An Overview of Computational Intelligence Methods

Computational Intelligence is a relatively new branch of science which, to some extent, can be regarded as a successor of “traditional” Artificial Intelligence. Unlike AI, which relies on symbolic learning and heuristic approaches to problem solving, CI mainly involves systems that are inspired from nature, such as (artificial) neural networks, evolutionary computation, fuzzy systems, chaos theory, probabilistic methods, swarm intelligence, ant systems, and artificial immune systems. In a wider perspective CI includes also part of machine learning, in particular the reinforcement learning methods.

Applying either one or a combination of the above-mentioned disciplines allows implementation of the elements of learning and adaptation in the proposed solutions which make such systems somehow intelligent.

In the game playing domain the most popular CI disciplines are neural networks, evolutionary and neuro-evolutionary methods, and reinforcement learning. These domains are briefly introduced in the remainder of this chapter along with sample game-related applications. The focus of the presentation is on the aspects of learning and autonomous development. Relevant literature is provided for possible further reading.

5.1 Artificial Neural Networks

Artificial neural networks or neural networks (NNs) in short, are computational structures capable of processing information (provided as their input) in order to accomplish a given task. A NN is composed of many simple elements (called neurons) each of which receives input from selected other neurons, performs basic operations on this input information and sends its response out to other neurons in the network.

An inspiration for the above way of processing information is biological nervous system and in particular biological brain. NN models can therefore be regarded as very crude simplification and abstraction of biological networks.
There are several types of NN models. Depending on particular task to be solved an appropriate neural architecture is selected. The basic division of neural models can be made according to the training mode, i.e. supervised vs. unsupervised. A typical example of the supervised network is Multilayer Perceptron (MLP) described below in section 5.1.1 An example of unsupervised network is Self Organizing Map \[171\]. Another possibility is to divide neural architectures based on the information flow direction into feed-forward (e.g. MLP), partly recurrent (e.g. Elman \[89\] or Jordan \[162\] networks), and recurrent (e.g. Hopfield models \[151\]).

Neural networks have been successfully applied to various recognition and classification tasks \[33\], optimization and control problems \[152, 205\], medicine \[202\], financial modeling and prediction \[223\], bioinformatics and chemistry \[192\], games \[13\], and many other fields. Due to variety of possible applications NNs are often regarded as \textit{the second best way of solving any control/classification problem}. In other words, for any given problem at hand a dedicated heuristic approach will most probably be more effective than a NN-based solution, but on the other hand, the generality of NN methodology allows its application to a wide spectrum of problems.

The reader interested in other than games applications of NNs is advised to refer to the above cited books and articles. Fundamentals of neural networks can be studied for example by reading the books by Bishop \[33\], Hassoun \[146\], or Haykin \[147\].

5.1.1 Multilayer Perceptron

Multilayer Perceptron is the most popular type of feed-forward neural networks, which in turn are the most popular neural architectures (in general and in particular in game domain). Neurons in an MLP follow McCulloch-Pitts

![Fig. 5.1. A model of McCulloch-Pitts neuron. See description within the text](image-url)