# Artificial Immune Systems

Mark Read, Paul Andrews and Jon Timmis

**Abstract** The field of artificial immune systems (AIS) can be thought of comprising two threads of research: the employment of mathematical and computational techniques in the modelling of immunology, and the incorporation of immune system inspired mechanisms and metaphors in the development of engineering solutions. The former permits the integration of immunological data and sub-models into a coherent whole, which may be of value to immunologists in the facilitation of immunological understanding, hypothesis testing, and the direction of future research. The latter attempts harness the perceived properties of the immune system in the solving of engineering problems. This chapter concentrates on the latter: the development and application of immune inspiration to engineering solutions.

## 1 Introduction

Artificial Immune Systems (AIS) is a branch of biologically inspired computation focusing on many aspects of immune systems. AIS development can be seen as having two target domains: the provision of solutions to engineering problems through the adoption of immune system inspired concepts; and the provision of models and simulations with which to study immune system theories.

Mark Read
Department of Computer Science, University of York, Heslington, York. YO10 5DD. e-mail: mark.read@cs.york.ac.uk

Paul Andrews
Department of Computer Science, University of York, Heslington, York. YO10 5DD. e-mail: psa@cs.york.ac.uk

Jon Timmis
Department of Computer Science and Department of Electronics, University of York, Heslington, York. YO10 5DD. e-mail: jtimmis@cs.york.ac.uk

The motivation for building immune inspired solutions to engineering problems arises from the identification of properties within the immune system that are attractive from an engineering perspective. These include [de Castro & Timmis 2002a]: the self-organisation of huge numbers of immune cells; the distributed operation of the immune system throughout the body; pattern recognition and anomaly detection to enable the immune system to recognise pathogens; and optimisation and memory to improve and remember immune responses. AIS take inspiration from these properties and associated immune processes, and have been defined as:

> "adaptive systems, inspired by theoretical immunology and observed immune function, principles and models, which are applied to problem solving."
>
> [de Castro & Timmis 2002a]

The field of AIS also encompasses modelling and simulation techniques to understand the immune system in general (see [Timmis *et al.* 2008a] for a review), however, our focus in this article is on immune inspired systems for engineering problems.

This chapter is not intended as an extensive review chapter, but its purpose is to provide a general introduction to the area and provide discussion on the major research issues relating to the field of AIS. Therefore, in this chapter, we briefly explore the underlying immunology that has served as inspiration for the development of immune-inspired algorithms. We have chosen not to focus on the modelling aspect of AIS, but rather focus on the main algorithms that have been developed over the recent years, this is undertaken in section 3. In this section we discuss four main immune-inspired algorithms that dominate the literature namely clonal selection, immune network, negative selection and dendritic cell algorithms and highlight their usage in terms of applications. Section 4 then follows with a discussion AIS and how researchers have begun to evaluate current AIS and details new frameworks and methodologies that are being developed that aim to help develop AIS in a more principled manner. We also briefly discuss the application of AIS to a variety of different domains and the types of applications that AIS might be better suited too, and we finally provide a very brief outline of the modelling approaches that can be found in the literature that are employed to help further the understanding of immunology. Section 5 provides a chapter summary.

## 2 The Immune System

Immunology concerns the study of the immune system and the effects of its operation on the body. The immune system is normally defined in relation to its perceived function: a defence system that has evolved to protect its host from pathogens (harmful micro-organisms such as bacteria, viruses and parasites) [Goldsby *et al.* 2003]. It comprises a variety of specialised cells that circulate and monitor the body, various extra-cellular molecules, and immune organs that provide an environment for immune cells to interact, mature and respond. The collective action of immune cells

and molecules forms a complex network leading to the detection and recognition of pathogens within the body. This is followed by a specific effector response aimed at eliminating the pathogen. This recognition and response process is vastly complicated with many of the details not yet properly understood.

In mammals, the immune system can be classified into two components based on functionality: a less specific component termed innate immunity and a more specific component termed adaptive (or acquired) immunity. The mechanisms of innate immunity are generic defence mechanisms that are non-specific to particular examples of pathogen, but act against general classes of pathogen. They are encoded within the genes of the species, and do not adapt during the lifetime of the individual. Examples include the inflammatory response, phagocytic immune cells (those that can ingest and kill pathogens), anatomic barriers such as skin, and physiologic barriers such as temperature.

By contrast, the mechanisms of adaptive immunity enable the immune system to adapt to previously unseen pathogens based upon exposure to them [Goldsby *et al.* 2003]. This is achieved through a learning mechanism that operates during the lifetime of the individual. Additionally, once exposed to a pathogen, memory mechanisms exist to allow the immune system to remember the pathogen. This enables a faster and more effective secondary response that can be elicited against the pathogen if it is encountered again. The adaptive and innate arms of the immune system interact to provide the body with a comprehensive defence mechanism against pathogens.

All immune cells, and the majority of other cells of the body, possess protein molecules on their surface that act as receptors to other extra-cellular molecules. When a sufficiently strong chemical bond occurs between a receptor and another molecule (a ligand), a cascade of intra-cellular signals is initiated, the outcome of which depends on the initiating receptors. This process provides the immune system with a mechanism for recognition at the molecular level. Two types of immune cell receptor exist: innate receptors that have evolved to recognise specific molecules; and the unique receptors of lymphocytes that are generated during the life time of the individual to recognise previously unseen molecules. The later of these molecules are generically known as *antigens*, a term given to any molecular structure that can chemically bind to the unique receptors of immune cells known as T and B-cells. The antigen receptors of the B-cell are called *antibodies*, and those of the T-cell are called T cell receptors (TCR). They are both generated via a stochastic process, and are vital to the body's adaptive immune response. Communication between immune cells involves a number of immune molecules. They include cytokines, immune cell receptors, antibodies, enzymes, plasma proteins and adhesion molecules. The cytokines, for example, are signalling molecules secreted by both immune and other bodily cells, which are then detected via specific cellular receptors. Many different types of cytokine exist and their effects include the activation, differentiation, growth, movement and death of many types of cell [Cohen 2000].

### *2.1 Motivation for Immune Inspired Engineering Solutions*

Why is it that engineers are attracted to the immune system for inspiration? The immune system exhibits several properties that engineers recognise as being desirable in their systems. [Timmis & Andrews 2007, Timmis *et al.* 2008a, de Castro & Timmis 2002a] have identified these as:

Distribution and self-organisation. The behaviour of the immune system is deployed through the actions of billions of agents (cells and molecules) distributed throughout the body. Their collective effects can be highly complex with no central controller. An organised response emerges as a system wide property derived from the low level agent behaviours. These immune agents act concurrently making immune processes naturally parallised.

Learning, adaption, and memory. The immune system is capable of recognising previously unseen pathogens, thus exibits the ability to learn. Learning implies the presence of memory, which is present in the immune system enabling it to 'remember' previously encounted pathogens. This is encapsuatled by the phenomenon of primary and secondary responses: the first time a pathogen is encountered an immune response (the primary response) is elicited. The next time that pathogen is encounted a faster and often more aggressive response is mounted (the secondary response).

