Neural Networks

Prof. Dr. Rudolf Kruse

University of Magdeburg
Faculty of Computer Science
Magdeburg, Germany
rudolf.kruse@cs.uni-magdeburg.de

Allgemeines

Vorlesung: Neuronale Netze

Termin   Do 9:15-10:45 G29-307

Dozent Prof. Dr. Rudolf Kruse

Sprechstunde Mittwochs, 11-12 Uhr, Gebäude 29, Raum 008

bevorzugt erreichbar per Email: kruse@iws.cs.uni-magdeburg.de

Übungen: Neuronale Netze

Frank Rügheimer

Literatur

C. Borgelt, F. Klawonn, R. Kruse, D. Nauck,

Neuro-Fuzzy Systeme, 3.Auflage, Vieweg, 2003 (unter anderem)

Chapter 1:

Neural Networks and Computational Intelligence

Computational Intelligence (CI)

CI Core Technologies

- Neural Nets (NN)
- Fuzzy Logic (FL)
- Probabilistic Reasoning (PR)
- Genetic Algorithms (GA)
- Hybrid Systems

Related Technologies

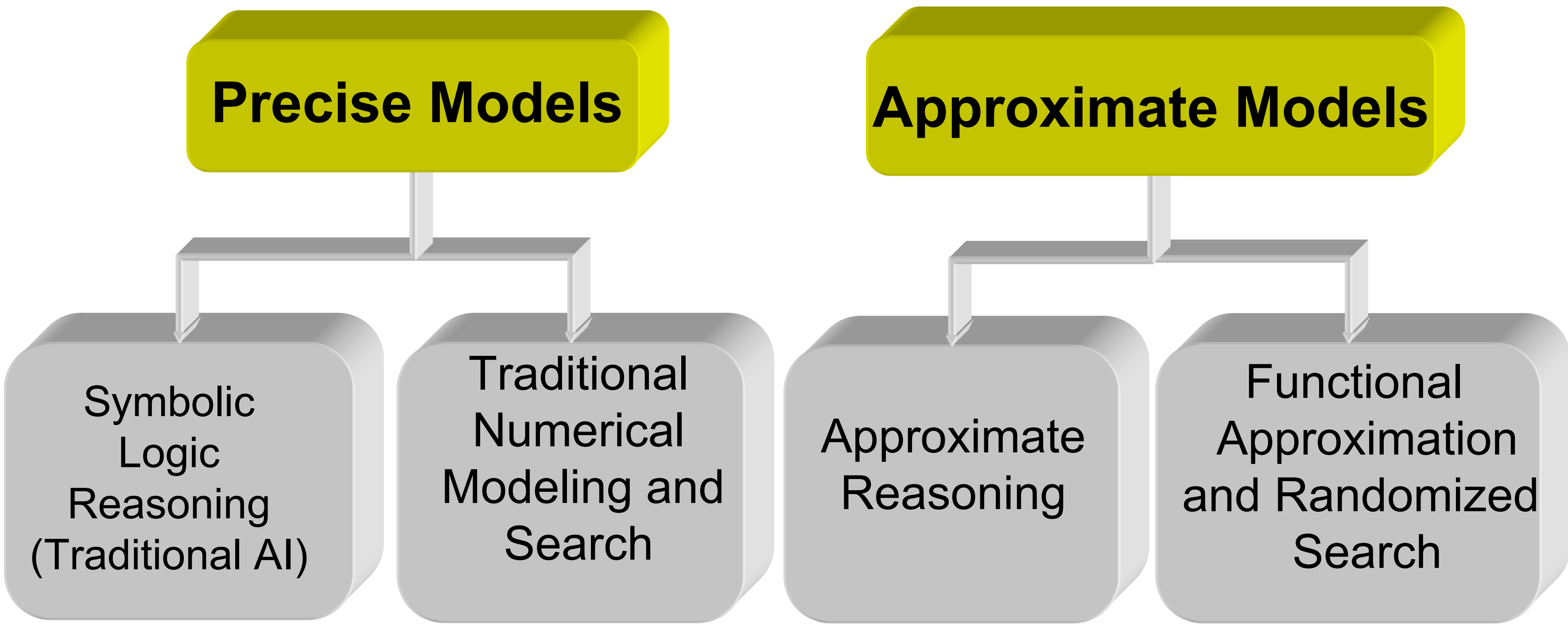
- Statistics (Stat.)
- Artificial Intelligence (AI):
  - Case-Based Reasoning (CBR)
  - Rule-Based Expert Systems (RBR)
  - Machine Learning (Induction Trees)
  - Bayesian Belief Networks (BBN)

Applications

- Classification
  - Monitoring/Anomaly Detection
  - Diagnostics
  - Prognostics
  - Configuration/Initialization
- Prediction
  - Quality Assessment
  - Equipment Life Estimation
- Scheduling
  - Time/Resource Assignments
- Control
  - Machine/Process Control
  - Process Initialization
  - Supervisory Control
- DSS/Auto-Decisioning
  - Cost/Risk Analysis
  - Revenue Optimization

