Chapter 1

Introduction to Computational Intelligence

A major thrust in algorithmic development is the design of algorithmic models to solve increasingly complex problems. Enormous successes have been achieved through the modeling of biological and natural intelligence, resulting in so-called "intelligent systems". These intelligent algorithms include artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, and fuzzy systems. Together with logic, deductive reasoning, expert systems, case-based reasoning and symbolic machine learning systems, these intelligent algorithms form part of the field of *Artificial Intelligence* (AI). Just looking at this wide variety of AI techniques, AI can be seen as a combination of several research disciplines, for example, computer science, physiology, philosophy, sociology and biology.

But what is intelligence? Attempts to find definitions of intelligence still provoke heavy debate. Dictionaries define intelligence as the ability to comprehend, to understand and profit from experience, to interpret intelligence, having the capacity for thought and reason (especially to a high degree). Other keywords that describe aspects of intelligence include creativity, skill, consciousness, emotion and intuition.

Can computers be intelligent? This is a question that to this day causes more debate than the definitions of intelligence. In the mid-1900s, Alan Turing gave much thought to this question. He believed that machines could be created that would mimic the processes of the human brain. Turing strongly believed that there was nothing the brain could do that a well-designed computer could not. More than fifty years later his statements are still visionary. While successes have been achieved in modeling small parts of biological neural systems, there are still no solutions to the complex problem of modeling intuition, consciousness and emotion – which form integral parts of human intelligence.

In 1950 Turing published his test of computer intelligence, referred to as the *Turing test* [858]. The test consisted of a person asking questions via a keyboard to both a person and a computer. If the interrogator could not tell the computer apart from the human, the computer could be perceived as being intelligent. Turing believed that it would be possible for a computer with 10^9 bits of storage space to pass a 5-minute version of the test with 70% probability by the year 2000. Has his belief come true? The answer to this question is left to the reader, in fear of running head first into

Computational Intelligence: An Introduction, Second Edition A.P. Engelbrecht

another debate! However, the contents of this book may help to shed some light on the answer to this question.

A more recent definition of artificial intelligence came from the IEEE Neural Networks Council of 1996: the study of how to make computers do things at which people are doing better. A definition that is flawed, but this is left to the reader to explore in one of the assignments at the end of this chapter.

This book concentrates on a sub-branch of AI, namely Computational Intelligence (CI) – the study of adaptive mechanisms to enable or facilitate intelligent behavior in complex and changing environments. These mechanisms include those AI paradigms that exhibit an ability to learn or adapt to new situations, to generalize, abstract, discover and associate. The following CI paradigms are covered: artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, and fuzzy systems. While individual techniques from these CI paradigms have been applied successfully to solve real-world problems, the current trend is to develop hybrids of paradigms, since no one paradigm is superior to the others in all situations. In doing so, we capitalize on the respective strengths of the components of the hybrid CI system, and eliminate weaknesses of individual components.

The rest of this chapter is organized as follows: Section 1.1 of this chapter presents a short overview of the different CI paradigms, also discussing the biological motivation for each paradigm. A short history of AI is presented in Section 1.2.

At this point it is necessary to state that there are different definitions of what constitutes CI. This book reflects the opinion of the author, and may well cause some debate. For example, swarm intelligence (SI) and artificial immune systems (AIS) are classified as CI paradigms, while many researchers consider these paradigms to belong only under Artificial Life. However, both particle swarm optimization (PSO) and ant colony optimization (ACO), as treated under SI, satisfy the definition of CI given above, and are therefore included in this book as being CI techniques. The same applies to AISs.

1.1 Computational Intelligence Paradigms

This book considers five main paradigms of Computation Intelligence (CI), namely artificial neural networks (NN), evolutionary computation (EC), swarm intelligence (SI), artificial immune systems (AIS), and fuzzy systems (FS). Figure 1.1 gives a summary of the aim of the book. In addition to CI paradigms, probabilistic methods are frequently used together with CI techniques, which is also shown in the figure. Soft computing, a term coined by Lotfi Zadeh, is a different grouping of paradigms, which usually refers to the collective set of CI paradigms and probabilistic methods. The arrows indicate that techniques from different paradigms can be combined to form hybrid systems.

