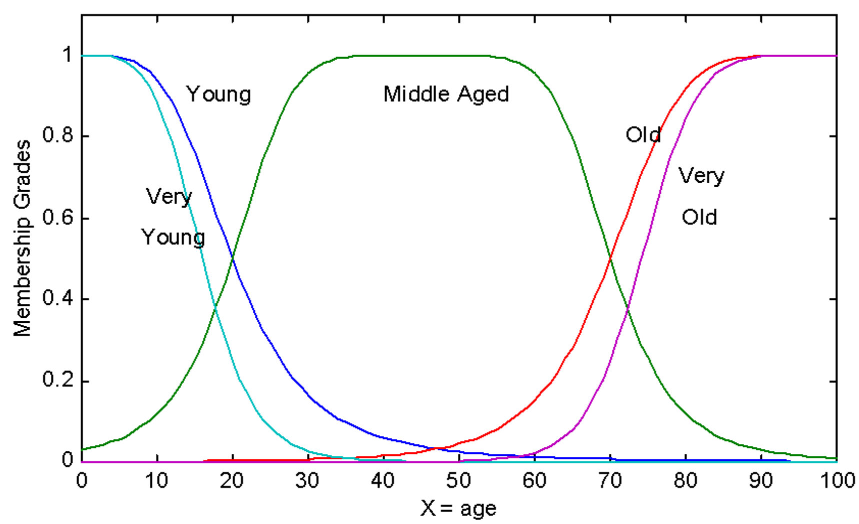
- Aristotelian's logic
  - Tall person = 1 IF height >180cm
- Fuzzy logic
  - Tall person x
- Membership functions
  - Assignes each object to a grade of membership

$$\mu \in \{0,1\}$$



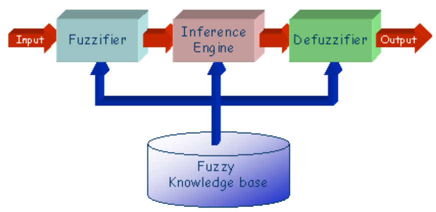


Linguistic variables



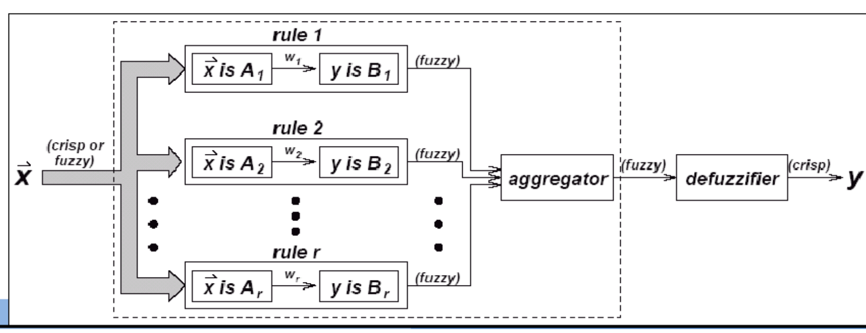
Fuzzy Inference System (FIS)

- Also known as Fuzzy Model
- Three components
  - Knowledge base – IF THEN rules
  - Database (dictionary) – defines membership functions, ...
  - Inference engine/ Reasoning mechanism – defines the defuzzification, ...
- Different types
  - Mamdani FIS
  - Takagi-Sugeno FIS
  - ...

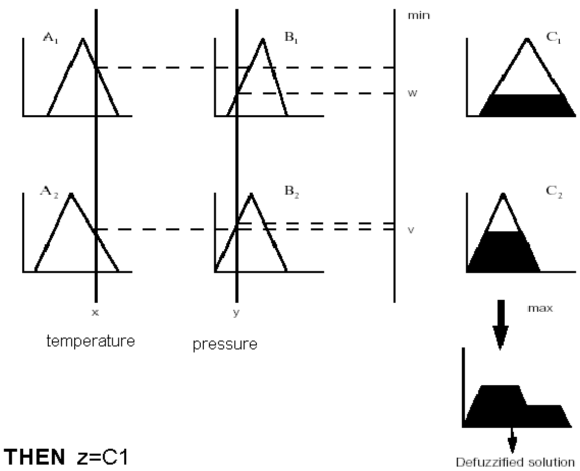


Fuzzy reasoning

1. *Fuzzification*
   Compare the input variables with the membership functions on the antecedent part to obtain the membership values of each linguistic label. (this step is often called
2. Combine (*usually multiplication or* min) the membership values on the premise part to get *firing strength* (*deree of fullfillment) of each rule.*
3. Generate the qualified consequents (either fuzzy or crisp) or each rule depending on the firing strength.
4. Defuzzification
   Aggregate the qualified consequents to produce a crisp output.