Pattern recognition. Through its various receptors and molecules the immune system is capable of recognising a diverse range of patterns. This is accomplished through receptors that perceive antigenic materials in differing contexts (processed molecules, whole molecules, additional signals etc). Receptors of the innate immune system vary little, whilst receptors of the adaptive immune system, such as as antibodies and T-cell receptors are subject to huge diversity.

Classification. The immune system is very effective at distinguishing harmful substances (non-self) from the body's own tissues (self), and directing its actions accordingly. From a computational perspective, it does this with access to only a single class of data, self molecules [Stibor *et al.* 2005]. Creation of a system that effectively classifies data into two classes, having been trained on examples from only one, is a challenging task.

## 3 Engineering Artificial Immune Systems

[de Castro & Timmis 2002a] have proposed a flexible and generic layered approach to the development of immune inspired engineering solutions, shown in Figure 1. This framework identifies the main design decisions that need to be addressed in the deployment of an immune inspired engineering solution: representations, affinity measures and immune algorithms.

Given a particular application domain, an appropriate representation of the data must be chosen. In AIS, this typically follow the notion of 'shape-space' [Perelson

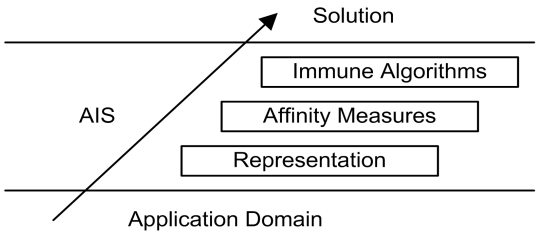


**Fig. 1** The layered framework approach to constructing AIS solutions, taken from [de Castro & Timmis 2002a]

& Oster 1979]. Here, molecules $m$ (such as receptors and antigen) exist as points in a shape space $S$, and can be represented as strings of attributes $m = < m_1, m_2, ..., m_L >$ in an $L$ dimensional space, $m \in S^L$. The attributes $m_i$ will represent aspects of a problem domain: patterns to be recognised, functional values to be optimised, combinations of input and proposed actions, etc. [de Castro & Timmis 2002a] suggest four data types of which these attributes may belong: real valued; integer valued; hamming valued, finite length strings composed of digits a finite alphabet; and categorically valued, where values include items such as 'name', or 'colour'. The affinity measures are functions or criteria through which interactions of the AIS elements are quantified. They are highly dependent on the representation chosen, for example, continuous variables typically employ the Euclidean distance measure, whereas bit string representations may use the hamming distance. Work in [McEwan *et al.* 2008] provides a very convincing critique of the shape-space paradigm and discusses the limitations of such an approach. We discuss their paper in more detail in section 4.1

The highest layer of the framework details the selection of an immune inspired algorithm to operate over the immune elements of the system. Various types of immune inspired algorithm exist, which can operate independently of the choice of representation and affinity measure, adding dynamics to the algorithm populations based on measurements the affinity functions provide. Despite this, immune algorithms should be chosen with care based on the problem's data [Freitas & Timmis 2007].

In what is considered to be one of first papers in AIS, [Farmer *et al.* 1986] examined the immune system in the context of classifier systems, essentially highlighting the parallels of the immune network theory [Jerne 1974] and artificial intelligence. AIS has since been applied to a large range of domains that can broadly be classified as learning, anomaly detection, and optimisation problems [Hart & Timmis 2008]. Four main classes of AIS algorithm have been applied to these problems and we outline each below.

**Algorithm 1** A generic clonal selection algorithm, based on CLONALG [de Castro & Von Zuben 2000, de Castro & Von Zuben 2002].

**input:** $S$ = a set of antigens, representing data elements to be recognised.

**output:** $M$ = set of memory B-cells capable of classifying unseen data elements.

**begin**
  Generate set of random specificity B-cells $B$.
  **for all** antigens $ag \in S$ **do**
    Calculate affinity of all B-cells $b \in B$ with $ag$.
    Select highest affinity B-cells, perform affinity proportional cloning, place clones in $C$.
    **for all** B- cell clones $c \in C$ **do**
      Mutate $c$ at rate inversely proportional to affinity.
      Determine affinity of $c$ with $ag$.
    **end for**
    Copy all $c \in C$ into $B$.
    Copy the highest affinity clones $c \in C$ into memory set $M$.
    Replace lowest affinity B-cells $b \in B$ with randomly generated alternatives.
  **end for**
**end**

## *3.1 Clonal Selection Theory Inspired AIS*

Clonal selection based algorithms attempt to capture the antigen driven proliferation of B-cells that results in their improved binding abilities. Using a process known as affinity maturation, the receptors of B-cell are mutated and subsequent B-cell selection results in a population of B-cell with better overall affinity for the antigen. Clonal selection algorithms capture the properties of learning, memory, adaption, and pattern recognition. [Timmis *et al.* 2008a].

A generic clonal selection inspired algorithm, based on CLONALG [de Castro & Von Zuben 2002, de Castro & Von Zuben 2000], is presented in Algorithm 1. A set of patterns (antigens) is input to the algorithm, and output is a set of memory B-cells capable of recognising unseen patterns. A randomly initialised set of B-cells are preferentially selected based on their affinity for the antigen. The higher affinity cells are cloned proportionally to their affinity, and mutated at a rate inversely proportional to affinity. The higher affinity clones will replace the lower affinity cells of the previous generation. Very high affinity clones compete for a place in the set of memory cells. This algorithm can be tailored towards optimisation problems by removing the antigen set $S$, and directly representing the function or domain to be optimised as the affinity function. As clonal selection algorithms employ mutation and selection of a population of candidate solutions, they tend to be similar to other evolutionary algorithms [Newborough & Stepney 2005].

In Algorithm 1 we have outlined a *generic* clonal selection algorithm, however, there are many variants in the literature that have been augmented and altered to fit specific application areas. For example, work in [Watkins *et al.* 2004] developed a reinforcement learning approach known as AIRS (Artificial Immune Recognition

System), based on the ideas of clonal selection for the classification of unseen data items. In effect AIRS is an instance creation algorithm which acts as a preprocessor to the *k*-nearest neighbor approach that has been found to perform well on certain types of classification problems [Secker & Freitas 2007]. In the context of dynamic learning, work by [Kim & Bentley 2002a, Kim & Bentley 2002c, Kim & Bentley 2002b] developed a network intrusion detection system based on a dynamic variant of the clonal selection paradigm that was capable of identifying potential attacks to computer networks in an on-line manner and then be able to, in a limited manner, adapt to new types of attacks. As a final example, work by [Cutello *et al.* 2004, Nicosia *et al.* 2004a, Cutello *et al.* 2005] have developed particularly effective optimisation algorithms based on variants of clonal selection by making use of novel selection and mutation mechanisms tailored specifically for certain types of optimisation problems.

