Advanced Automatic Control MDP 444

If you have a smart project, you can say "I'm an engineer"

Lecture 10

Staff boarder

Prof. Dr. Mostafa Zaki Zahran

Dr. Mostafa Elsayed Abdelmonem

Advanced Automatic Control
MDP 444

- **Lecture aims:**
  - Solve simple problems using the Artificial Intelligent.
  - Formulate advanced problems for neural networks.

Introduction

- A **hybrid intelligent system** is one that combines at least two intelligent technologies. For example, combining a neural network with a fuzzy system results in a hybrid neuro-fuzzy system.

- The combination of probabilistic reasoning, fuzzy logic, neural networks and evolutionary computation forms the core of **soft computing**, an emerging approach to building hybrid intelligent systems capable of reasoning and learning in an uncertain and imprecise environment.

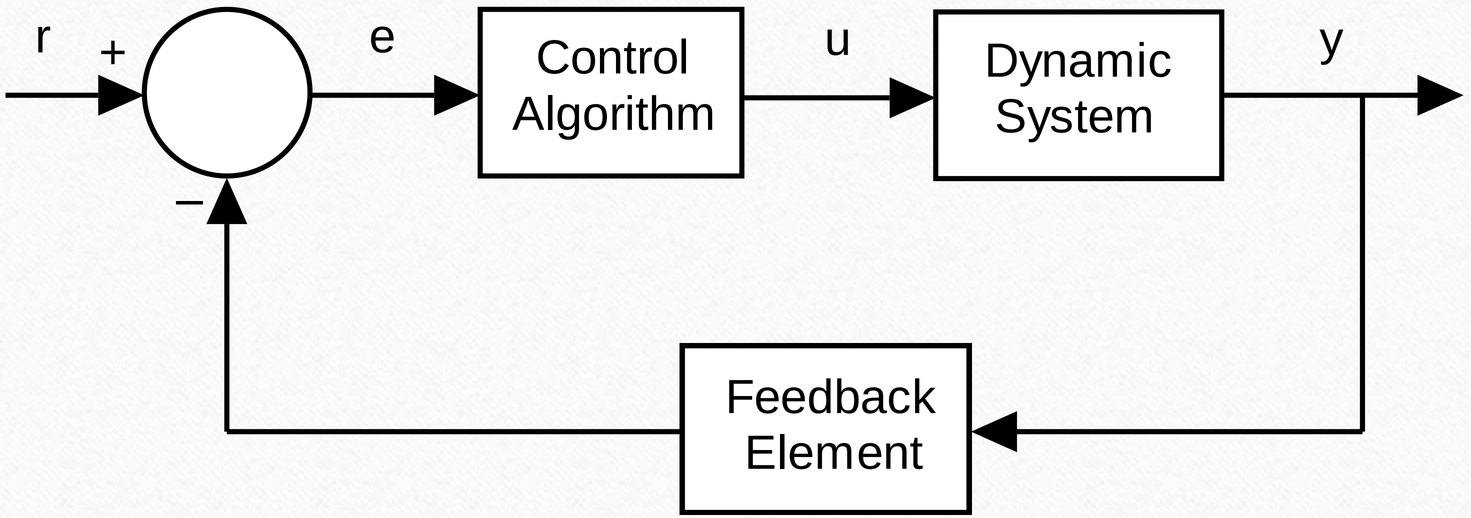
Introduction

- Although words are less precise than numbers, precision carries a high cost. We use words when there is a tolerance for imprecision.