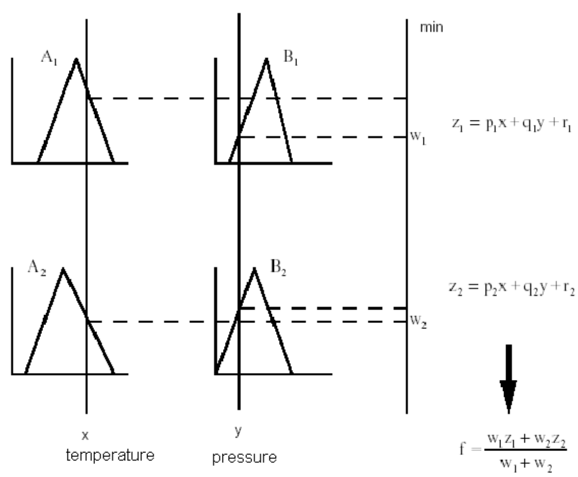
Fuzzy Inference System (Mamdani Model)



RULES:

IF x = A1 **AND** y = B1 **THEN** z=C1
IF x = A2 **AND** y = B2 **THEN** z=C2

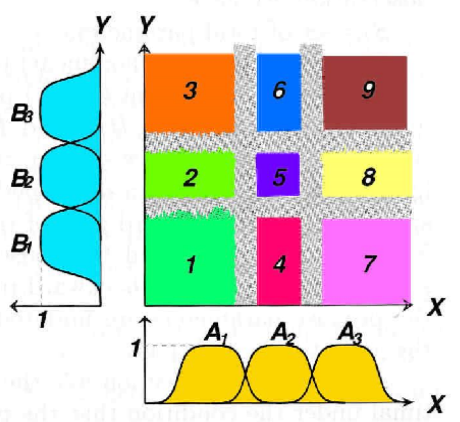
Fuzzy Inference System (Takagi-Sugeno Model)



RULES:

IF x = A1 **AND** y = B1 **THEN** $z1=p_1x+q_1y+r_1$
IF x = A2 **AND** y = B2 **THEN** $z2=p_2x+q_2y+r_2$

Example of fuzzy rules matrix representation



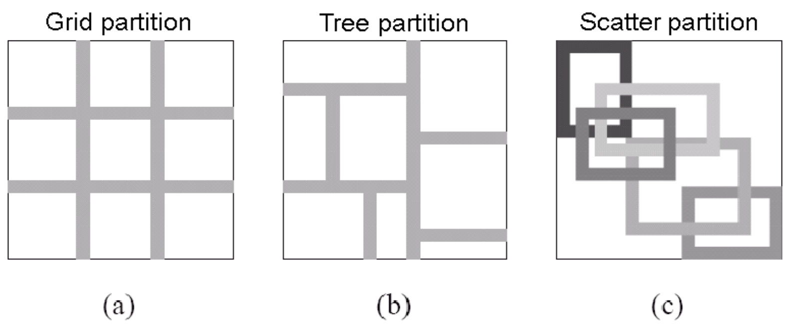
s \ m	VS	S	M
L	M	S	VS
M	RL	RS	S
S	VL	L	M

Features of FS

- Knowledge is represented in the form of comprehensive linguistic rules
  - Transparent control systems
- Able to deal with uncertain and imprecise information
- Suitable for problems involving human behavior
- Suitable for non-linear problems
- Uses expert knowledge
- Problems
  - No standard method for transformation of human knowledge or experience into the FIS
  - Even when human operators exist, their knowledge is often incomplete and episodic, rather then systematic
  - No general procedure for calibrating the system
  - Need for a good method for tuning the membership functions in order to maximize a performance index
  - Curse of dimensionality