## *3.2 Immune Network Theory AIS*

The Immune network theory as proposed by [Jerne 1974] views the immune system as a regulated network of molecules and cells that recognise each other producing a self-organising behaviour, and memory even in the absence of antigen. B-cells interact via receptors to stimulate and suppress each other to form a regulatory network that forms an *internal image* of the antigenic patterns that the immune system observes [Farmer *et al.* 1986].

As with clonal selection, the immune network theory has provided inspiration for many algorithms ranging from optimisation to machine learning [de Castro & Timmis 2002b, Honorio *et al.* 2007, Timmis & Neal 2000, de Castro & Von Zuben 2001, Bezerra *et al.* 2004]. From a machine learning perspective, many of the systems are unsupervised and produce an instance reduction of the data space. They present clusters of this reduced data as networks of connected B-cells, where a B-cell may be considered a point $m$ in the shape space $S^L$ discussed above. The motivation for such algorithms is that the resulting networks highlight structures inherent in the data set, can reduce the dimensionality and complexity of the data [Neal 2003]. A generic immune network algorithm, based on aiNet [de Castro & Von Zuben 2001], is presented in Algorithm 2. It is a modified version of CLONALG that incorporates a mechanism of suppression amongst B-cells.

In aiNet, data items are represented as antigen which B-cells detectors recognise. Like clonal section algorithms (Algorithm 1), affinity maturation produces B-cells with differing specificities, and competition removes the worst of these cells from the population. A suppressive mechanism then prunes cells of similar specificities from the population. The resulting network of B-cells is then representative of clusters within the data.

Despite possessing suppressive mechanisms, early immune network algorithms suffered from an excess of B-cells, which hindered run time efficiency and rendered the resulting networks overly complex [Timmis & Neal 2000]. To address this, work

**Algorithm 2** A generic immune network algorithm, based on aiNet [de Castro & Von Zuben 2001], taken from [Timmis *et al.* 2008a].

**input:** $S$ = a set of antigens, representing data elements to be clustered,
$nt$ network affinity threshold, $ct$ clonal pool threshold, $h$ number of highest affinity clones,
$a$ number of new antibodies to introduce.
**output:** $N$ = set of memory detectors capable of classifying unseen patterns.

**begin**
Generate set of random specificity B-cells $N$.
**repeat**
**for all** antigens $ag \in S$ **do**
Calculate affinity of all B-cells $b \in N$ with $ag$.
Select highest affinity B-cells, perform affinity proportional cloning, place clones in $C$.
**for all** B- cell clones $c \in C$ **do**
Mutate $c$ at rate inversely proportional to affinity.
Determine affinity of $c$ with $ag$.
**end for**
Select $h$ highest affinity clones $c \in C$ and place in $D$.
Remove all elements of $D$ whose affinity with $ag$ is less than $ct$.
Remove elements of $D$ whose affinity with other elements in $D$ is less than $ct$.
Insert remaining elements of $D$ into $N$.
**end for**
Determine affinity between each pair of B-cells in $N$.
Systemically remove all B cells whose affinity to another B cell is less than $nt$.
Introduce $a$ new, randomly generated, B-cells into $N$.
**until** a stopping condition has been satisfied
**end**

by [Timmis & Neal 2000] incorporated the notion of an artificial recognition ball (ARB), a bounded area surrounding a point in antigenic space. All B-cells exhibiting specificities within an ARB's area are represented by that ARB, thus removing the requirement to explicitly represent each of them. To further regulate the network's population size, ARBs lie in competition with one another for a share of finite system wide resource; ARBs that are unable to claim sufficient resource are removed from the network. Resource is allocated on the basis of ARB stimulation, derived from antigen affinity, and from low affinity to the other ABRs with which they are linked. Hence, the pressures of the algorithm are to derive clusters of linked but well spread out ARBs that represent structure in the data.

A similar, but modified, immune network algorithm was published by [Neal 2003]. Both cloning and hypermutation are absent in this algorithm; new ARBs are created from antigen that fall outside the range of existing ARBs in the network. The algorithm does not incorporate any stopping criteria, and can be used to create cluster based representations of dynamically changing data. This algorithm removed the requirement for central control over the allocation of resources; ARBs are responsible for determining their own stimulation and acting accordingly. The nature of the stimulation calculation prevents ARB population explosion and renders the algorithm robust regarding exact parameter values. The algorithm captures

**Algorithm 3** Generic negative selection algorithm, based on [Forrest *et al.* 1994]

```
input: S = set of self strings characterising benign, normal data.
output: A = Stream of nonself strings detected.

begin
    Create empty set of detector strings D          ▷ Generation of detector strings
    Generate random strings R.
    for all random strings r ∈ R do
        for all self strings s ∈ S do
            if r matches s then
                Discard r
            else
                Place r in D
            end if
        end for
    end for

    while There exist protected strings p to check do          ▷ Detection stage
        Retrieve protected string p
        for all detector strings d ∈ D do
            if p matches d then
                Place p in A and output.          ▷ Nonself string detected
            end if
        end for
    end while
end
```

well the properties of self-organisation and population regulation as exhibited by the immune system. [Galeano *et al.* 2005] provides a good review of many other immune networks that appear in the literature.

## *3.3 Negative Selection AIS*

Inspired by the observation that the immune system protects the host body from invading pathogens, early AIS mapped these qualities to the invasion of computers and computer networks by viruses, worms, and intruders. The concept of self non-self discrimination provided the basis for the development of security AIS algorithm. Specifically, a process called negative selection that aims to derive a set of detectors capable of recognising only nonself. An example algorithm, based on [Forrest *et al.* 1994], is shown in Algorithm 3.

This algorithm was applied to protecting a computer from unauthorised changes, such as infection with a virus. There are two main stages to the algorithm: the generation of detectors; and the online monitoring of data and programs for changes. In the detector generation stage, the collection of self strings $S$ represents data and programs stored on the computer. The randomly generated detectors $R$ are matched against elements in $S$, and those $r$ that match (based on an affinity function) are re-

moved. In [Forrest *et al.* 1994][1] an affinity function that checked for the similarity of $r$ consecutive characters at any point in the detector and self strings, called the $r$-contiguous matching rule, was used. The randomly generated detectors that are not removed form the detector collection are used to check for alterations to the system.

Negative selection algorithms have not been constrained to detection of viruses; they have also found application as intrusion detection systems. In this context the self strings $S$ could be a concatenation of source IP, destination IP, and port addresses [Forrest & Beauchemin 2007]. The detector generation stage would be executed during a time when the network was known to be secure. Consequently, a match during the monitoring phase could indicate an anomalous connection, an intrusion. A large amount of work has been dedicated to the development of negative selection algorithms in a variety of application areas and from a theoretical perspective [Balthrop *et al.* 2002, Gonzalez & Dasgupta 2003, Esponda *et al.* 2004].