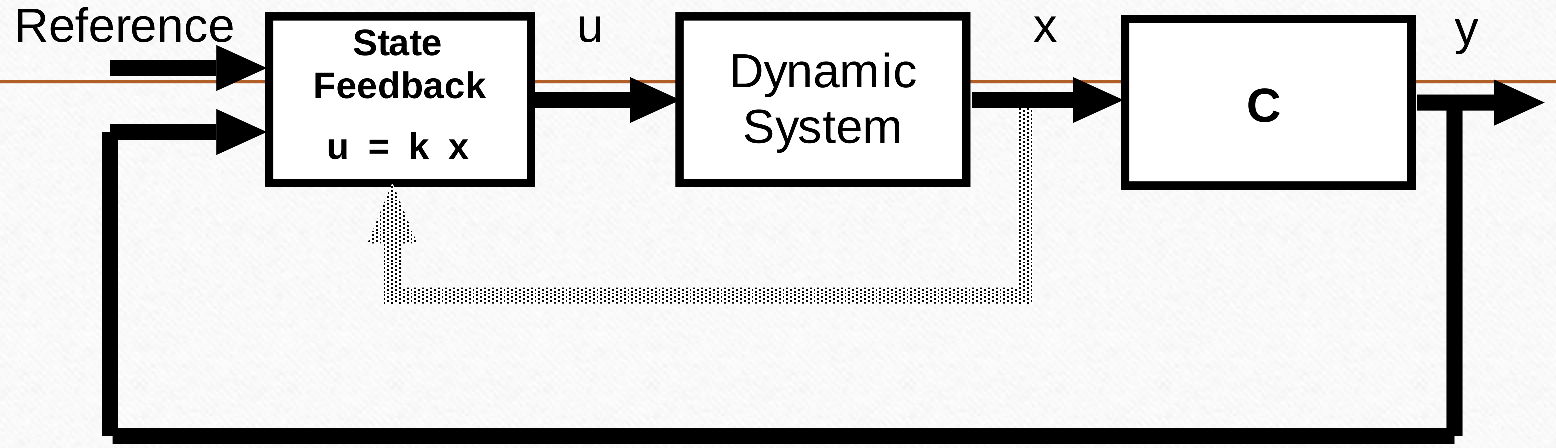
- Soft computing exploits the tolerance for uncertainty and imprecision to achieve greater tractability and robustness, and lower the cost of solutions.

- We also use words when the available data is not precise enough to use numbers. This is often the case with complex problems, and while "hard" computing fails to produce any solution, soft computing is still capable of finding good solutions.