Induction (Design) of Fuzzy Systems

- Expert knowledge
- Grid partitioning
- Cluster analysis
- Least square identification
- Decision tree technique



Applications of Fuzzy Systems

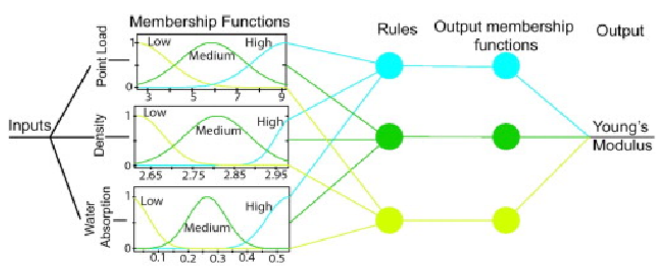
- When human reasoning and decision-making are involved
  - Supervising, planning, scheduling
- Various types of information are involved
  - Measurements and linguistic information
- Problems using natural language
- Very complex systems
- When there is some prior heuristic knowledge

Application of Fuzzy Systems to transportation - examples

- Human choice and decisions
  - Route choice
  - Mode choice
- Driver behavior
  - Car-following behavior
  - Lane-choice
- Control
  - Parking space forecasting
  - Ramp metering
  - Intersection control
  - Incident detection
- Other
  - Vehicle routing problem
  - Vehicle assignment problem
  - Air traffic flow management

Organization

- Introduction, definition, history
- Theoretical Basics
  - Artificial Neural Networks
  - Fuzzy System
  - **ANFIS - Adaptive Neuro-Fuzzy Inference System**
  - Genetic algorithms
- Some Real World Applications – overview
- Discussion and Conclusions



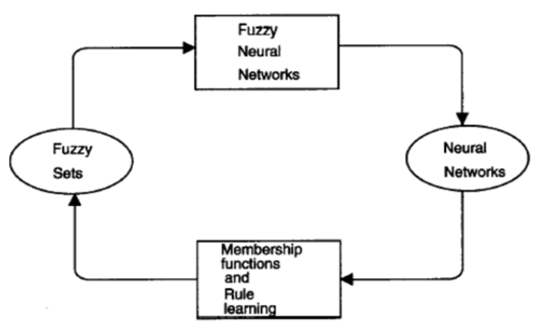
Why ANFIS?

- Drawbacks of artificial neural networks
  - Prior rule-based knowledge cannot be used
  - Learning from scratch
  - "Black box"
  - Difficult to extract knowledge
  - Requires large training data set
- Drawbacks of fuzzy systems
  - Cannot learn
  - There is no standard way how to represent human knowledge in rule base in FIS
  - The human knowledge is often incomplete
  - No known method how to design the membership functions