Each of the CI paradigms has its origins in biological systems. NNs model biological

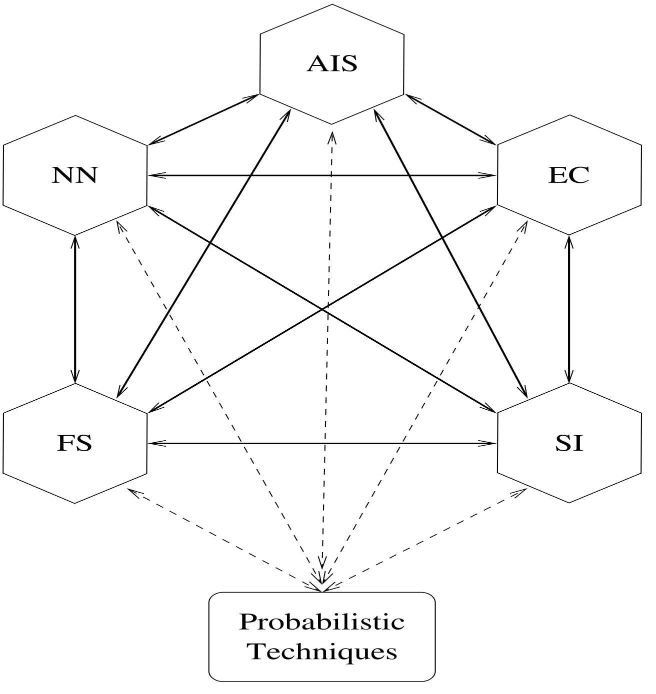

Figure 1.1 Computational Intelligence Paradigms

neural systems, EC models natural evolution (including genetic and behavioral evolution), SI models the social behavior of organisms living in swarms or colonies, AIS models the human immune system, and FS originated from studies of how organisms interact with their environment.

1.1.1 Artificial Neural Networks

The brain is a complex, nonlinear and parallel computer. It has the ability to perform tasks such as pattern recognition, perception and motor control much faster than any computer – even though events occur in the nanosecond range for silicon gates, and milliseconds for neural systems. In addition to these characteristics, others such as the ability to learn, memorize and still generalize, prompted research in algorithmic modeling of biological neural systems – referred to as *artificial neural networks* (NN).

It is estimated that there is in the order of 10-500 billion neurons in the human cortex, with 60 trillion synapses. The neurons are arranged in approximately 1000 main modules, each having about 500 neural networks. *Will it then be possible to truly model the human brain?* Not now. Current successes in neural modeling are for small artificial NNs aimed at solving a specific task. Problems with a single objective can be solved quite easily with moderate-sized NNs as constrained by the capabilities of modern computing power and storage space. The brain has, however, the ability to solve several problems simultaneously using distributed parts of the brain. We still have a long way to go ...

The basic building blocks of biological neural systems are nerve cells, referred to as neurons. As illustrated in Figure 1.2, a neuron consists of a cell body, dendrites and an axon. Neurons are massively interconnected, where an interconnection is between the axon of one neuron and a dendrite of another neuron. This connection is referred to as a *synapse*. Signals propagate from the dendrites, through the cell body to the axon; from where the signals are propagated to all connected dendrites. A signal is transmitted to the axon of a neuron only when the cell "fires". A neuron can either inhibit or excite a signal.