Classical Feedback Control



Modern Control

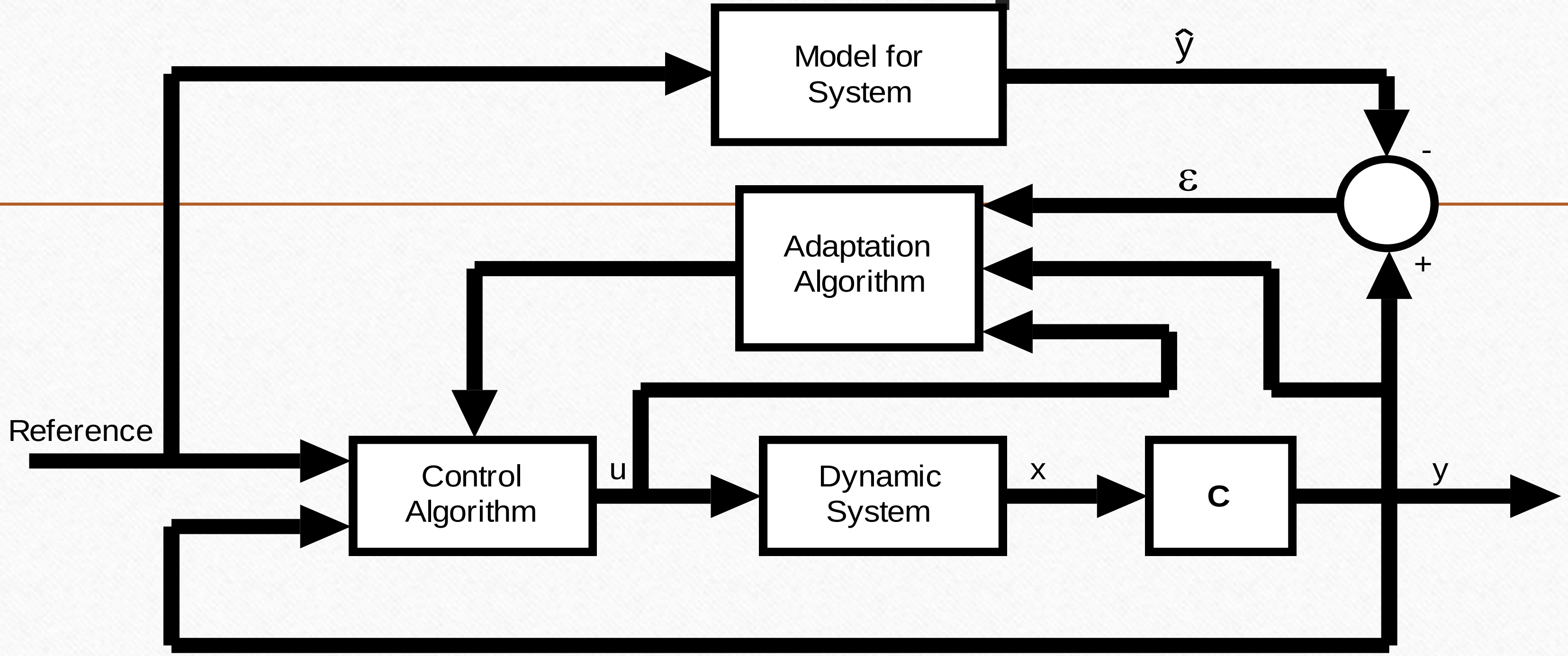


$$\text{State: } \frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t), \quad \text{Output: } \mathbf{y}(t) = \mathbf{Cx}(t)$$

$$\text{Performance Index: } J = \int_0^t \mathbf{x}'(\tau)\mathbf{Rx}(\tau) + \mathbf{u}'(\tau)\mathbf{Qu}(\tau)\, d\tau$$

$$\text{Control: } \quad \mathbf{u} = \mathbf{k\,x}\,; \quad \mathbf{k = f(A,B,C,J,Q,R)}$$

Model Reference Adaptive Control



State: $\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$, Output: $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

Performance Index: $J = \int_0^t \mathbf{x}'(\tau)\mathbf{R}\mathbf{x}(\tau) + \mathbf{u}'(\tau)\mathbf{Q}\mathbf{u}(\tau)\, d\tau$

Control: $\mathbf{u} = \mathbf{k}(\theta, \mathbf{R}, \mathbf{Q})\, \mathbf{x}$

Prediction: $\hat{y} = \theta^T\phi$, $\phi^T = [y(k), y(k\text{-}1), ..u(k), u(k\text{-}1)...]$

Adaptation: $\theta^i = \theta^{i-1} + \mathbf{P}[y - \hat{y}]$, $\theta^T = [a_1, a_2, ... b_1, b_2, ...]$

Introduction – Making the Right Choice

- Lotfi Zadeh is reputed to have said that a good hybrid would be "British Police, German Mechanics, French Cuisine, Swiss Banking and Italian Love".
- But "British Cuisine, German Police, French Mechanics, Italian Banking and Swiss Love" would be a bad one.
- Likewise, a hybrid intelligent system can be good or bad – it depends on which components constitute the hybrid.
- So our goal is to select the right components for building a good hybrid system.

AI Tool Comparison

	ES	*FS*	*NN*	*GA*
Knowledge representation	rather good	good	bad	rather bad
Uncertainty tolerance	rather good	good	good	good
Imprecision tolerance	bad	good	good	good
Adaptability	bad	rather bad	good	good
Learning ability	bad	bad	good	good
Explanation ability	good	good	bad	rather bad
Knowledge discovery and data mining	bad	rather bad	good	rather good
Maintainability	bad	rather good	good	rather good

* The terms used for grading are:

- bad, - rather bad, - rather good and - good

Neural Network

Neural Networks -
A comprehensive foundation

Simon Haykin

Prentice-Hall, 1998

2nd edition

Neural Expert Systems

- Expert systems rely on logical inferences and decision trees and focus on modeling human reasoning. Neural networks rely on parallel data processing and focus on modeling a human brain.