Broad technology base and wide range of application tasks

Problem Solving Technologies



Computational Intelligence

Approximate Reasoning

Functional Approximation/ Randomized Search

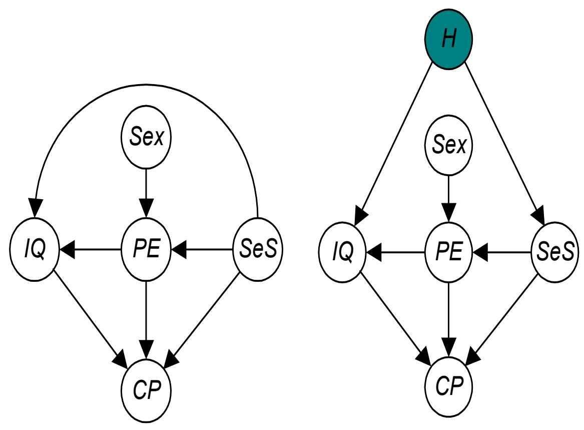
Probabilistic Models

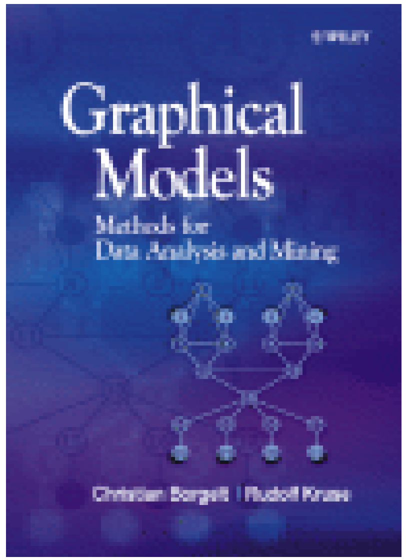
Multivalued & Fuzzy Logics

Neural Networks

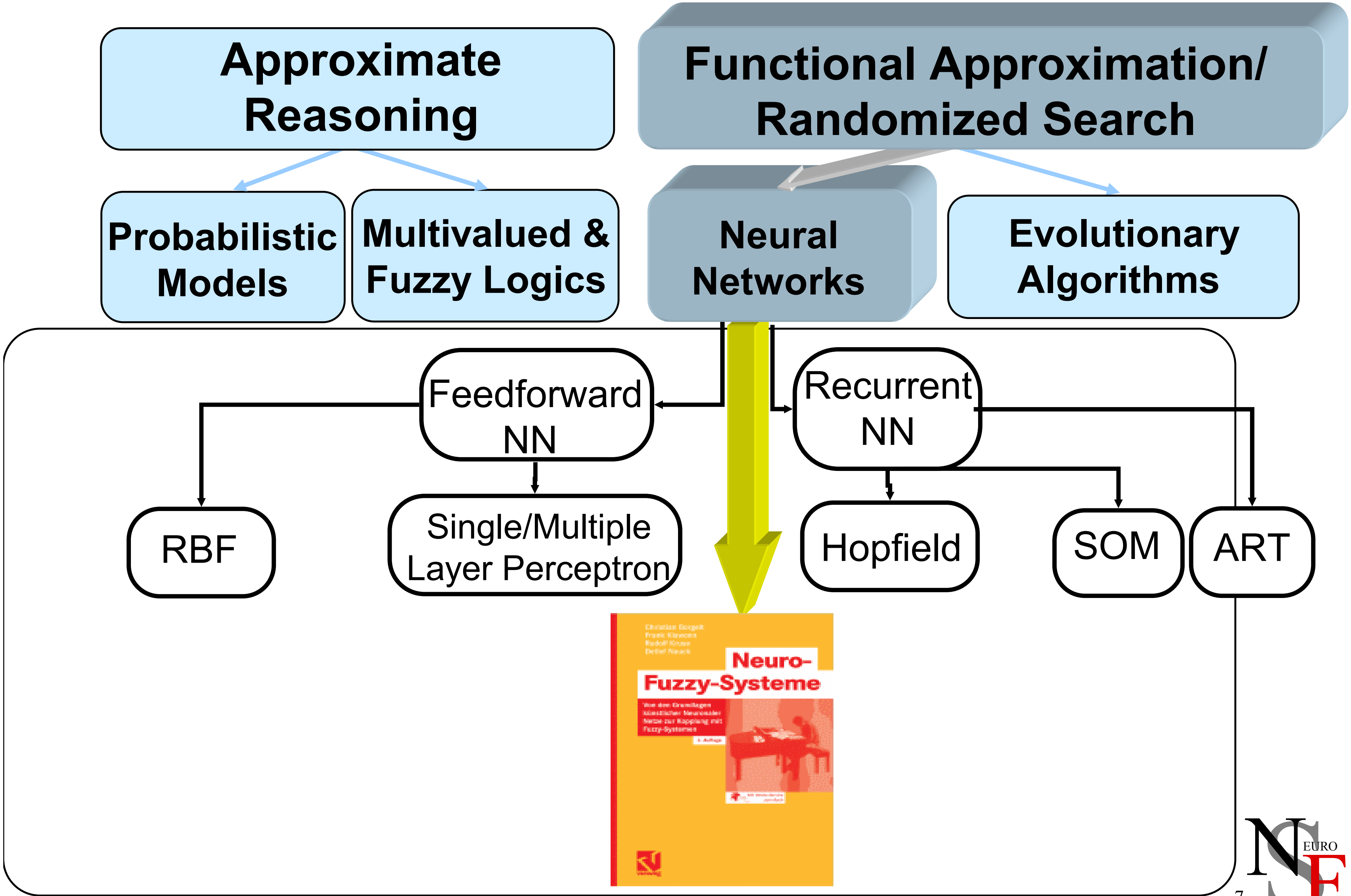
Evolutionary Algorithms

Graphical Models





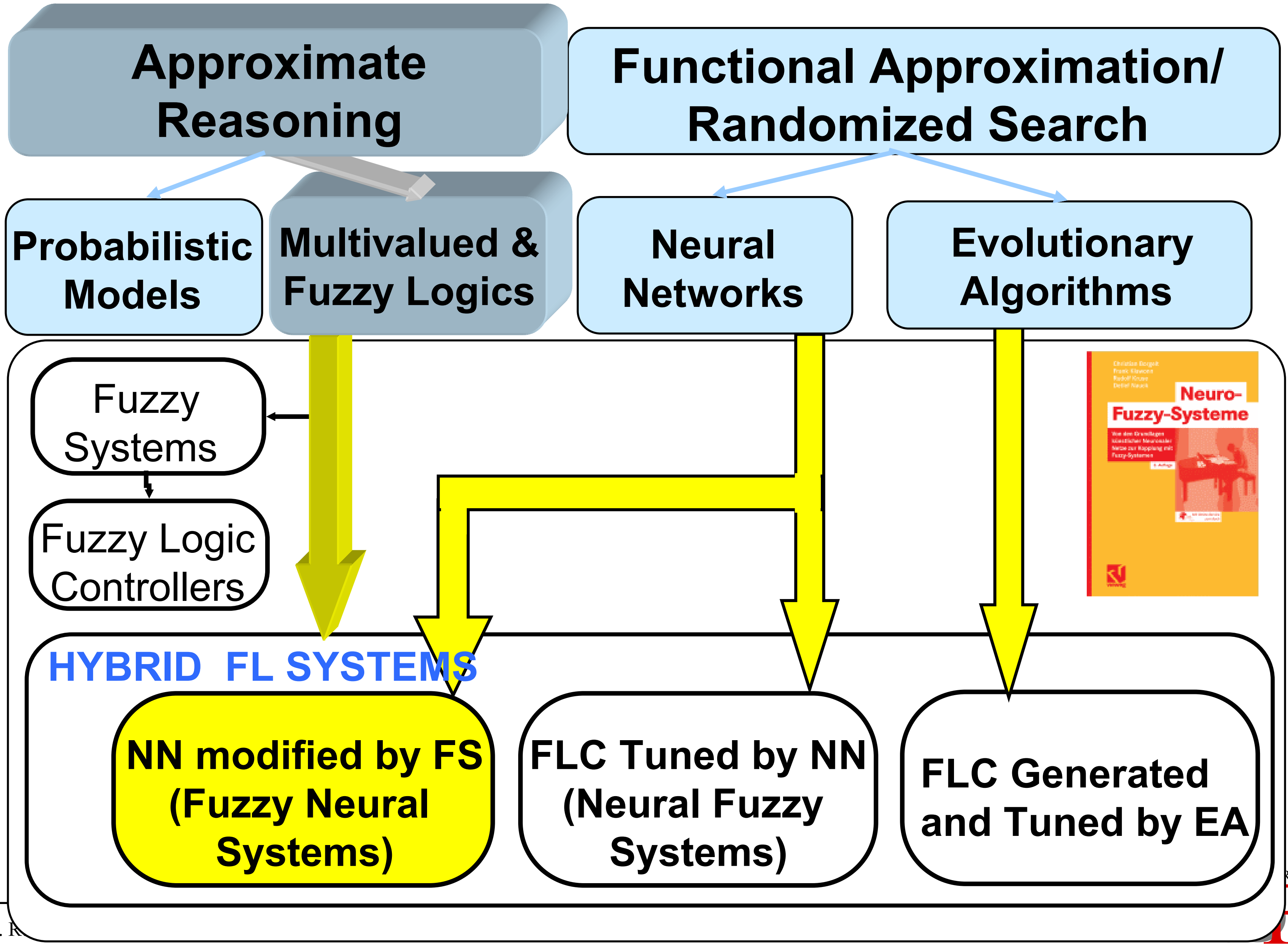
Computational Intelligence: Neural Networks