Despite a considerable amount of examples in the literature, negative selection suffers several critical drawbacks. Defining self can prove problematic; in the case of a network the total variety of safe packets can be enormous, the logistics of capturing this self set can prove difficult. In deriving the set $D$ a huge quantity of randomly generated detectors that match self will have been deleted, thus it can become very inefficient [Freitas & Timmis 2007]. Furthermore, algorithms of this variety have been seen to suffer certain scaling problems: as the universe in which self and nonself elements are defined grows (reflecting the complexity of the detection problem), the number of detectors requires to effectively cover the nonself space becomes difficult to generate [Stibor *et al.* 2005, Timmis *et al.* 2008b].

## *3.4 Danger Theory AIS*

It has been suggested that the integration of mechanisms derived from 'danger theory' [Matzinger 1994] could provide for more effective intrusion detection algorithms than traditional negative selection approaches[Aickelin & Cayzer 2002]. Rather than monitor for the explicit presence of the intruder, danger theory inspired systems could be alerted by the anomalous intruder behaviour. The shift in emphasis is subtle, but not insignificant. Such an intrusion detection system would monitor for signs of 'danger', such as abnormalities in memory usage or disk activity, unexpected or unwarranted frequencies of file changes [Aickelin & Cayzer 2002, Aickelin *et al.* 2003].

An interesting consequence of danger inspired AIS lies in the interpretation of the danger zone. In vivo this is the spatial neighbourhood from where the danger signals originate. In the artificial domain this concept need not be spatial, [Secker *et al.* 2003] place the danger zone in the temporal domain. The concept of danger signals provides danger theory inspired engineering solutions with several advantages over self nonself inspired approaches. Danger signals restrict the domain of nonself to a

[1] This work was the first instance of negative selection being employed in the context of computer security.

manageable size, remove the requirement to observe all self, and instills adaptability regarding scenarios where self and non self boundaries are dynamic [Aickelin & Cayzer 2002].

The main danger theory inspired algorithm that has been developed is the dendritic cell algorithm (DCA) [Greensmith *et al.* 2005, Greensmith *et al.* 2006a]. The DCA is a signal processing algorithm, inspired by the behaviour of dendritic cells. These reside in the body tissues and collect antigen and other (danger) signals that provide a picture of the current state of the tissues. This picture determines whether the antigen has been collected in a safe or dangerous context, and causes dendrictic cells to change into a 'semi-mature' or 'mature' state. The task of the DCA is to classify data items (antigens) as being either benign or malignant in nature. Antigen are associated with concentrations of PAMP signal, danger signal, safe signal, and pro-inflammatory signals. These signals are derived from real biological signals and are mapped onto attributes associated with the data items as follows [Greensmith *et al.* 2006a]:

PAMP. A know signature of abnormal behaviour. This attribute of the data item is highly indicative of an anomaly.
Danger signal. A moderate degree of confidence that this attribute of the data item is associated with abnormal behaviour.
Safe signal. Indicative of normal system operation.
Pro-inflammatory signal. A general sign of system distress.

The main challenge of implementing the DCA is defining how these signals map onto the data items derived from the problem domain [Greensmith *et al.* 2006a].

The DCA, shown in Algorithm 4, operates by maintaining a pool of dendritic cells. From this pool, dendritic cells are randomly selected to sample data items (and related signals) that are presented to the algorithm in a sequential manner. Based on the signals received, dendritic cells produce 'semi-mature' and 'mature' cytokines (immune signalling molecules). At the end of antigen processing DCs are assigned semi mature or mature status according to the levels of the cytokines produced. Every data item is then classified as being benign or malignant on the basis of a majority vote amongst the DCs that sampled it, each voting in accordance to its level of maturity.

Through its focus on behavioural consequences (derived from the signals described above) as opposed to physical presence (in the case of negative selection algorithms), the DCA is able to operate in the presence of dynamically changing environments. However, in its current state [Greensmith *et al.* 2005, Greensmith *et al.* 2006a, Greensmith *et al.* 2006b] the DCA is not able to operate in a true online fashion; data must be collected *a priori* and classification is performed as a final batch operation. Hence, anomalies cannot be detected as they occur. A second potential problem for the DCA is that misclassification can occur around the boundaries where data items switch between 'safe' and 'dangerous' contexts. This is due to multiple sampling of antigen by each DC. The consequence is that the DCA will exhibit significant misclassification when applied to problems where context switches in the data items are frequent [Greensmith *et al.* 2005]. In order to over-

**Algorithm 4** The Dendritic Cell Algorithm (DCA) [Greensmith *et al.* 2005].

```
input: S = a set of antigens, representing data elements to classified as safe or dangerous.
output: K = set of antigens classified as safe.
        L = set of antigens classified as dangerous.

begin
    Create DC pool of 100 dendritic cells.
    for all antigen ag ∈ S do                          ▷ Perform signal processing on ag
        for 10 randomly selected dendritic cells dc ∈ DC do
            Sample ag.
            Update dc.danger, dc.PAMP, and dc.safe signals based on ag.
            Calculate and update concentration of dc.semimatureCytokine output cytokine.
            Calculate and update concentration of dc.matureCytokine output cytokine.
            Calculate and update concentration of dc.coStimulatory output molecules.
            if concentration of dc.coStimulatory > threshold then
                Remove dc from DC and place in M.
                Insert new dc into DC.
            end if
        end for
    end for

    for all dendritic cells dc ∈ M do                  ▷ Differentiation of dendritic cells.
        if concentration of dc.semimatureCytokine > dc.matureCytokine then
            dc.class = semi mature.
        else
            dc.class = mature.
        end if
    end for

    for all antigen ag ∈ S do                          ▷ Classification of antigens
        for all dendritic cells dc ∈ M that sampled ag do
            Calculate if ag presented in mature or semimature context by dc.
        end for
        if ag presented as semimature majority of time then
            Place ag in K.                             ▷ ag is benign
        else
            Place ag in L.                             ▷ ag is malignant
        end if
    end for
end
```

come the limitation of operating in an off-line manner, work by [Lay & Bate 2007] have developed a real-time DCA that is capable of altering schedule overruns in real-time operating systems. The DCA has also been used for behaviour classification on a robotics platform [Castro *et al.* 2007a].

## 4 Reflections and Projections

Artificial immune systems has matured into a well recognised field that tackles a broad range of problem domains. This is best illustrated from the proceedings of the International Conference of Artificial Immune Systems ICARIS[2] [Timmis *et al.* 2003, Nicosia *et al.* 2004b, Jacob *et al.* 2005, Bersini & Carneiro 2006, Castro *et al.* 2007b, Bentley *et al.* 2008]. The field is now at a stage where a number of researchers are reflecting upon its contributions to the wider academic and engineering communities. We assess a number of these reflections and proposed future directions for AIS here.