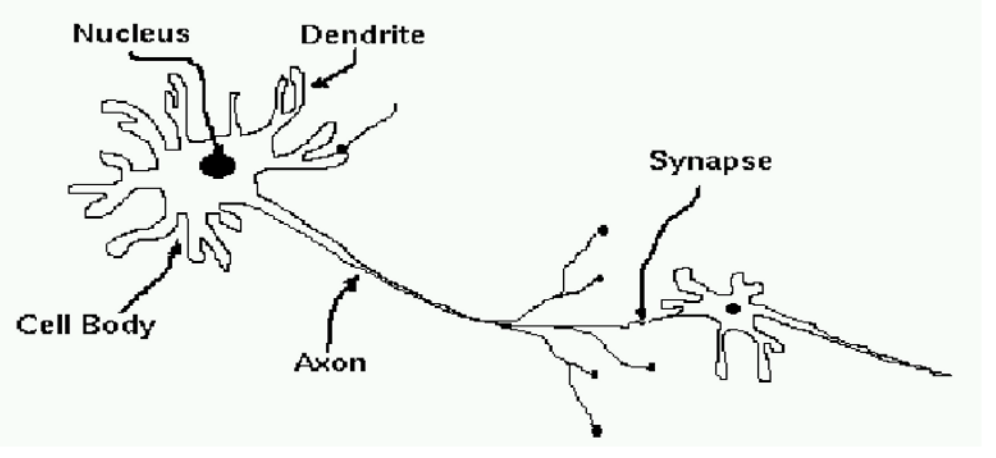

Figure 1.2 A Biological Neuron

An artificial neuron (AN) is a model of a biological neuron (BN). Each AN receives signals from the environment, or other ANs, gathers these signals, and when fired, transmits a signal to all connected ANs. Figure 1.3 is a representation of an artificial neuron. Input signals are inhibited or excited through negative and positive numerical weights associated with each connection to the AN. The firing of an AN and the strength of the exiting signal are controlled via a function, referred to as the activation function. The AN collects all incoming signals, and computes a net input signal as a function of the respective weights. The net input signal serves as input to the activation function which calculates the output signal of the AN.

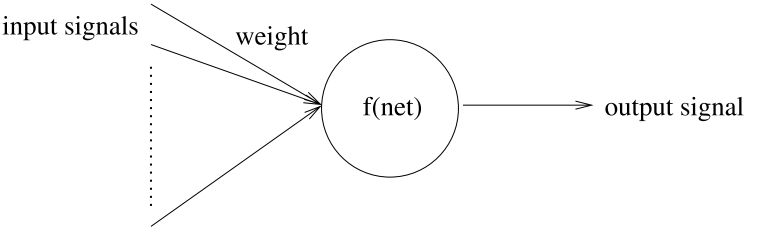

Figure 1.3 An Artificial Neuron

An artificial neural network (NN) is a layered network of ANs. An NN may consist of an input layer, hidden layers and an output layer. ANs in one layer are connected, fully or partially, to the ANs in the next layer. Feedback connections to previous layers are also possible. A typical NN structure is depicted in Figure 1.4.