- Expert systems treat the brain as a black-box. Neural networks look at its structure and functions, particularly at its ability to learn.

- Knowledge in a rule-based expert system is represented by IF-THEN production rules. Knowledge in neural networks is stored as synaptic weights between neurons.

Neural Expert Systems

- In expert systems, knowledge can be divided into individual rules and the user can see and understand the piece of knowledge applied by the system.

- In neural networks, one cannot select a single synaptic weight as a discrete piece of knowledge. Here knowledge is embedded in the entire network; it cannot be broken into individual pieces, and any change of a synaptic weight may lead to unpredictable results.

- A neural network is, in fact, a **black-box** for its user.

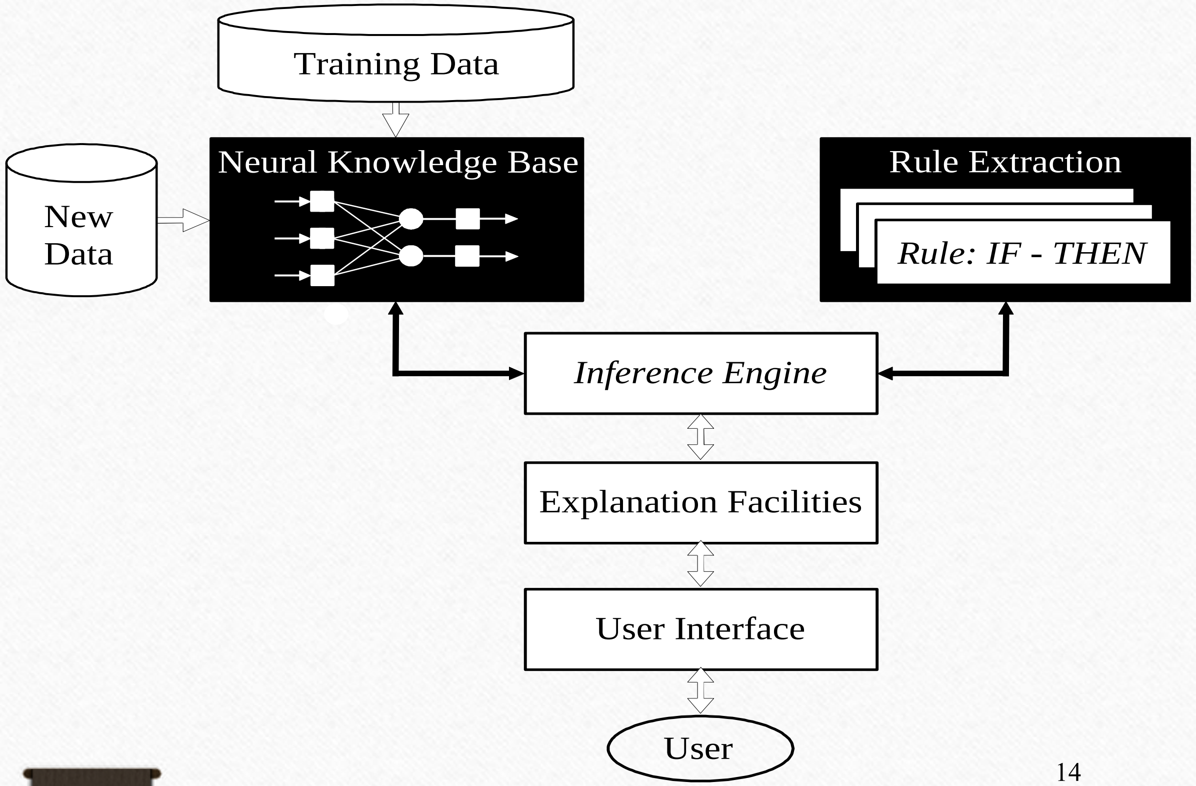
Neural Expert Systems

- Can we combine advantages of expert systems and neural networks to create a more powerful and effective expert system?

- A hybrid system that combines a neural network and a rule-based expert system is called a **neural expert system** (or a **connectionist expert system**).