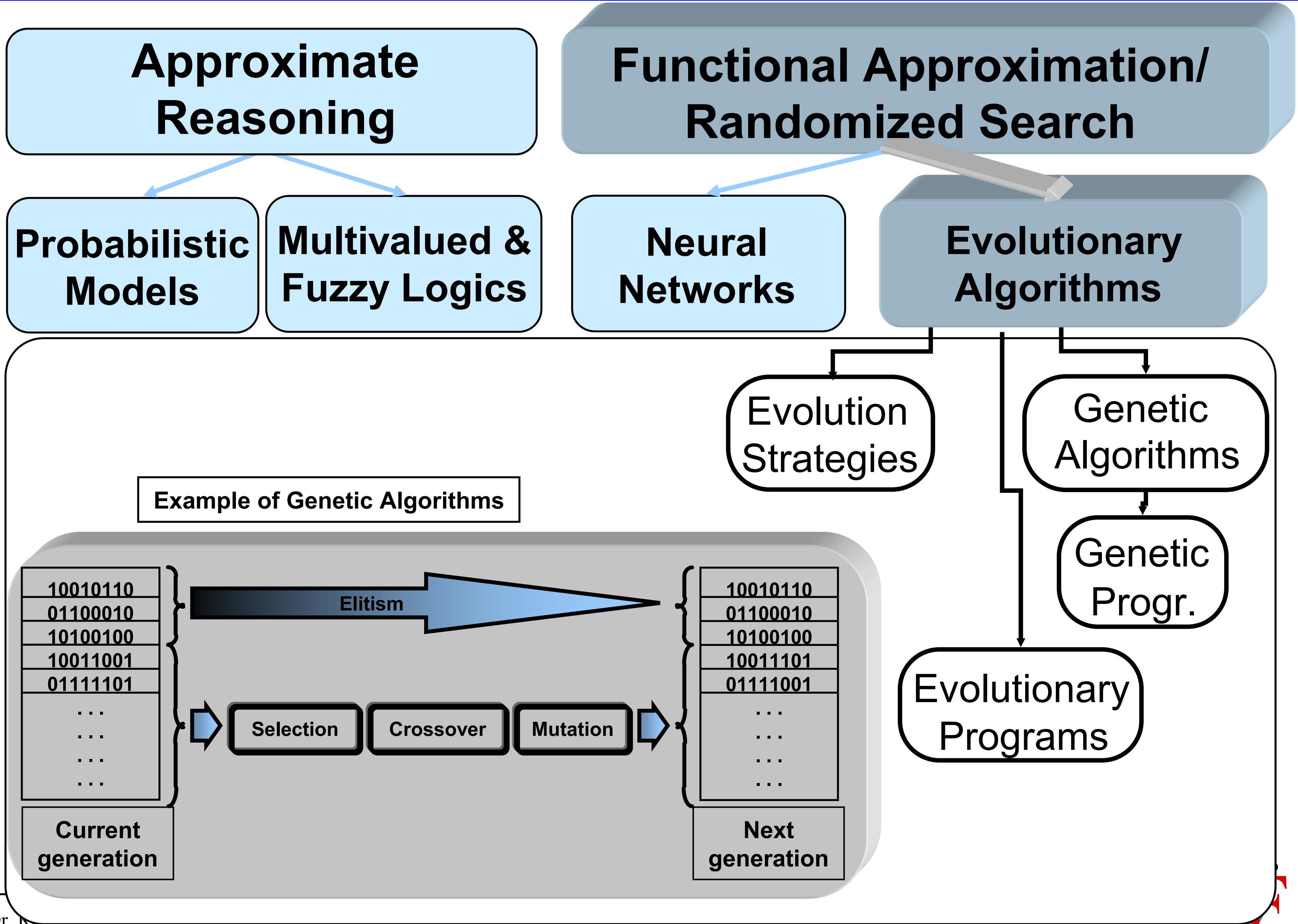
Types of Neural Networks

- **Introduction**
  Motivation, Biological Background
- **Threshold Logic Units**
  Definition, Geometric Interpretation, Limitations, Networks of TLUs, Training
- **General Neural Networks**
  Structure, Operation, Training
- **Multilayer Perceptrons**
  Definition, Function Approximation, Gradient Descent, Backpropagation, Variants, Sensitivity Analysis
- **Radial Basis Function Networks**
  Definition, Function Approximation, Initialization, Training, Generalized Version
- **Self-Organizing Maps**
  Definition, Learning Vector Quantization, Neighborhood of Output Neurons
- **Hopfield Networks**
  Definition, Convergence, Associative Memory, Solving Optimization Problems
- **Recurrent Neural Networks**
  Differential Equations, Vector Networks, Backpropagation through Time

Comp Int. : Hybrid Neuro-Fuzzy Systems



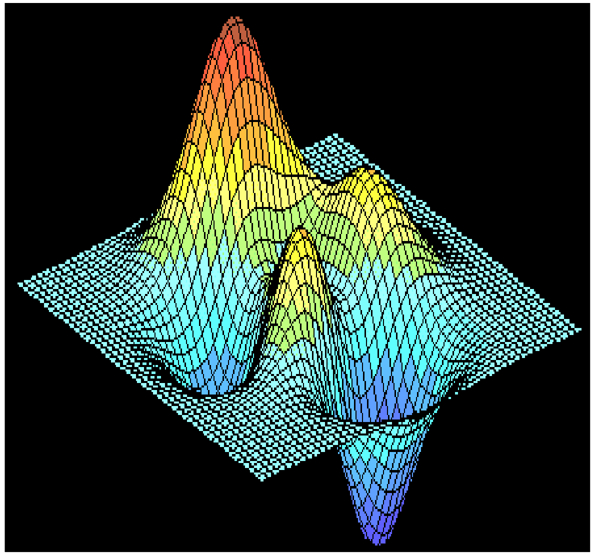
Computational Intelligence : EA Systems

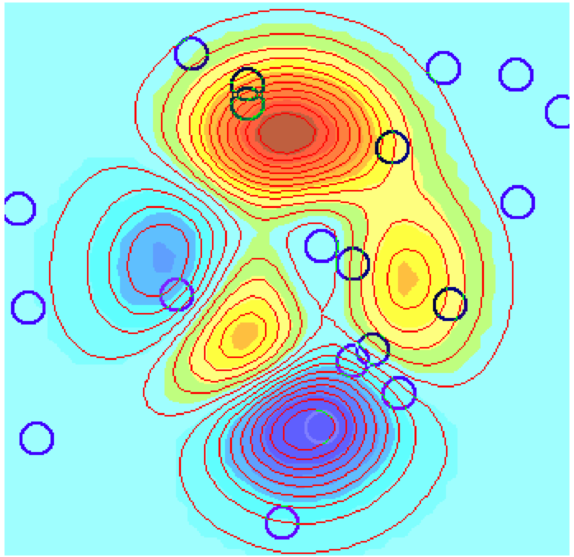


Evolutionary Algorithms: Scalar-Valued Fitness Function Optimization

- Example: Find the maximum of the function z(x,y)