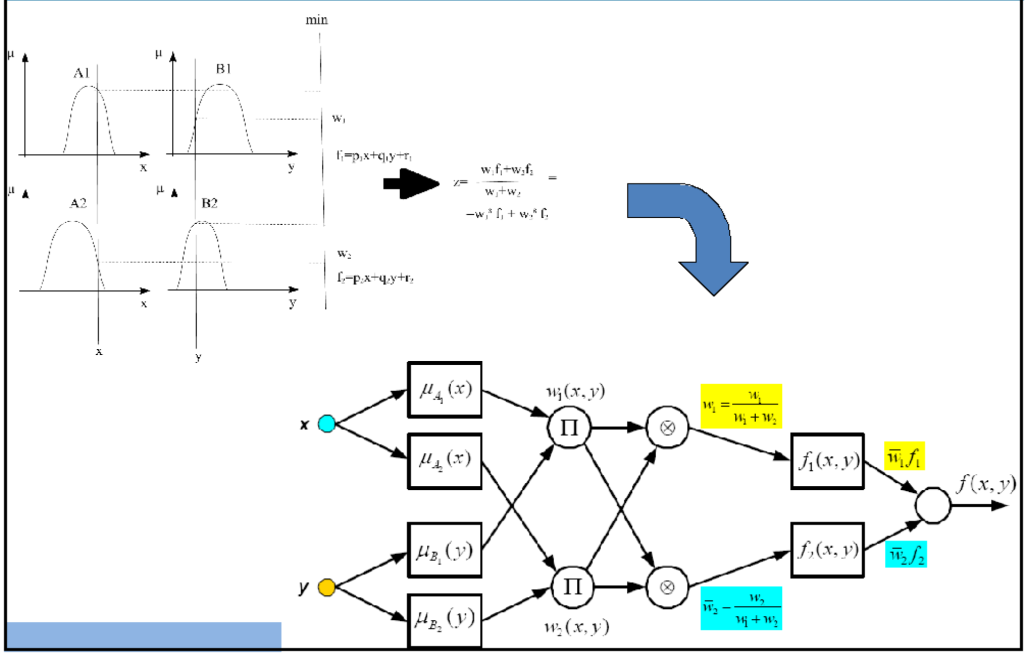
ANFIS

- An adaptive network that is functionally equivalent to FIS
- The parameters (i.e. the membership functions) are modified from examples – learning step

- The process
  - Design a fuzzy system (using prior knowledge)
  - Convert it into an adaptive network
  - Train the network (modify its parameters based on examples)
  - Convert it back into the fuzzy system



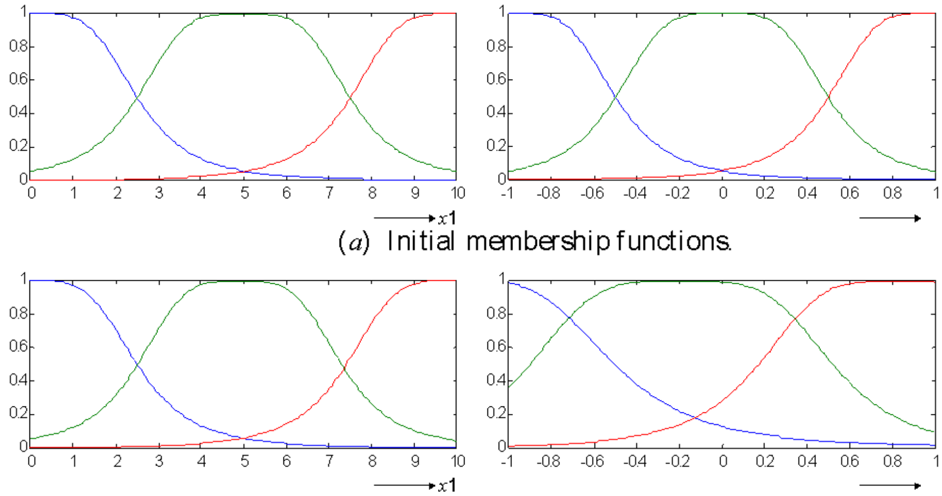
Sugeno FIS and its ANFIS equivalent



Learning in ANFIS

	forward pass	backward pass
MF param. (premise)	fixed	back propagation
Rule param. (consequence)	least-squares	fixed

Membership functions after training (Example)



(*a*) Initial membership functions.

(*b*) Membership functions after 100 epochs of training.

Source: Negnevitski, Pearson education 2005

Advantages of ANFIS

- More robust than ANN
- Rule based representation
- Uses prior knowledge
- Adaptive learning!

Application of ANFIS

- Similar to applications fields of FS and ANN

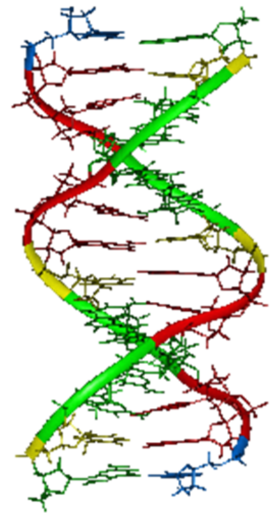
Comparison of different technic

Technique	Model free	Can resist outliers	Explains output	Suits small data sets	Can be adjusted for new data	Reasoning process is visible	Suits complex models	Include known facts
Least squares regression	No	No	Partially	No	No	Yes	No	Partially
Neural networks	Yes	No	No	No	Partially	No	Yes	Partially
Fuzzy Systems	Yes	Partially	Yes	Yes	Partially	Yes	Yes	Yes
ANFIS	Yes	Partially	Yes	Partially	Partially	Partially	Yes	Yes