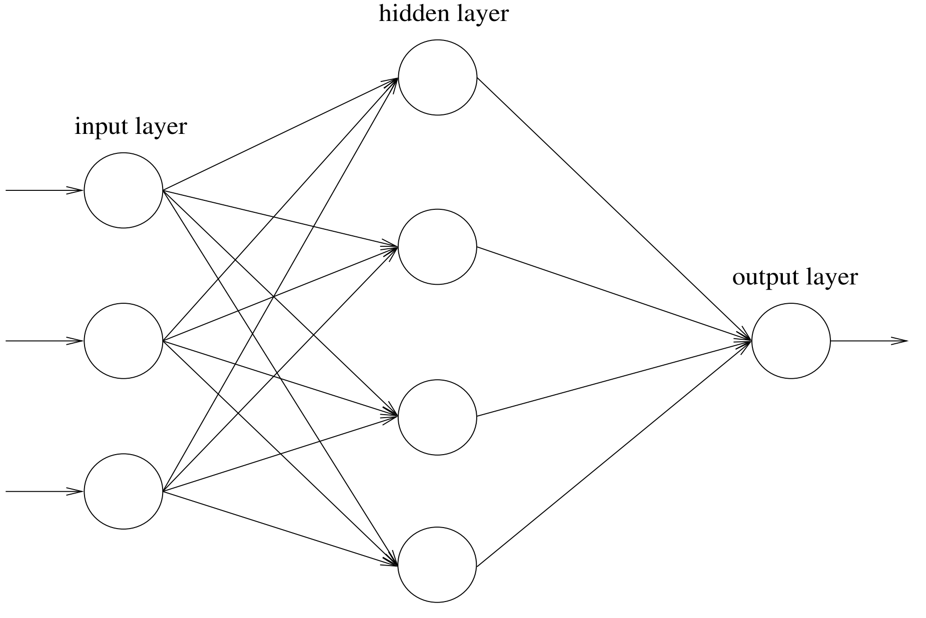

Figure 1.4 An Artificial Neural Network

Several different NN types have been developed, for example (the reader should note that the list below is by no means complete):

- single-layer NNs, such as the Hopfield network;
- multilayer feedforward NNs, including, for example, standard backpropagation, functional link and product unit networks;
- temporal NNs, such as the Elman and Jordan simple recurrent networks as well as time-delay neural networks;
- self-organizing NNs, such as the Kohonen self-organizing feature maps and the learning vector quantizer;
- combined supervised and unsupervised NNs, e.g. some radial basis function networks.

These NN types have been used for a wide range of applications, including diagnosis of diseases, speech recognition, data mining, composing music, image processing, forecasting, robot control, credit approval, classification, pattern recognition, planning game strategies, compression, and many others.

1.1.2 Evolutionary Computation

Evolutionary computation (EC) has as its objective to mimic processes from natural evolution, where the main concept is survival of the fittest: the weak must die. In natural evolution, survival is achieved through reproduction. Offspring, reproduced from two parents (sometimes more than two), contain genetic material of both (or all) parents – hopefully the best characteristics of each parent. Those individuals that inherit bad characteristics are weak and lose the battle to survive. This is nicely illustrated in some bird species where one hatchling manages to get more food, gets stronger, and at the end kicks out all its siblings from the nest to die.

Evolutionary algorithms use a *population* of individuals, where an individual is referred to as a *chromosome*. A chromosome defines the characteristics of individuals in the population. Each characteristic is referred to as a *gene*. The value of a gene is referred to as an *allele*. For each generation, individuals compete to reproduce offspring. Those individuals with the best survival capabilities have the best chance to reproduce. Offspring are generated by combining parts of the parents, a process referred to as *crossover*. Each individual in the population can also undergo mutation which alters some of the allele of the chromosome. The survival strength of an individual is measured using a *fitness* function which reflects the objectives and constraints of the problem to be solved. After each generation, individuals may undergo *culling*, or individuals may survive to the next generation (referred to as *elitism*). Additionally, behavioral characteristics (as encapsulated in phenotypes) can be used to influence the evolutionary process in two ways: phenotypes may influence genetic changes, and/or behavioral characteristics evolve separately.

Different classes of evolutionary algorithms (EA) have been developed:

- **Genetic algorithms** which model genetic evolution.
- **Genetic programming** which is based on genetic algorithms, but individuals are programs (represented as trees).
- **Evolutionary programming** which is derived from the simulation of adaptive behavior in evolution (*phenotypic* evolution).
- **Evolution strategies** which are geared toward modeling the strategy parameters that control variation in evolution, i.e. the evolution of evolution.
- **Differential evolution**, which is similar to genetic algorithms, differing in the reproduction mechanism used.
- **Cultural evolution** which models the evolution of culture of a population and how the culture influences the genetic and phenotypic evolution of individuals.
- **Coevolution** where initially “dumb” individuals evolve through cooperation, or in competition with one another, acquiring the necessary characteristics to survive.