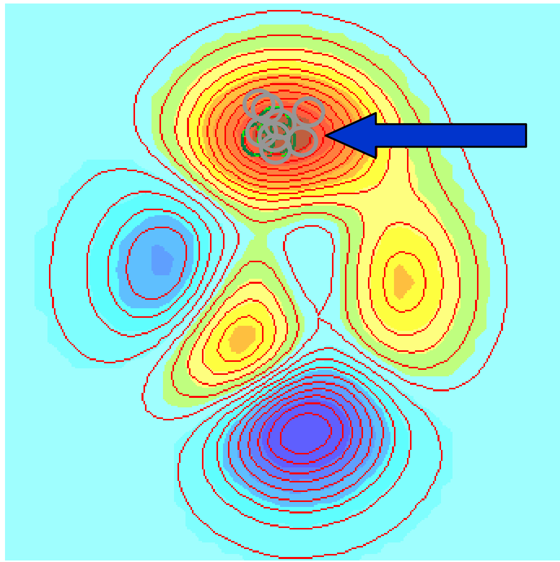
z = f(x, y) = 3*(1-x)^2*exp(-(x^2) - (y+1)^2) - 10*(x/5 - x^3 - y^5)*exp(-x^2-y^2) -1/3*exp(-(x+1)^2 - y^2).





Generation 0

Initialization of population providing a random sample of solution space

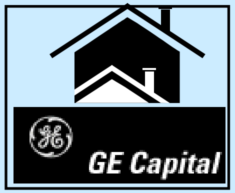


Generation 10

By evolving the individuals, we create a bias in the sampling and over-sample the best region(s) getting “close” to the optimal point(s)

Soft Computing Applications

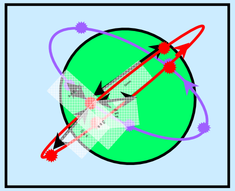
Appliances
• Preferred Service Contracts (Stat.)
• Call Center Support (CBR)



Capital Services
• Mortgage Collateral Evaluation (Fusion/FL/CBR)

Financial Assurance
• GEFA LTC Preferred Customer (Stat./NN)
• GEFA Fixed Life Digital Underwriter (Stat, CBR, FL, GA)

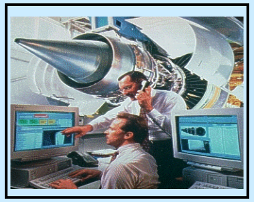
Plastics
• Automated Color Matching (CBR)



LM Fed. Systems
• Scheduling Maintenance for Constellation of Satellites (GA)

LM ORSS
• Vessel Management Syst. (AI/GA)

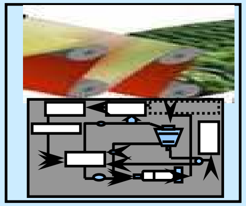
Medical Systems
• SPT Auto Analysis for MRI (FL)
• Reverse Engineering of Picker (FL)
• FE Analysis tool (FL)
• X-Ray error Logs Analysis (CBR)



Aircraft Engines
• Center for Remote Diagn. (CBR)
• Customer Response Center (CBR)
• Anomaly Detection (FL/Stat.)
• IMATE - Maintenance Advisor (NN/FL)
• Resolver Drift - Sensor Fusion (FL)

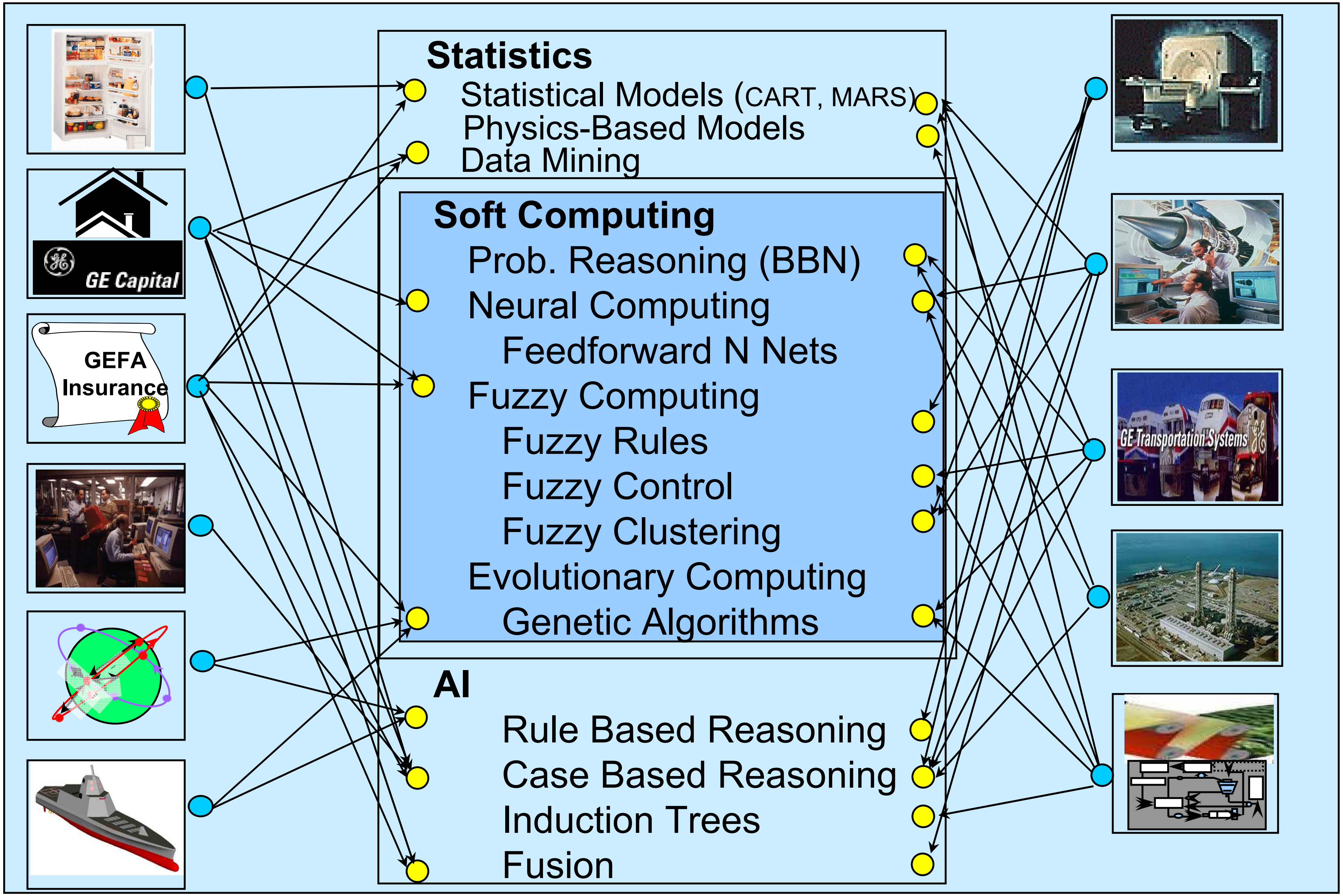
Transportation Systems
• Log from Transportation DB (CBR)
• Prototype Train Handling Cntrl. (FL/GA)
• Prototype Trend Analysis (Stat.)
• Embedded/Remote Diagnostics (BBN)