Neural Expert Systems

The heart of a neural expert system is the **inference engine**. It controls the information flow in the system and initiates inference over the neural knowledge base. A neural inference engine also ensures **approximate reasoning**.



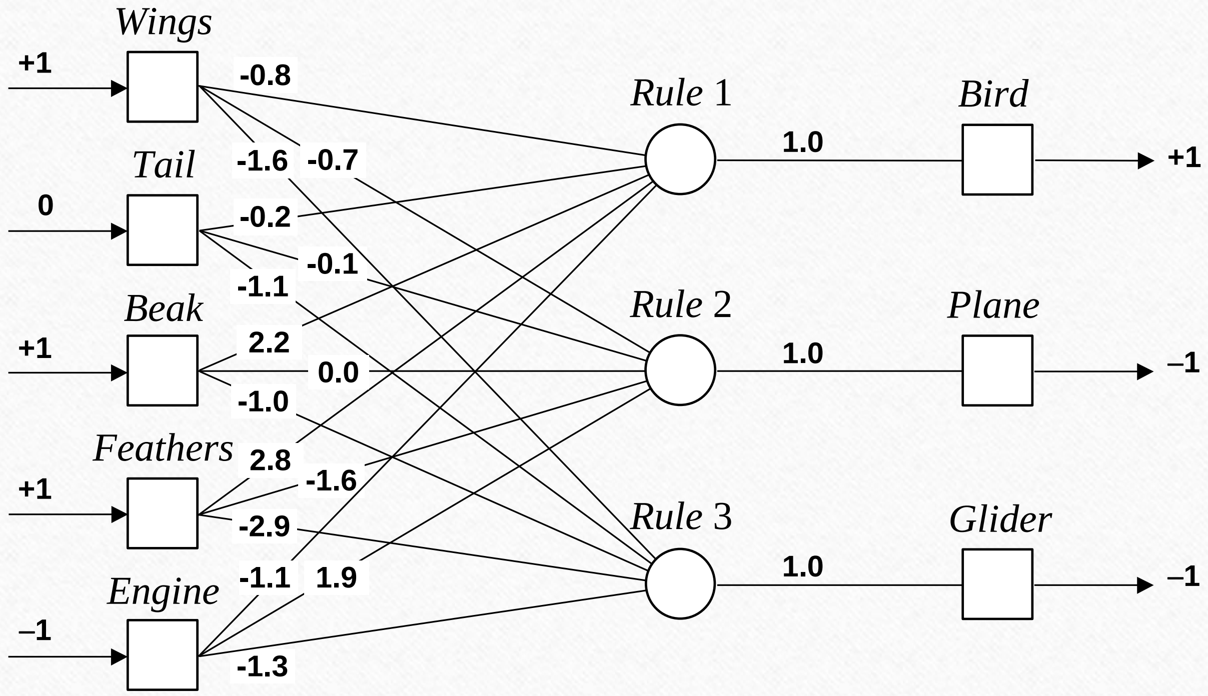
Approximate Reasoning

- In a rule-based expert system, the inference engine compares the condition part of each rule with data given in the database. When the IF part of the rule matches the data in the database, the rule is fired and its THEN part is executed. The **precise matching** is required (inference engine cannot cope with noisy or incomplete data).

- Neural expert systems use a trained neural network in place of the knowledge base. The input data does not have to precisely match the data that was used in network training. This ability is called **approximate reasoning**.

Rule Extraction

- Neurons in the network are connected by links, each of which has a numerical weight attached to it.
- The weights in a trained neural network determine the strength or importance of the associated neuron inputs.



Neuro-Fuzzy Systems

- Fuzzy logic and neural networks are natural complementary tools in building intelligent systems.

- While neural networks are low-level computational structures that perform well when dealing with raw data, fuzzy logic deals with reasoning on a higher level, using linguistic information acquired from domain experts.

- However, fuzzy systems lack the ability to learn and cannot adjust themselves to a new environment. On the other hand, although neural networks can learn, they are opaque to the user.