Yes	(red)
No	(white)
Partially	(yellow)

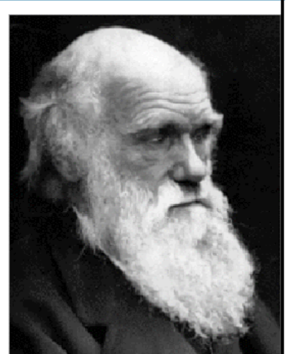
Organization

- Introduction, definition, history
- Theoretical Basics
  - Artificial Neural Networks
  - Fuzzy System
  - ANFIS - Adaptive Neuro-Fuzzy Inference System
  - **Genetic algorithms**
- Some Real World Applications – overview
- Discussion and Conclusions



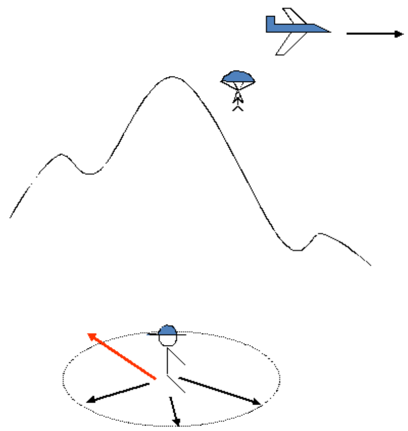
What are genetic algorithms?

- Probabilistic search algorithm
- Based on Darwin's Evolutionary theory
  - Survival of the fittest
  - Natural selection
- Terminology
  - Population
    - Set of solutions
  - Chromosome
    - Defines a solution
    - Usually binary representation
  - Fitness function
    - Expresses the "quality of a solution"



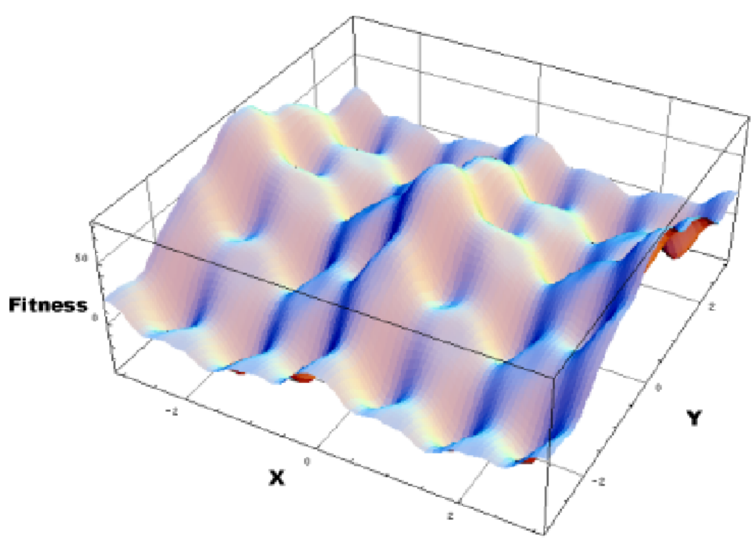
Charles Darwin 1809 - 1882

Hill climbing methods



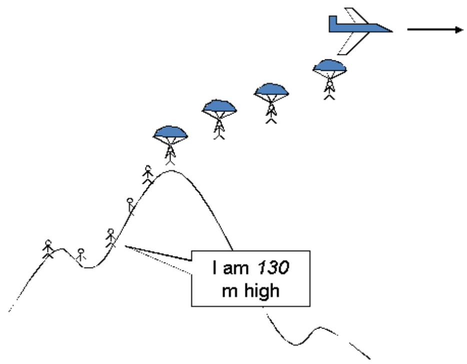
- Problems
  - Function must have "nice" properties (differentiable function)
  - Not always finds the global extreme - the initial conditions influence the result