Other aspects of natural evolution have also been modeled. For example, mass extinction, and distributed (island) genetic algorithms, where different populations are maintained with genetic evolution taking place in each population. In addition, aspects such as migration among populations are modeled. The modeling of parasitic behavior has also contributed to improved evolutionary techniques. In this case parasites infect individuals. Those individuals that are too weak die. On the other hand, immunology has been used to study the evolution of viruses and how antibodies should evolve to kill virus infections.

Evolutionary computation has been used successfully in real-world applications, for example, data mining, combinatorial optimization, fault diagnosis, classification, clustering, scheduling, and time series approximation.

1.1.3 Swarm Intelligence

Swarm intelligence (SI) originated from the study of colonies, or swarms of social organisms. Studies of the social behavior of organisms (individuals) in swarms prompted the design of very efficient optimization and clustering algorithms. For example, simulation studies of the graceful, but unpredictable, choreography of bird flocks led to the design of the particle swarm optimization algorithm, and studies of the foraging behavior of ants resulted in ant colony optimization algorithms.

Particle swarm optimization (PSO) is a stochastic optimization approach, modeled on the social behavior of bird flocks. PSO is a population-based search procedure where the individuals, referred to as particles, are grouped into a swarm. Each particle in the swarm represents a candidate solution to the optimization problem. In a PSO system, each particle is "flown" through the multidimensional search space, adjusting its position in search space according to its own experience and that of neighboring particles. A particle therefore makes use of the best position encountered by itself and the best position of its neighbors to position itself toward an optimum solution. The effect is that particles "fly" toward an optimum, while still searching a wide area around the current best solution. The performance of each particle (i.e. the "closeness" of a particle to the global minimum) is measured according to a predefined fitness function which is related to the problem being solved. Applications of PSO include function approximation, clustering, optimization of mechanical structures, and solving systems of equations.

Studies of ant colonies have contributed in abundance to the set of intelligent algorithms. The modeling of pheromone depositing by ants in their search for the shortest paths to food sources resulted in the development of shortest path optimization algorithms. Other applications of ant colony optimization include routing optimization in telecommunications networks, graph coloring, scheduling and solving the quadratic assignment problem. Studies of the nest building of ants and bees resulted in the development of clustering and structural optimization algorithms.

1.1.4 Artificial Immune Systems

The natural immune system (NIS) has an amazing pattern matching ability, used to distinguish between foreign cells entering the body (referred to as *non-self*, or *antigen*) and the cells belonging to the body (referred to as *self*). As the NIS encounters antigen, the adaptive nature of the NIS is exhibited, with the NIS memorizing the structure of these antigen for faster future response the antigen.

In NIS research, four models of the NIS can be found:

- The **classical view** of the immune system is that the immune system distinguishes between self and non-self, using lymphocytes produced in the lymphoid organs. These lymphocytes "learn" to bind to antigen.
- **Clonal selection theory**, where an active B-Cell produces antibodies through a cloning process. The produced clones are also mutated.
- **Danger theory**, where the immune system has the ability to distinguish between dangerous and non-dangerous antigen.
- **Network theory**, where it is assumed that B-Cells form a network. When a B-Cell responds to an antigen, that B-Cell becomes activated and stimulates all other B-Cells to which it is connected in the network.

An artificial immune system (AIS) models some of the aspects of a NIS, and is mainly applied to solve pattern recognition problems, to perform classification tasks, and to cluster data. One of the main application areas of AISs is in anomaly detection, such as fraud detection, and computer virus detection.

1.1.5 Fuzzy Systems

Traditional set theory requires elements to be either part of a set or not. Similarly, binary-valued logic requires the values of parameters to be either 0 or 1, with similar constraints on the outcome of an inferencing process. Human reasoning is, however, almost always not this exact. Our observations and reasoning usually include a measure of uncertainty. For example, humans are capable of understanding the sentence: "Some Computer Science students can program in most languages". But how can a computer represent and reason with this fact?