Power Gen. Systems
• Remote Anomaly Detection (Stat.)
• Embedded/Remote Diagnostics (BBN)
• Call Center Problem/Solution (CBR)



Industrial Systems
• Paper Web Breakage Prediction (NN/Stat./Induction)
• Control Mixing of Cement (FL/GA)

Enabling Soft Computing and Related Technologies



Chapter 2:

Threshold Logic Units (Perceptrons)

Motivation: Why (Artificial) Neural Networks?

- (Neuro-)Biology / (Neuro-)Physiology / Psychology:
  - Exploit similarity to real (biological) neural networks.
  - Build models to understand nerve and brain operation by simulation.
- Computer Science / Engineering / Economics
  - Mimic certain cognitive capabilities of human beings.
  - Solve learning/adaptation, prediction, and optimization problems.
- Physics / Chemistry
  - Use neural network models to describe physical phenomena.
  - Special case: spin glasses (alloys of magnetic and non-magnetic metals).

Motivation: Why Neural Networks in AI?

Physical-Symbol System Hypothesis [Newell and Simon 1976]

A physical-symbol system has the necessary and sufficient means
for general intelligent action.

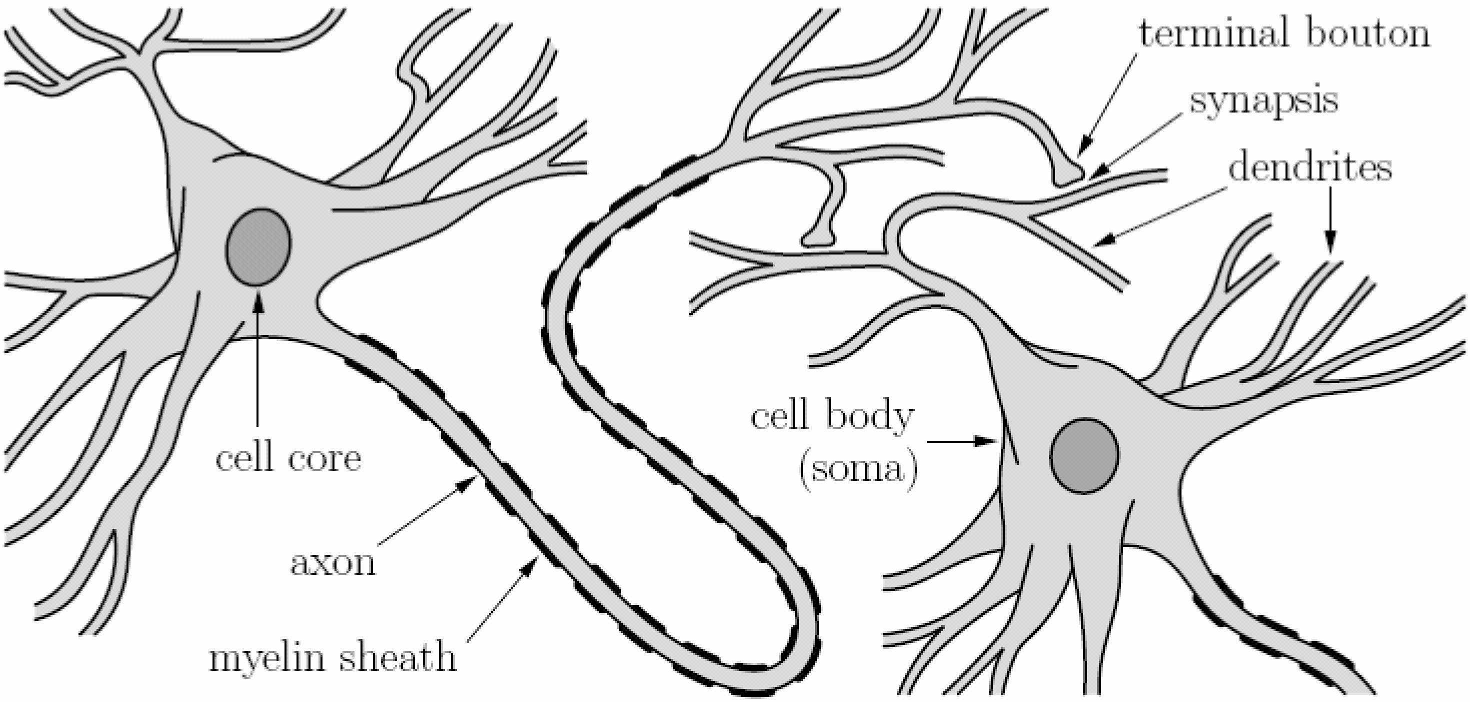
Neural networks process simple signals, not symbols.

So why study neural networks in Artificial Intelligence?

- Symbol-based representations work well for inference tasks, but fairly bad for perception tasks.
- Symbol-based expert systems tend to get slower with growing knowledge, human experts tend to get faster.
- Neural networks allow for highly parallel information processing.
- There are several successful applications in industry and finance.

Biological Background

Structure of a prototypical biological neuron



Biological Background

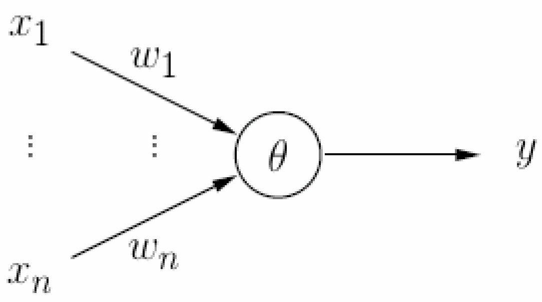
(Very) simplified description of neural information processing

- Axon terminal releases chemicals, called **neurotransmitters**.
- These act on the membrane of the receptor dendrite to change its polarization. (The inside is usually 70mV more negative than the outside.)
- Decrease in potential difference: **excitatory** synapse
  Increase in potential difference: **inhibitory** synapse
- If there is enough net excitatory input, the axon is depolarized.
- The resulting **action potential** travels along the axon. (Speed depends on the degree to which the axon is covered with myelin).
- When the action potential reaches the terminal boutons, it triggers the release of neurotransmitters.