### *4.1 Evaluation of Current AIS*

[Hart & Timmis 2008] analyse a large collection of AIS engineering applications and categorise these into three classes of problem: anomaly detection, optimisation, and clustering and classification. Considering key works from each class in turn, they attempt to assess and evaluate whether the application of AIS brings any benefits that could not be derived from applying alternative, existing techniques to the problem. Their criteria asserts that it is not sufficient to simply outperform other algorithms on benchmark tests; to be truly successful the AIS must contain features that are not present in alternative paradigms.

Anomaly detection AIS are assessed by [Hart & Timmis 2008] as having had limited success, but the authors make note of recent advances that danger theory inspired algorithms have provided, and state that significant breakthroughs are still possible. For optimisation problems, it is concluded that although optimisation based AIS can and will provide comparable performance to existing methods, they will not offer any distinguishing features that cannot be found elsewhere. For classification and clustering applications the authors conclude that the naturally distributed nature of some AIS algorithms allows for natural parallelisation and distribution across several processors, offering something potentially distinctive. Regarding operation over dynamic data sets, the authors state that by definition, AIS algorithms incorporate some notion of memory, and could therefore outperform alternative learning systems which are purely reactive in nature.

Though their assessment of AIS accomplishments concludes that many are not truly successful, the authors note this is partly due to several shortcomings that have characterised AIS design and application to date [Hart & Timmis 2008]. These include: the methodology through which AIS algorithms capture their inspiring immunology; the attention paid to the effects that certain design decisions impose when engineering AIS systems; the theoretical understanding of AIS algorithms; and the nature of the problems to which AIS have been applied.

---

[2] *http://www.artificial-immune-systems.org/icaris.shtml*

In a similar vein to [Hart & Timmis 2008], [Garrett 2005] studies various AIS to attempt to answer the question of whether AIS research has delivered anything *useful* to date. A useful algorithm in this context is defined by being distinct and effective. An algorithm's distinctiveness is assessed through criteria covering the algorithm's internal representation of the problem and potential solutions, and its computational components. Effectiveness is assessed on the algorithm's performance, including the path through which solutions are obtained, the quality of results obtained through its application to benchmark problems, and the speed at which results can be obtained. In combining the two sets of criteria, and algorithm is said to be useful if it is both effective and distinctive.

The fact that work reflecting on the state of AIS is being conducted is encouraging, and is healthy for the discipline. However it should be noted that the method and criteria employed by [Garrett 2005] in arriving at its conclusions has been criticised for being more of an exercise in classification than in detailed evaluation, and for being highly subjective in nature [Timmis *et al.* 2008a]. Additionally, the criteria focus on performance in relation to benchmark problems. It has been suggested that a downfall of AIS research to date has been its repeated application to benchmark problems, and to areas for which many quality solutions already exist [Hart & Timmis 2008, Timmis *et al.* 2008a]. The effectiveness criteria do not reflect the need for AIS to carve its own niche [Hart & Timmis 2008, Timmis *et al.* 2008a], and provide quality solutions in a problem domain that no other technique can match.

[McEwan *et al.* 2008] question the appropriateness of the shape space representation for AIS with respect to machine learning problems. Typical machine learning problems entail datasets of very high dimensionality. In such a scenario, the adoption of the shape space representation can lead to the "curse of dimensionality": as the dimension of the space increases linearly, its volume increases exponentially, and the quality of locality that affinity measures attempt to discern becomes meaningless as all points approach an equidistance to one another. It has been noted by [Stibor *et al.* 2005] that the task of generating, maintaining, and exploiting an effective set of detectors within such a high dimensional space is computationally intractable.

As an alternative, [McEwan *et al.* 2008] propose marrying the machine learning technique of 'boosting' with immune inspiration. Boosting proposes a strong learning strategy that is derived as a compound decision between multiple (slightly better than random) weak learners. The authors draw analogy to the co-operative nature in which many varieties of immune cells with differing specificities and recognition targets are able to co-operatively mount an effective immune response that hones on a specific target[3].

[3] Cohen's *co-respondence* [Cohen 2000]

## *4.2 Inspiration, Frameworks and Methodologies*

In recent years, there has been a gradual shift in some AIS towards paying more attention to the underlying biological system that serves as inspiration. For example, the development of the DCA (see section 3.4) involved the input from real biological experimentation as inspiration. However, there was no reported sophisticated biological modelling to understand the underlying biology as is suggested by [Stepney *et al.* 2005] and [Timmis *et al.* 2006]. Other examples of this shift back to the underlying biology include [Wilson & Garrett 2004] and [Jacob *et al.* 2004], who have used modelling techniques to build AIS in order to understand underlying immune properties.

The majority of AIS such as those detailed in sections 3.1, 3.2, and 3.3 have taken as their inspiration from well established immunological perspectives. In contrast, [Andrews & Timmis 2005, Andrews & Timmis 2007] advocate expoiting conflicting immune theories as a rich source of potential ideas for the engineer. This is an approach that has been succesfully carried out by [Aickelin & Cayzer 2002, Secker *et al.* 2003, Greensmith *et al.* 2005] in exploiting danger theory and the development of the DCA. AIS can draw significantly more inspiration from the immune system, and the immunological debate surrounding its higher functions, than the relatively simplistic subset of concepts that have served thus far.

Taking this approach, the AIS engineer enjoys the freedom of adopting various immunological models and concepts that best suit the application domain. It is, however, essential to ensure that the concepts employed are correctly abstracted and reasoned about to accurately capture the emergent phenomena from which they are inspired. It has been argued that various immune inspired algorithms have been hampered with a lack of biological accuracy [Timmis 2007]. Typically immune inspired algorithms have fallen prey to "reasoning by metaphor", wherein their operation and structures bear only a weak resemblance to the biological phenomenon that inspired them [Stepney *et al.* 2005], and thus consequentially fail to unlock their full potential [Hart & Timmis 2008].

To combat the problems associated with this apparent weakness in biological metaphors a conceptual framework approach to the development of bio-inspired algorithms has been proposed [Stepney *et al.* 2005], shown in Figure 2. This provides a structure and methodology for biological investigation, abstraction, modelling, and ultimately the construction of algorithms. The process should by interdisciplinary involving at the very least, biologists, mathematicians, and computer scientists. The framework aims to facilitate a better understanding of the targeted underlying biological concepts and to ultimately build more powerful bio-inspired algorithms whilst simultaneously gaining a better understanding of which application domains these algorithms are best suited to.