Fuzzy sets and fuzzy logic allow what is referred to as *approximate reasoning*. With fuzzy sets, an element belongs to a set to a certain degree of certainty. Fuzzy logic allows reasoning with these uncertain facts to infer new facts, with a degree of certainty associated with each fact. In a sense, fuzzy sets and logic allow the modeling of common sense.

The uncertainty in fuzzy systems is referred to as *nonstatistical uncertainty*, and should not be confused with *statistical uncertainty*. Statistical uncertainty is based on the laws of probability, whereas nonstatistical uncertainty is based on vagueness, imprecision and/or ambiguity. Statistical uncertainty is resolved through observations. For example, when a coin is tossed we are certain what the outcome is, while before tossing the coin, we know that the probability of each outcome is 50%. Nonstatistical uncertainty, or fuzziness, is an inherent property of a system and cannot be altered or resolved by observations.

Fuzzy systems have been applied successfully to control systems, gear transmission and braking systems in vehicles, controlling lifts, home appliances, controlling traffic signals, and many others.

1.2 Short History

Aristotle (384–322 bc) was possibly the first to move toward the concept of artificial intelligence. His aim was to explain and codify styles of deductive reasoning, which he referred to as *syllogisms*. Ramon Llull (1235–1316) developed the *Ars Magna*: an optimistic attempt to build a machine, consisting of a set of wheels, which was supposed to be able to answer all questions. Today this is still just a dream – or rather, an illusion. The mathematician Gottfried Leibniz (1646–1716) reasoned about the existence of a *calculus philosophicus*, a universal algebra that can be used to represent all knowledge (including moral truths) in a deductive system.

The first major contribution was by George Boole in 1854, with his development of the foundations of propositional logic. In 1879, Gottlieb Frege developed the foundations of predicate calculus. Both propositional and predicate calculus formed part of the first AI tools.

It was only in the 1950s that the first definition of artificial intelligence was established by Alan Turing. Turing studied how machinery could be used to mimic processes of the human brain. His studies resulted in one of the first publications of AI, entitled *Intelligent Machinery*. In addition to his interest in intelligent machines, he had an interest in how and why organisms developed particular shapes. In 1952 he published a paper, entitled *The Chemical Basis of Morphogenesis* – possibly the first studies in what is now known as *artificial life*.

The term *artificial intelligence* was first coined in 1956 at the Dartmouth conference, organized by John MacCarthy – now regarded as the father of AI. From 1956 to 1969 much research was done in modeling biological neurons. Most notable was the work on perceptrons by Rosenblatt, and the *adaline* by Widrow and Hoff. In 1969, Minsky and Papert caused a major setback to artificial neural network research. With their book, called *Perceptrons*, they concluded that, in their “intuitive judgment”, the extension of simple perceptrons to multilayer perceptrons “is sterile”. This caused research in NNs to go into hibernation until the mid-1980s. During this period of hibernation a few researchers, most notably Grossberg, Carpenter, Amari, Kohonen and Fukushima, continued their research efforts.

The resurrection of NN research came with landmark publications from Hopfield, Hinton, and Rumelhart and McLelland in the early and mid-1980s. From the late 1980s research in NNs started to explode, and is today one of the largest research areas in Computer Science.

The development of evolutionary computation (EC) started with genetic algorithms in the 1950s with the work of Fraser, Bremermann and Reed. However, it is John Holland who is generally viewed as the father of EC, most specifically of genetic algorithms. In these works, elements of Darwin’s theory of evolution [173] were modeled algorithmically. In the 1960s, Rechenberg developed evolutionary strategies (ES). Independently from this work, Lawrence Fogel developed evolutionary programming as an approach to evolve behavioral models. Other important contributions that shaped the field were by De Jong, Schaffer, Goldberg, Koza, Schwefel, Storn, and Price.