A two dimensions optimisation problem



Source: Frederic Dreier July 2002

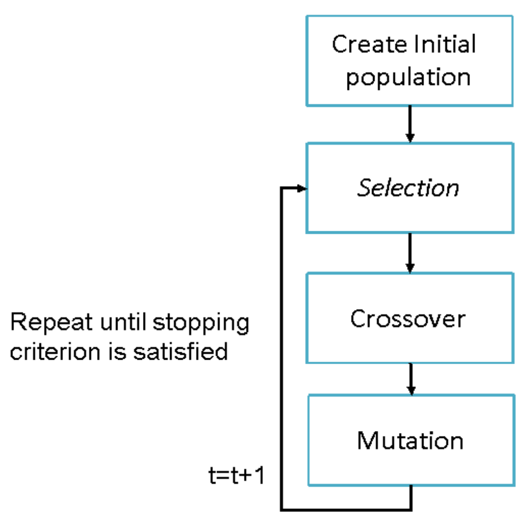
GA-Based methods



- Advantages of GA-based approaches
  - No requirements on the function
- Disadvantages of GA-based approach
  - No guarantee of the result
  - Sensitive to parameter setting
- In general we look for a compromise between:
  - Exploration (crossover and selection)
    - Never converges
  - Local improvements (mutation and selection)
    - Finding local extremes

Ondřej Přibyl | K611MDS | 45

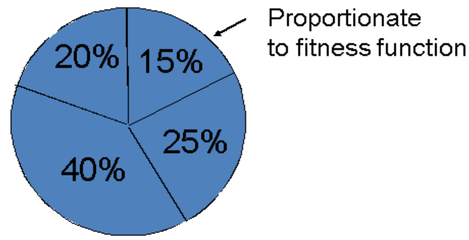
The principle of GA



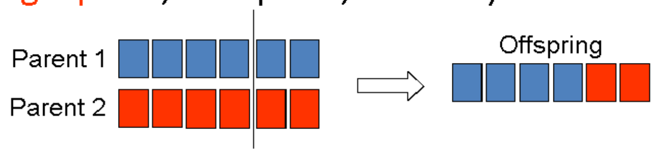
Ondřej Přibyl | K611MDS | 46

GA Operators

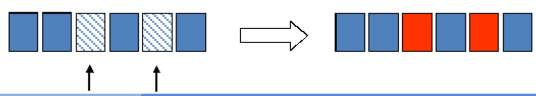
- Selection - Roulette wheel
  (Tournament sel., rank sel., ...)



- Crossover (single point, two-point, uniform)