The first step in the conceptual framework appoach is to probe the biological system through observation and experimentation. These probes are biased towards extracting information concerning the particular biological phenomena of interest. From the information gained, careful abstraction and mathematical modelling will highlight the central processes responsible for the the observed biological phenom-

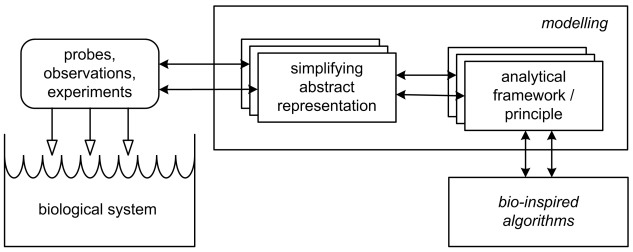


**Fig. 2** The Conceptual Framework approach to deriving biologically inspired algorithms, [Stepney *et al.* 2005]

ena. Analytical computational models may be constructed, which allow for the execution and animation of any underlying model, and can provide a deeper insight into the workings of the model. The observations and mechanisms perceived at this stage will be free from any particular application bias. Finally, these insights can serve as design principles for bio-inspired algorithms which may be applied to non-biological problems [Stepney *et al.* 2005, Hart & Timmis 2008].

A number of AIS works have been inspired by the conceptual framework principles. These include: a computational model of degenerate T-cell receptors [Andrews & Timmis 2006] and adaptable degenerate immune cell receptors [Andrews & Timmis 2008]; and an instantiation of an artificial cytokine network [Hone & van den Berg 2007], which examined the behaviour of the network to elicit any useful properties that could be applied to solving engineering problems [Read *et al.* 2008]. [Newborough & Stepney 2005] also apply many of the conceptual framework ideas to produce a generic framework for population-based bio-inspired algorithms including genetic algorithms, negative selection, clonal selection, particle swarm optimisation and ant colony optimisation.

The conceptual framework of [Stepney *et al.* 2005] also influenced [Twycross & Aickelin 2005] who present a general meta-framework for models incorporating innate immunity. A table of six general properties of the innate immune system is presented and it is claimed that AIS will need to incorporate properties such as these to realise functions of the immune system. Similarly, [Guzella *et al.* 2007] highlight a class of T cell, T regulatory cells, as inspiration for AIS. They suggest that incorporating these cells might lead to more biologically plausible models and algorithms that achieve better results in real-life problems.

While the conceptual framework offers a structured methodology for the development of immune (and other biologically) inspired algorithms, the deployment of these AIS in a particular engineering context also requires careful consideration. Through their examination of AIS application to classification problems [Freitas & Timmis 2007] note several considerations, frequently overlooked, which can significantly affect an algorithm's suitability and performance. They state that the implementor of an AIS algorithm should note the nature of the problem's data, and chose

a representation that intuitively maps the data's characteristics. Altering the data to suit a particular representation, in particular discarding data that is of a different type (for example, disposing of categorical data to fit a continuous valued representation), is bad practice. Rather, the immune inspired algorithm's representation should be tailored to suit the problem's data.

[Freitas & Timmis 2007] also advise careful consideration of the choice of affinity measure for the chosen representation. An affinity measure can be associated with an inductive bias: some basis through which one hypothesis will be favoured over another. An inductive bias is not an undesirable trait, it forms the basis of learning. Yet, care must be taken to ensure that the inductive bias incurred is appropriate for the problem at hand. For example, certain affinity measures have a positional bias, whereby the order of data within the representation can affect the outcome of the affinity measure. If the order of the data is irrelevant to the problem being tackled, then an affinity measure yielding a positional bias might be an inappropriate choice. This work is supported by empirical investigations into the effects of different affinity measures by [Hart & Ross 2004] and [Hart 2005].

## *4.3 Application Domains*

It has been suggested by [Hart & Timmis 2008] that there will be little benefit from applying AIS algorithms to problems of a static nature, over existing and established paradigms. The authors conjecture that the distinctive 'killer application' niche for AIS will require algorithms to exhibit the following properties (quoted verbatim):

- They will be *embodied*.
- The will exhibit *homeostasis*.
- They will benefit from interactions between *innate* and *adaptive* immune models.
- The will consist of *multiple, heterogeneous interacting, communicating components*.
- Components can be easily and naturally *distributed*.
- They will be required to perform *life-long learning*.

Recent applications of AIS in novel problem domains have started to show indications of satisfying these properties, which we review here.

A central function of the immune system is its co-operation with the endocrine and neural systems in the provision of homeostasis to the host [Hart & Timmis 2008]. Homeostasis is "the tendency of a system, esp. the physiological system of higher animals, to maintain internal stability, owing to the coordinated response of its parts to any situation or stimulus tending to disturb its normal condition or function."[4] Hence, since the domain in which in vivo immune systems operate is

[4] American Psychological Association (APA):
homeostasis. (n.d.). Dictionary.com Unabridged (v 1.1). Retrieved June 25, 2008, from Dictionary.com website: *http://dictionary.reference.com/browse/homeostasis*

inherently dynamic; it is not unreasonable to surmise that immune inspired algorithms might be particularly well suited to operation in dynamic environments. The immune system's potential as inspiration for homeostasis in robotics is investigated by [Owens *et al.* 2007]. Here, homeostasis requires: the system to perceive the environment, from multiple perspectives to overcome the inherent problems of sensory malfunction; a repertoire of innate responses that can affect change in the environment or the system directly; the cognition that facilitates the selection of an appropriate effector action in response to perceived input state; and the ability to adaptively correlate sensory information and effector mechanisms such that its actions can dynamically evolve with a changing environment. Similarly, [Neal *et al.* 2006] outline an endocrine-immune inspired homeostatic control system. The artificial immune system allows for low level faults (for example, an overheating motor) to be corrected locally (for example, by turning on a local fan), whilst integration with an artificial endocrine system allows for chronic faults to propagate inflammation throughout the robot's systems. System wide inflammation influences the higher level function of the robot in a global attempt to rectify the fault, for example, the decision by the robot to stop moving, thus allowing the motor to cool down.

The potential for AIS application in the domain of real time systems was demonstrated by [Lay & Bate 2007], who employed the dendritic cell algorithm in the detection of process deadline over runs in an embedded system. The analysis of process executions, and the insurance that all deadlines are met is typically performed statically during the development process. By incorporating adaptive AIS techniques it is hoped that the system is rendered robust, whilst simultaneously reducing development time and costs.

Embodiment in bio-inspired engineering has been investigated by [Stepney 2007], who investigates the intimate coupled nature of a system and its environment. This includes their perceptions and consequent reactions in perturbing one another through high bandwidth, complex, feedback networks. The environment is open, with a quantitatively large and rich variety of information flowing through it, while the system exhibits highly nonlinear dynamics; small input perturbations need not equate to small behavioural modifications. A consequence of embodiment is the co-evolution of the environment with the system. In the context of the danger theory of the immune system [Matzinger 2002], immune cells (system) have learnt to perceive danger signals just as the body (environment) has learnt to provide them. Pathogens experience evolutionary pressure to evade detection, thus contributing to the environment's dynamics [Stepney 2007]. Thus the two are intimately bound. This concurs with the argument for the complex systems view of immunology presented by Cohen [Cohen 2000]. Thus, for engineers to truly capture the complexity of the biology from which they derive their inspiration, they must embody the artificial system within it's artificial environment. Rather than deliberately engineer the interfaces, sensors, and actuators through which they system interacts with its environment.