Threshold Logic Units

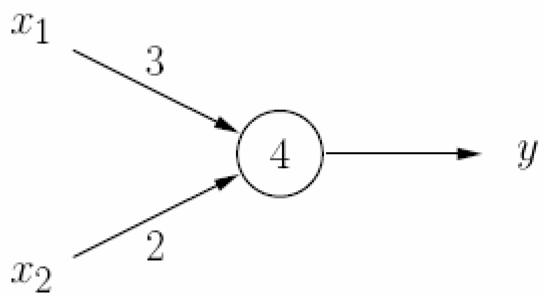
A **Threshold Logic Unit (TLU)** is a processing unit for numbers with n inputs $x_1, \ldots, x_n$ and one output y. The unit has a **threshold** θ and each input x_i is associated with a **weight** w_i. A threshold logic unit computes the function

$$y = \begin{cases} 1, & \text{if} \quad \vec{x}\vec{w} = \sum_{i=1}^{n} w_i x_i \geq \theta, \\ 0, & \text{otherwise.} \end{cases}$$



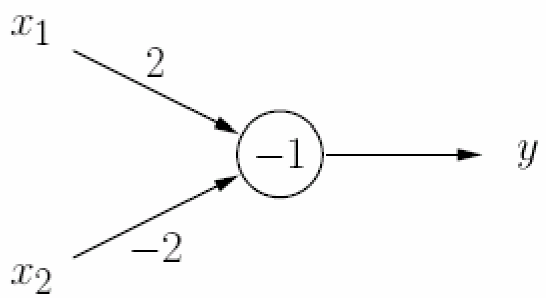
Threshold Logic Units: Examples

Threshold logic unit for the conjunction $x_1 \wedge x_2$.



x_1	x_2	$3x_1 + 2x_2$	y
0	0	0	0
1	0	3	0
0	1	2	0
1	1	5	1

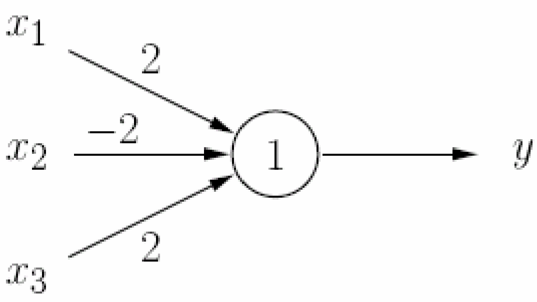
Threshold logic unit for the implication $x_2 \rightarrow x_1$.



x_1	x_2	$2x_1 - 2x_2$	y
0	0	0	1
1	0	2	1
0	1	-2	0
1	1	0	1

Threshold Logic Units: Examples

Threshold logic unit for $(x_1 \wedge \overline{x_2}) \vee (x_1 \wedge x_3) \vee (\overline{x_2} \wedge x_3)$.



x_1	x_2	x_3	$\sum_i w_i x_i$	y
0	0	0	0	0
1	0	0	2	1
0	1	0	-2	0
1	1	0	0	0
0	0	1	2	1
1	0	1	4	1
0	1	1	0	0
1	1	1	2	1

Threshold Logic Units: Geometric Interpretation

Review of line representations

Straight lines are usually represented in one of the following forms:

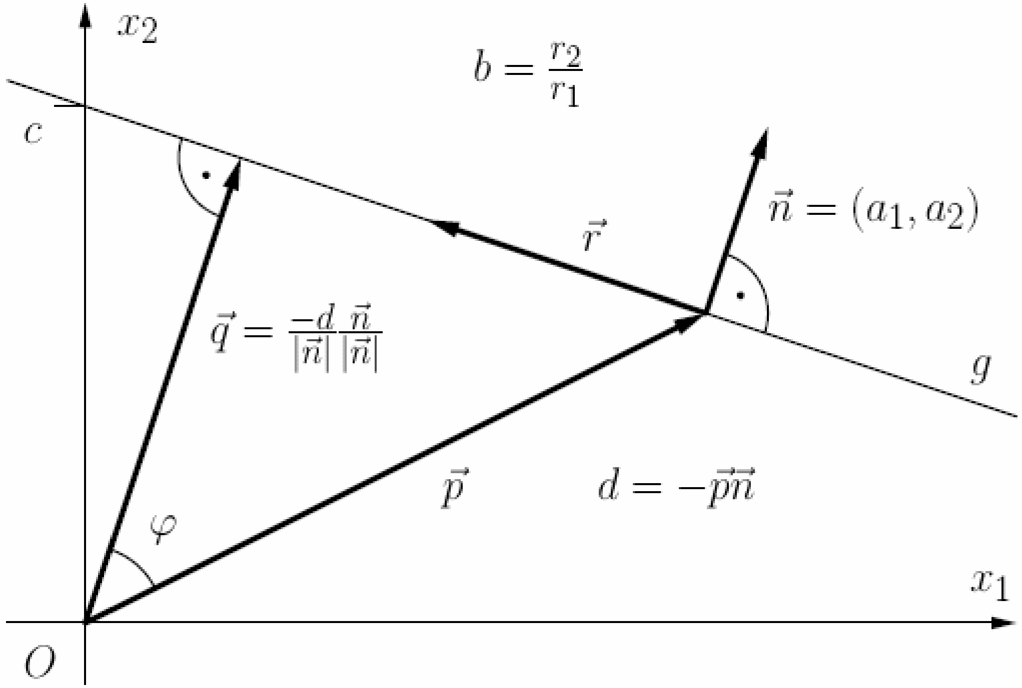
Explicit Form:	$g \equiv$	$x_2 = bx_1 + c$
Implicit Form:	$g \equiv$	$a_1x_1 + a_2x_2 + d = 0$
Point-Direction Form:	$g \equiv$	$\vec{x} = \vec{p} + k\vec{r}$
Normal Form:	$g \equiv$	$(\vec{x} - \vec{p})\vec{n} = 0$

with the parameters:

b :	Gradient of the line
c :	Section of the x_2 axis
$\vec{p}$:	Vector of a point of the line (base vector)
$\vec{r}$:	Direction vector of the line
$\vec{n}$:	Normal vector of the line