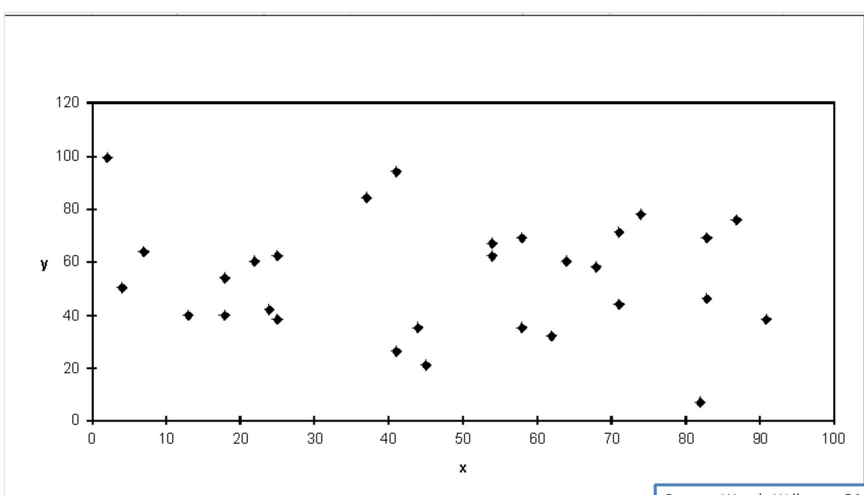


- Mutation



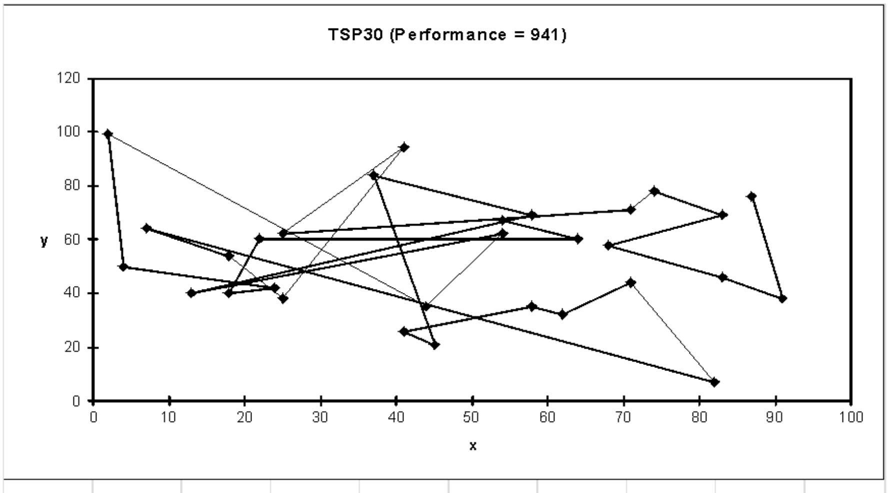
Example - Traveling salesmen problem

Find a tour of a given set of cities so that each city is visited only once the total distance traveled is minimized