Though no AIS currently satisfies the conceptual features of embodiment as outlined in [Stepney 2007], [Bentley *et al.* 2005] go some way to addressing similar issues by suggesting that a layer is missing from AIS design. They outline the con-

cept of an artificial tissue layer acting as an interface between a problem space and an AIS. The tissue layer performs some data pre-processing before presenting it to the AIS, allowing for the incorporation of domain specific knowledge and the integration of several data sources, and is the medium through which the AIS responds. An analogy is drawn by [Bentley *et al.* 2005] between the artificial tissue providing an innate response and the AIS providing the adaptive response. As a preliminary investigation, two tissue algorithms are presented by [Bentley *et al.* 2005]. In similar work, [Twycross & Aickelin 2006] present a framework that facilitates the complete encapsulation of an AIS, providing: artificial anatomical compartments within which the AIS's immune elements may operate; and generic receptors for other other immune cells, antigens, and cytokines contained within the compartment. The framework was partially motivated through the possibility to evaluate the performance of several alternative AIS algorithms on the same problem, but the manner in which it interfaces the AIS with the environment, and performs pre-processing is interesting from the perspective of embodiment.

## *4.4 Modelling and Simulating the Immune System*

In recent years building models and simulations of the immune system have become an important aspect of AIS, both as stand alone pieces of work, and as steps towards producing engineering applications using a methodology such as the conceptual framework approach of [Stepney *et al.* 2005]. [Forrest & Beauchemin 2007] note that there is a vast range of modelling approaches applicable to modelling the immune system, each with their own advantages and disadvantages operating at different levels of abstraction.

An overview of many mathematical techniques used for modelling the immune system is provided by [Perelson & Weisbuch 1997]. A large number of these approaches involve the use of differential equations, although other techniques can be applied such as Boolean networks [Weisbuch & Atlan 1988] and the work of [Kelsey *et al.* 2008] who present and analyse a Markov chain model of a cytokine network. Recently, process calculi have been applied to models of the immune system such as [Owens *et al.* 2008]. Process calculi are formal languages from computer science that are used to specify concurrent systems. As biological systems are inherently concurrent, these types of languages seem well suited to biological modelling.

[Forrest & Beauchemin 2007] provide a review of many of the modelling approaches in immunology, with a focus on agent based modelling (ABM). In ABM, components such as cells (and sometimes molecules) are represented individually as *agents* rather than as homogenous populations such as in differential equation techniques. Different agent types typically represent different immune cells types. These agent types are encoded with simple rules extracted from the real biology that govern how they behave and interact. ABM techniques typically employ a concrete notion of space such as that used in cellular automata-like models [Kleinstein & Seiden 2000]. The advantage of ABM is that it allows the observation of agent

population dynamics as they emerge from the interactions of individual agents. An example of ABM is [Beauchemin *et al.* 2006] who investigate the dynamics of *in vitro* infection with a strain of influenza. ABM has also been used to study more computational aspects of applied AIS such as a series of work by [Hart & Ross 2004, Hart 2005, Hart 2006, Hart *et al.* 2006]. These works use a simulation of an idiotypic network to investigate how different models of shape-space and affinity affect the dynamics of the network such as memory capacity and the structures formed, emphasising the need for careful choice of parameters in the engineered systems.

Diagramatic tools have also been used to model the immune system, the most widely used being the unified modelling language (UML) [Fowler 2000], which consists of a set of 13 different types of diagram that can model different aspects of structure and behaviour. The advantage of the UML is its non-domain specific nature and subsequent ability to capture abstractions. The UML (and related diagrams such as statecharts) have started to become a powerful tool in modelling aspects of biological systems. By far the most advanced use of the UML and statecharts in immunology is that of [Efroni *et al.* 2003], who have built a sophisticated and predictive model of T cell maturation in the thymus using a tool called reactive animation, which combines the use of statecharts and other UML diagrams. In addition to the UML, there are other techniques used in software engineering, which [Bersini 2006] suggests can facilitate the development and *communication* of immune modelling. These include object oriented technologies such as object oriented programming and design patterns [Gamma *et al.* 1995]. The perceived benefit is the clarification of immune objects and their relationships. To support this, [Bersini 2006] provides an example of how clonal selection can be modelled with a simple state diagram.

## 5 Summary

This chapter is intended as an overview of the area of artificial immune systems (AIS). It is not meant to be an exhaustive bibliography, but to illustrate that not only the area is one of great diversity, but is one that is expanding into new areas and challenges. It can be noted that AIS is a very diverse area of research, ranging from the modelling of immune systems to the development of algorithms for specific applications. The focus of this chapter has been predominantly on the algorithmic aspect of AIS, however this does not mean that the other aspects of AIS are any less important. Of great importance is the principled development of immune inspired algorithms that capture, in a more than superficial way, the properties and characteristics of the immune system. This is highlighted to different degrees in [Stepney *et al.* 2005] and [Timmis *et al.* 2006] who advocate the careful consideration of the underlying biological system, the use of modelling to help understand that system and the principled abstraction of algorithms and general frameworks from the models. It is also worth noting that there are many different aspects of the immune system that

have served as inspiration, we have reviewed the main ones in this chapter, namely clonal selection, immune networks, negative selection and danger theory. However, there are many untapped possibilities of inspiration from the immune system that have the potential to be developed into novel immune-inspired systems. Such investigations, however, should be firmly rooted in a principled framework [Stepney *et al.* 2005]. As pointed out by [Timmis *et al.* 2008a], a recent paper by Irun Cohen [Cohen 2007] identifies three types of AIS researcher. The first type he calls the "literal" school who build artificial systems that attempt to perform analogous tasks to the actual immune system (e.g. build computer security systems that discriminate between self and non-self ); the second type are those of the "metaphorical" school that take inspiration from the immune system and build artificial systems based on analogies, so the application may be far from analogous to what the immune system does) and a third type of researcher aims to understand immunity through the development of computer and mathematical models. This goes to illustrate the diversity of research that is within the area of AIS and makes the area of AIS a possible avenue for truly interdisciplinary research.

**Acknowledgements** Mark Read is sponsored by the Department of Computer Science, University of York, and Paul Andrews is supported by EPSRC grant number EP/E053505/1.

## References

[Aickelin & Cayzer 2002] Uwe Aickelin and Steve Cayzer. The danger theory and its application to artificial immune systems. In [Timmis & Bentley 2002], pages 141–148.

[Aickelin *et al.* 2003] Uwe Aickelin, Peter J. Bentley, Steve Cayzer, Jungwon Kim, and Julie McLeod. Danger theory: The link between AIS and IDS? In [Timmis *et al.* 2003], pages 147–155.