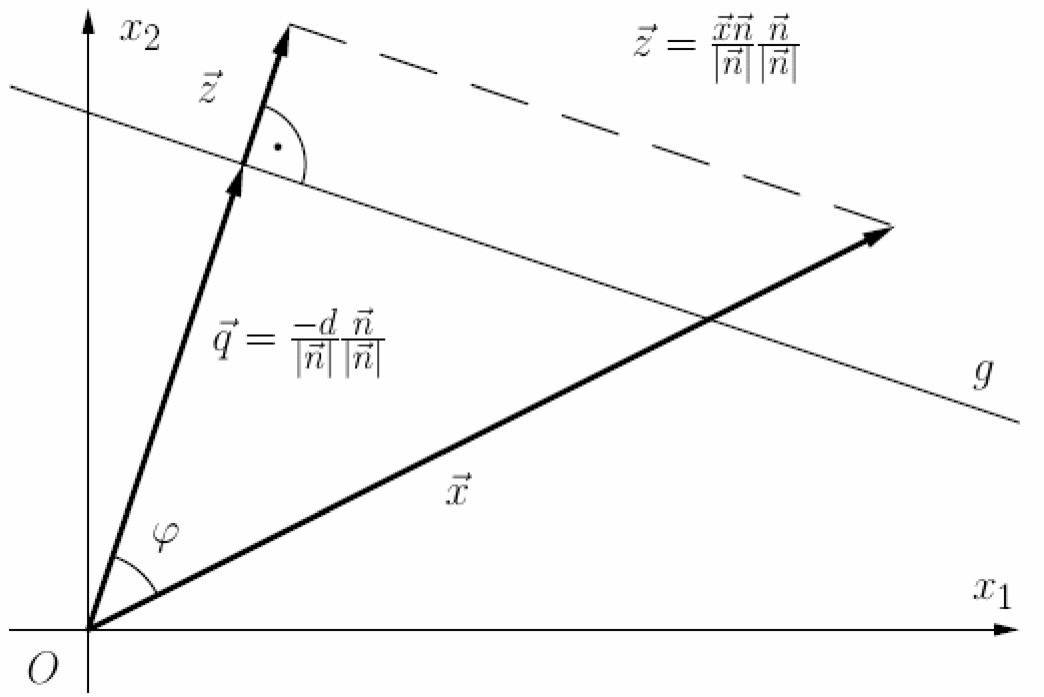
Threshold Logic Units: Geometric Interpretation

A straight line and its defining parameters.



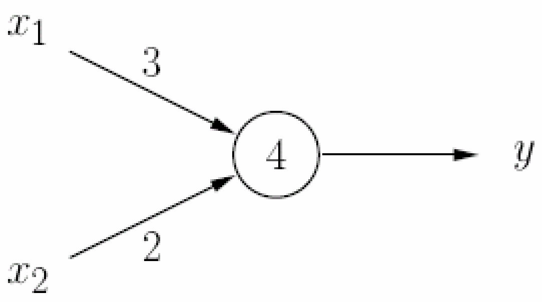
Threshold Logic Units: Geometric Interpretation

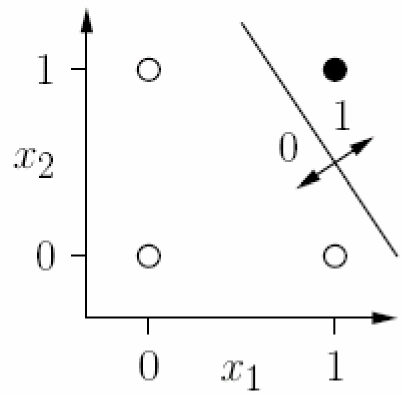
How to determine the side on which a point $\vec{x}$ lies.



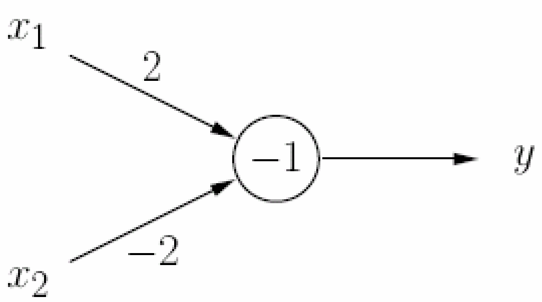
Threshold Logic Units: Geometric Interpretation

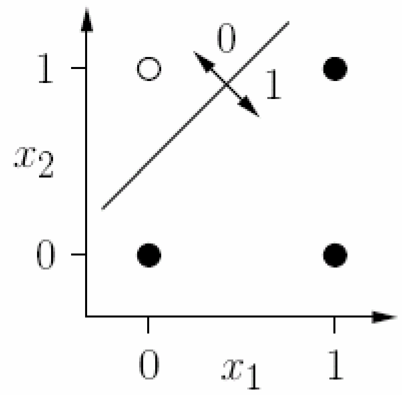
Threshold logic unit for $x_1 \wedge x_2$.





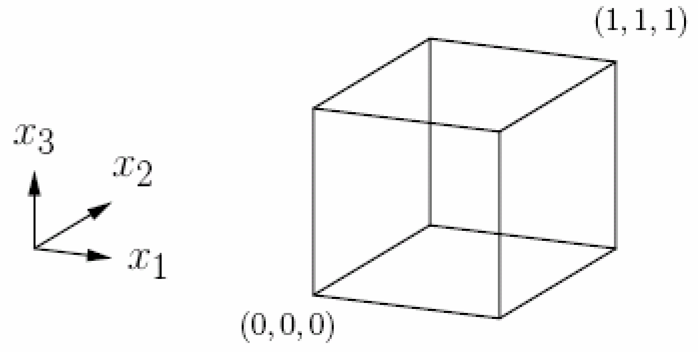
A threshold logic unit for $x_2 \rightarrow x_1$.



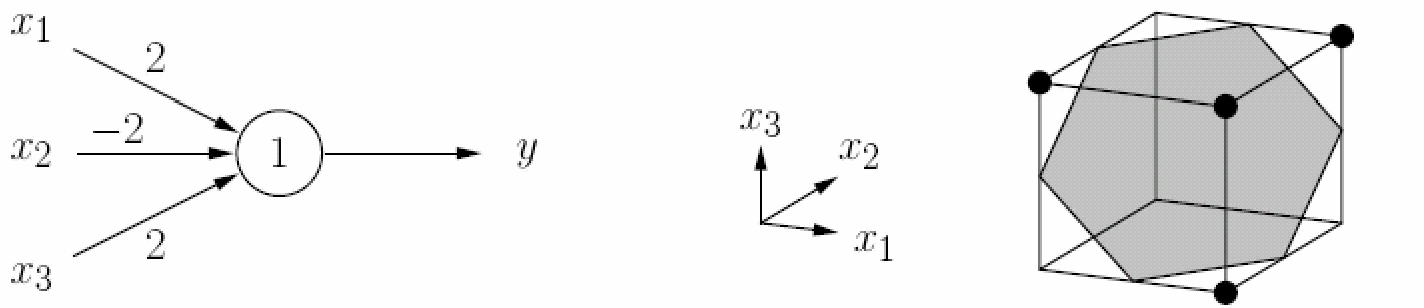


Threshold Logic Units: Geometric Interpretation

Visualization of 3-dimensional Boolean functions:



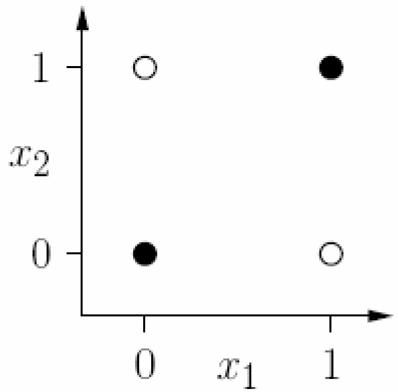
Threshold logic unit for $(x_1 \wedge \overline{x_2}) \vee (x_1 \wedge x_3) \vee (\overline{x_2} \wedge x_3)$.