Neuro-Fuzzy Systems

- Integrated neuro-fuzzy systems can combine the parallel computation and learning abilities of neural networks with the human-like knowledge representation and explanation abilities of fuzzy systems.

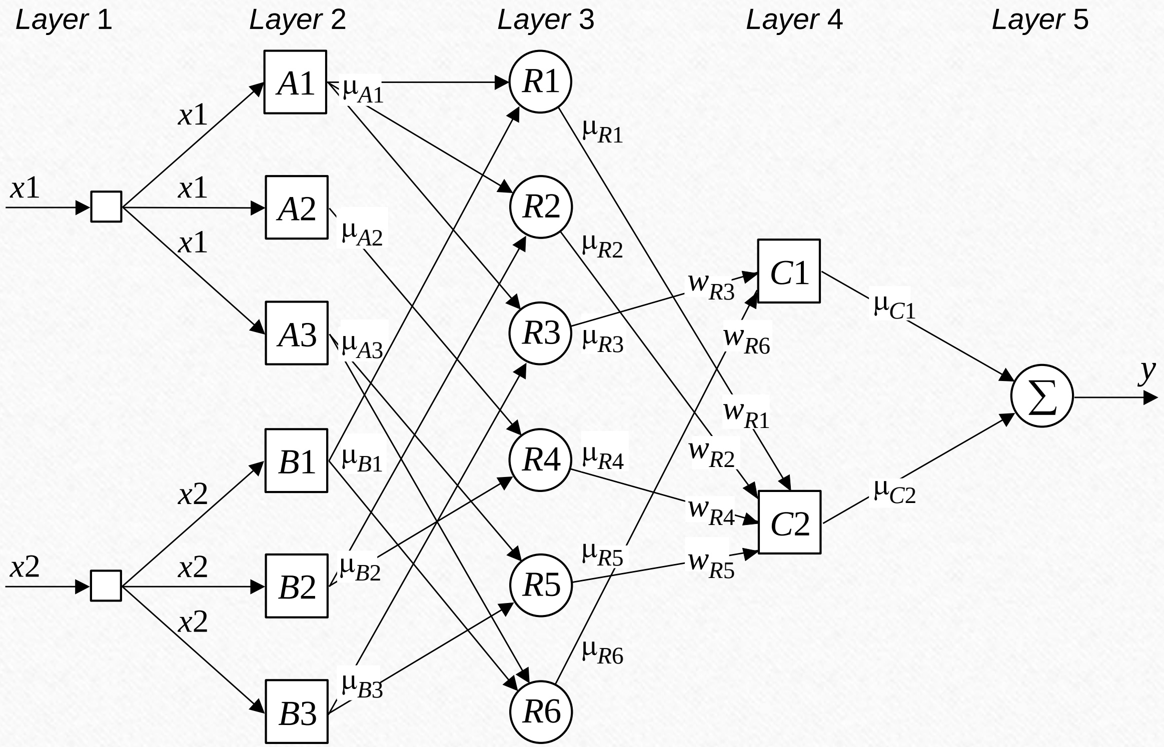
- As a result, neural networks become more transparent, while fuzzy systems become capable of learning.

- A neuro-fuzzy system is a neural network which is functionally equivalent to a fuzzy inference model. It can be trained to develop IF-THEN fuzzy rules and determine membership functions for input and output variables of the system.

- The connectionist structure avoids fuzzy inference, which entails a substantial computational burden.

Neuro-Fuzzy Systems

- The structure of a neuro-fuzzy system is similar to a multi-layer neural network. In general, a neuro-fuzzy system has input and output layers, and three hidden layers that represent membership functions and fuzzy rules.



Neuro-Fuzzy Systems

- The combination of fuzzy logic and neural networks constitutes a powerful means for designing intelligent systems.
- Domain knowledge can be put into a neuro-fuzzy system by human experts in the form of linguistic variables and fuzzy rules.
- When a representative set of examples is available, a neuro-fuzzy system can automatically transform it into a set of fuzzy IF-THEN rules, and reduce our dependence on expert knowledge when building intelligent systems.