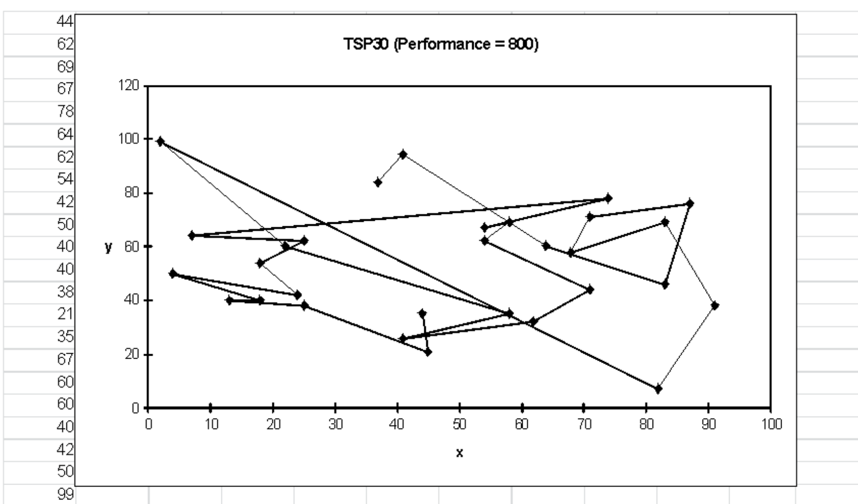
Source: Wendy Williams, GA: A Tutorial

Example TSP - progress



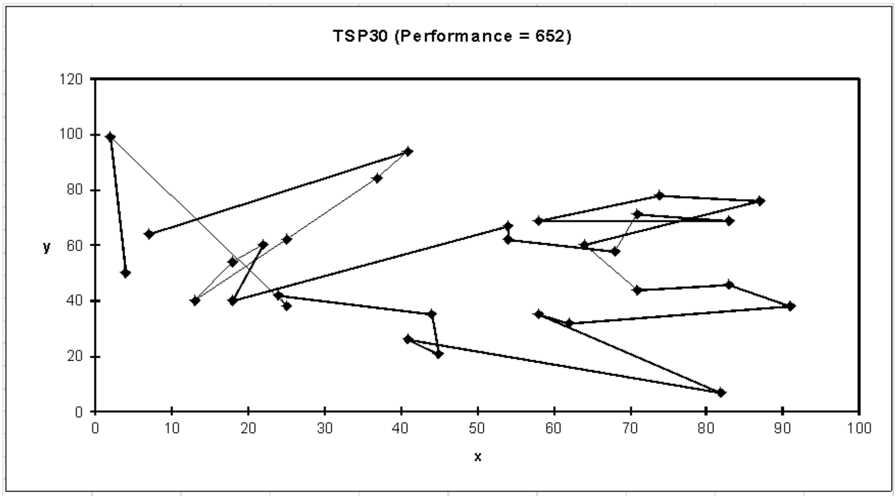
Source: Wendy Williams, GA: A Tutorial

Example TSP - progress



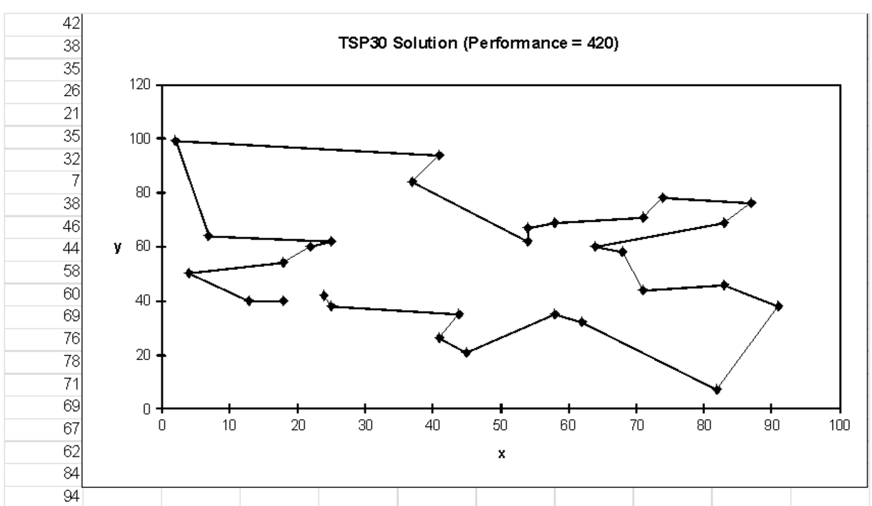
Source: Wendy Williams, GA: A Tutorial

Example TSP - progress



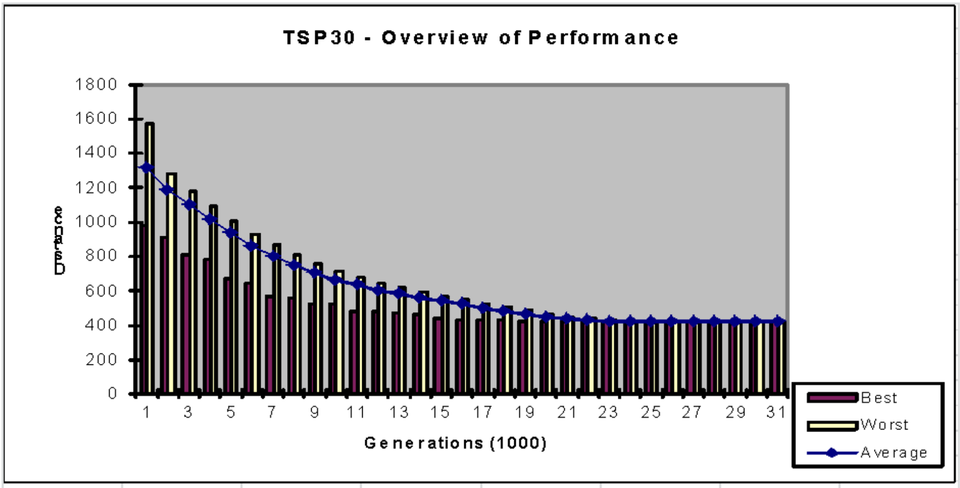
Source: Wendy Williams, GA: A Tutorial

Example TSP - progress



Source: Wendy Williams, GA: A Tutorial

Example TSP - Results



Source: Wendy Williams, GA: A Tutorial

Problem of GA

- Many parameters to be set
  - The number of individuals in a population?
  - Which selection operator?
  - Which crossover operator?
  - Which mutation operator?
  - Probabilities of selection, crossover, mutation?
  - Stopping criterion?

Application field

- NP-hard problems (Traveling Salesman Problem, ...)
  - Search
  - Optimization
  - Learning

Application of GA to transportation

- Genetic Fuzzy Systems (see Fuzzy systems)
  - **Genetic Algorithms** for Automated Tuning of Fuzzy Controllers
- Genetic Case-Based Reasoning (G-CBR)
- Multi-criteria transportation problems
- Vehicle Routing Problems
- Traveling Salesman Problem
- Optimization

SOFT COMPUTING - Final Remarks

- Soft computing is not a method to solve all problems
- We have to apply it carefully to the right set of problems!