Threshold Logic Units: Limitations

The biimplication problem $x_1 \leftrightarrow x_2$: There is no separating line.

x_1	x_2	y
0	0	1
1	0	0
0	1	0
1	1	1



Formal proof by *reductio ad absurdum*:

$$
\begin{array}{lrll}
\text{since } (0,0) \mapsto 1: & 0 & \geq \theta, & (1) \\
\text{since } (1,0) \mapsto 0: & w_1 & < \theta, & (2) \\
\text{since } (0,1) \mapsto 0: & w_2 & < \theta, & (3) \\
\text{since } (1,1) \mapsto 1: & w_1 + w_2 & \geq \theta. & (4)
\end{array}
$$

(2) and (3): $w_1 + w_2 < 2\theta$. With (4): $2\theta > \theta$, or $\theta > 0$. Contradiction to (1).

Threshold Logic Units: Limitations

Total number and number of linearly separable Boolean functions.
([Widner 1960] as cited in [Zell 1994])

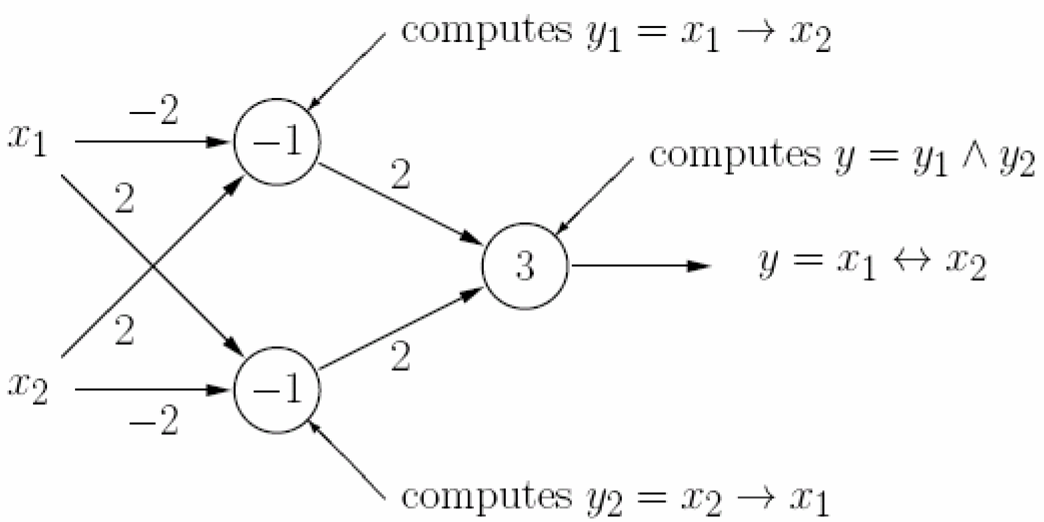
inputs	Boolean functions	linearly separable functions
1	4	4
2	16	14
3	256	104
4	65536	1774
5	$4.3 \cdot 10^9$	94572
6	$1.8 \cdot 10^{19}$	$5.0 \cdot 10^6$

- For many inputs a threshold logic unit can compute almost no functions.
- Networks of threshold logic units are needed to overcome the limitations.

Networks of Threshold Logic Units

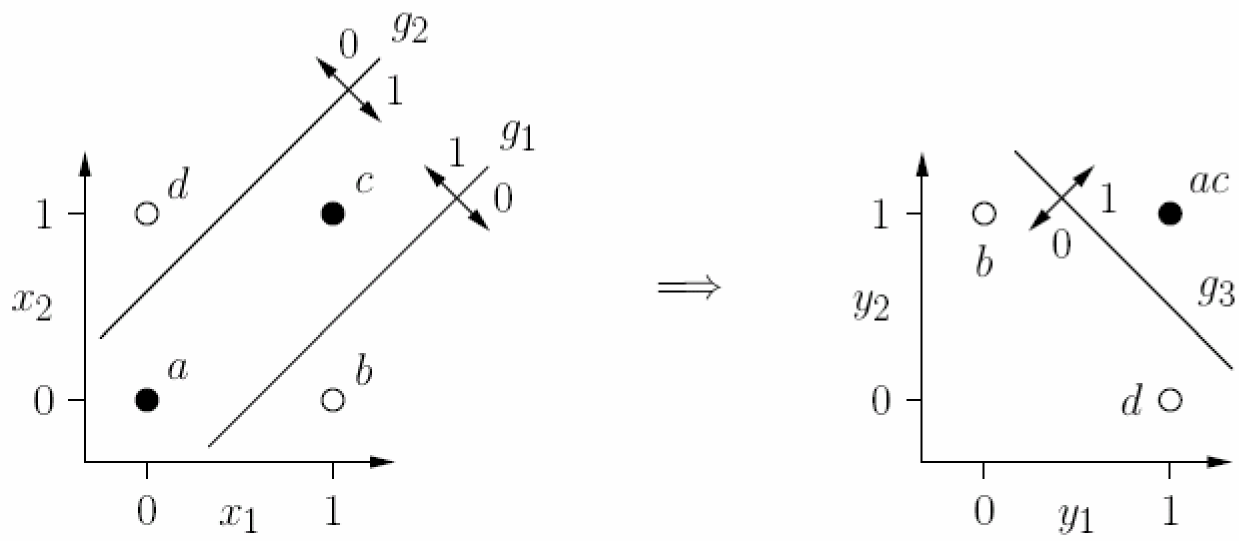
Solving the biimplication problem with a network.

Idea: Logical Decomposition $\quad x_1 \leftrightarrow x_2 \equiv (x_1 \to x_2) \wedge (x_2 \to x_1)$



Networks of Threshold Logic Units

Solving the biimplication problem: Geometric interpretation



- The first layer computes new Boolean coordinates for the points.
- After the coordinate transformation the problem is linearly separable.

Representing Arbitrary Boolean Functions

Let $y = f(x_1, \ldots, x_n)$ be a Boolean function of n variables.

(i) Represent $f(x_1, \ldots, x_n)$ in disjunctive normal form. That is, determine $D_f = K_1 \vee \ldots \vee K_m$, where all K_j are conjunctions of n literals, i.e., $K_j = l_{j1} \wedge \ldots \wedge l_{jn}$ with $l_{ji} = x_i$ (positive literal) or $l_{ji} = \neg x_i$ (negative literal).

(ii) Create a neuron for each conjunction K_j of the disjunctive normal form (having n inputs — one input for each variable), where

$$w_{ji} = \begin{cases} 2, & \text{if } l_{ji} = x_i, \\ -2, & \text{if } l_{ji} = \neg x_i, \end{cases} \qquad \text{and} \qquad \theta_j = n - 1 + \frac{1}{2}\sum_{i=1}^{n} w_{ji}.$$

(iii) Create an output neuron (having m inputs — one input for each neuron that was created in step (ii)), where

$$w_{(n+1)k} = 2, \quad k = 1, \ldots, m, \qquad \text{and} \qquad \theta_{n+1} = 1.$$