Many people believe that the history of fuzzy logic started with Gautama Buddha (563 bc) and Buddhism, which often described things in shades of gray. However, the Western community considers the work of Aristotle on two-valued logic as the birth of fuzzy logic. In 1920 Lukasiewicz published the first deviation from two-valued logic in his work on three-valued logic – later expanded to an arbitrary number of values. The quantum philosopher Max Black was the first to introduce quasi-fuzzy sets, wherein degrees of membership to sets were assigned to elements. It was Lotfi Zadeh who contributed most to the field of fuzzy logic, being the developer of fuzzy sets [944]. From then, until the 1980s fuzzy systems was an active field, producing names such as Mamdani, Sugeno, Takagi and Bezdek. Then, fuzzy systems also experienced a dark age in the 1980s, but was revived by Japanese researchers in the late 1980s. Today it is a very active field with many successful applications, especially in control systems. In 1991, Pawlak introduced rough set theory, where the fundamental concept is that of finding a lower and upper approximation to input space. All elements within the lower approximation have full membership, while the boundary elements (those elements between the upper and lower approximation) belong to the set to a certain degree.

Interestingly enough, it was an unacknowledged South African poet, Eugene N Marais (1871-1936), who produced some of the first and most significant contributions to swarm intelligence in his studies of the social behavior of both apes and ants. Two books on his findings were published more than 30 years after his death, namely *The Soul of the White Ant* [560] and *The Soul of the Ape* [559]. The algorithmic modeling of swarms only gained momentum in the early 1990s with the work of Marco Dorigo on the modeling of ant colonies. In 1995, Eberhart and Kennedy [224, 449] developed the particle swarm optimization algorithm as a model of bird flocks. Swarm intelligence is in its infancy, and is a promising field resulting in interesting applications.

The different theories in the science of immunology inspired different artificial immune models (AISs), which are either based on a specific theory on immunology or a combination of the different theories. The initial *classical view* and theory of *clonal selection* in the natural immune system was defined by Burnet [96] as B-Cells and Killer-T-Cells with antigen-specific receptors. This view was enhanced by the definition of Bretscher and Cohn [87] by introducing the concept of a helper T-Cell. Lafferty and Cunningham [497] added a co-stimulatory signal to the helper T-Cell model of Bretscher and Cohn [87].

The first work in AIS on the modeling of the discrimination between *self* and *non-self* with mature T-Cells was introduced by Forrest *et al.* [281]. Forrest *et al.* introduced a training technique known as the *negative selection* of T-Cells [281]. The model of Mori *et al* [606] was the first to implement the *clonal selection* theory, which was applied to optimization problems. The *network theory* of the natural immune system was introduced and formulated by Jerne [416] and further developed by Perelson [677]. The theory of Jerne is that the B-Cells are interconnected to form a network of cells

[416, 677]. The first mathematical model on the theory of Jerne was proposed by Farmer *et al.* [255]. The *network theory* has been modeled into artificial immune systems (AISs) for data mining and data analysis tasks. The earliest AIS research based on the mathematical model of the *network theory* [255], was published by Hunt and Cooke [398]. The model of Hunt and Cooke was applied to the recognition of DNA sequences. The *danger theory* was introduced by Matzinger [567, 568] and is based on the co-stimulated model of Lafferty and Cunningham [497]. The main idea of the *danger theory* is that the immune system distinguishes between what is dangerous and non-dangerous in the body. The first work on danger theory inspired AISs was published by Aickelin and Cayzer [14].

1.3 Assignments

1. Comment on the eligibility of Turing's test for computer intelligence, and his belief that computers with 10^9 bits of storage would pass a 5-minute version of his test with 70% probability.
2. Comment on the eligibility of the definition of artificial intelligence as given by the 1996 IEEE Neural Networks Council.
3. Based on the definition of CI given in this chapter, show that each of the paradigms (NN, EC, SI, AIS, and FS) does satisfy the definition.