[Andrews & Timmis 2005] Paul S. Andrews and Jon Timmis. Inspiration for the next generation of artificial immune systems. In [Jacob *et al.* 2005], pages 126–138.

[Andrews & Timmis 2006] Paul S. Andrews and Jon Timmis. A computational model of degeneracy in a lymph node. In [Bersini & Carneiro 2006], pages 164–177.

[Andrews & Timmis 2007] Paul S. Andrews and Jon Timmis. *Alternative Inspiration for Artificial Immune Systems: Exploiting Cohen's Cognitive Immune Model*, chapter 7. In Flower and Timmis [Flower & Timmis 2007], 2007.

[Andrews & Timmis 2008] Paul S. Andrews and Jon Timmis. Adaptable lymphocytes for artificial immune systems. In [Bentley *et al.* 2008], pages 376–386.

[Balthrop *et al.* 2002] J. Balthrop, F. Esponda, S. Forrest, and M. Glickman. Coverage and generalisation in an artificial immune system. In *Genetic and Evolutionary Computation*, pages 3–10, 2002.

[Beauchemin *et al.* 2006] Catherine Beauchemin, Stephanie Forrest, and Frederick T. Koster. Modeling influenza viral dynamics in tissue. In [Bersini & Carneiro 2006], pages 23–36.

[Bentley *et al.* 2005] Peter J. Bentley, Julie Greensmith, and Supiya Ujjin. Two ways to grow tissue for artificial immune systems. In [Jacob *et al.* 2005], pages 139–152.

[Bentley *et al.* 2008] Peter J. Bentley, Doheon Lee, and Sungwon Jung, editors. *Artificial Immune Systems, 7th International Conference, ICARIS 2008*, volume 5132 of *Lecture Notes in Computer Science*. Springer, 2008.

[Bersini & Carneiro 2006] Hugues Bersini and Jorge Carneiro, editors. *Artificial Immune Systems, 5th International Conference, ICARIS 2006*, volume 4163 of *Lecture Notes in Computer Science*. Springer, 2006.

[Bersini 2006] Hugues Bersini. Immune system modeling: The OO way. In [Bersini & Carneiro 2006], pages 150–163.

[Bezerra *et al.* 2004] G. B. Bezerra, L. N. de Castro, and F. J. Von Zuben. A hierachical immune network applied to gene expression data. In *Proc. of the Third Int. Conf. on Artificial Immune Systems (ICARIS'04)*, pages 14–27, Catania, Italy, September 2004.

[Castro *et al.* 2007a] Leandro Nunes de Castro, Fernando Jose Von Zuben, and Helder Knidel, editors. *The Application of a Dendritic Cell Algorithm to a Robotic Classifier*, volume 4628 of *Lecture Notes in Computer Science*. Springer, 2007.

[Castro *et al.* 2007b] Leandro Nunes de Castro, Fernando Jose Von Zuben, and Helder Knidel, editors. *Artificial Immune Systems, 6th International Conference, ICARIS 2007*, volume 4628 of *Lecture Notes in Computer Science*. Springer, 2007.

[Cohen 2000] Irun R. Cohen. *Tending Adam's Garden: Evolving the Cognitive Immune Self*. Elsevier Academic Press, 2000.

[Cohen 2007] Irun R. Cohen. Real and artificial immune systems: computing the state of the body. *Nature Reviews Immunology*, 7:569–574, July 2007.

[Cutello *et al.* 2004] Vincenzo Cutello, Giuseppe Nicosia, and Mario Pavone. Exploring the capability of immune algorithms: A characterization of hypermutation operators. In [Nicosia *et al.* 2004b], pages 263–276.

[Cutello *et al.* 2005] Vincenzo Cutello, Giuseppe Narzisi, Giuseppe Nicosia, and Mario Pavone. Clonal selection algorithms: A comparative case study using effective mutation potentials. In [Jacob *et al.* 2005], pages 263–276.

[de Castro & Timmis 2002a] Leandro N. de Castro and Jon Timmis. *Artificial Immune Systems: A New Computational Approach*. Springer-Verlag, London. UK., September 2002.

[de Castro & Timmis 2002b] L.N. de Castro and J. Timmis. An Artificial Immune Network for Multi Modal Optimisation. In *Proceedings of the World Congress on Computational Intelligence WCCI*, pages 699–704, Honolulu, HI., 2002.

[de Castro & Von Zuben 2000] Leandro N. de Castro and Fernando J. Von Zuben. The clonal selection algorithm with engineering applications. In *Proceedings of GECCO'00, Workshop on Artificial Immune Systems and Their Applications*, July 2000.

[de Castro & Von Zuben 2001] Leandro N. de Castro and Fernando J. Von Zuben. *aiNet: An Artificial Immune Network for Data Analysis*, pages 231–259. Idea Group Publishing, USA, 2001.

[de Castro & Von Zuben 2002] Leandro N. de Castro and Fernando J. Von Zuben. Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 6(2):239–251, June 2002.

[Efroni *et al.* 2003] Sol Efroni, David Harel, and Irun R. Cohen. Towards rigorous comprehension of biological complexity: Modeling, execution, and visualization of thymic t-cell maturation. *Genome Research*, 13:2485–2497, 2003.

[Esponda *et al.* 2004] F. Esponda, S. Forrest, and P. Helman. A formal framework for positive and negative detection schemes. *IEEE Trans. on Systems, Man and Cybernetics Part B: Cybernetics*, 34(1):357–373, 2004.

[Farmer *et al.* 1986] J. D. Farmer, N. H. Packard, and A. S. Perelson. The immune system, adaptation, and machine learning. *Phys. D*, 2(1-3):187–204, 1986.

[Flower & Timmis 2007] Darren Flower and Jon Timmis, editors. In Silico *Immunology*. Springer, 2007.

[Forrest & Beauchemin 2007] Stephanie Forrest and Catherine Beauchemin. Computer immunology. *Immunological Reviews*, 216(1):176–197, April 2007.

[Forrest *et al.* 1994] Stephanie Forrest, Alan S. Perelson, Lawrence Allen, and Rajesh Cherukuri. Self-nonself discrimination in a computer. In *SP '94: Proceedings of the 1994 IEEE Symposium on Security and Privacy*, pages 202–212, Washington, DC, USA, 1994. IEEE Computer Society.

[Fowler 2000] Martin Fowler. *UML Distilled: A Brief Guide To The Standard Object Modeling Language*. Addison Wesley, second edition, 2000.

[Freitas & Timmis 2007] A. Freitas and J. Timmis. Revisiting the foundations of artificial immune systems for data mining. *IEEE Trans. Evol. Comp.*, 11(4):521–540, 2007.

[Galeano *et al.* 2005] Juan Carlos Galeano, Angélica Veloza-Suan, and Fabio A. González. A comparative analysis of artificial immune network models. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2005)*. Springer, 2005.

[Gamma *et al.* 1995] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.

[Garrett 2005] Simon Garrett. How do we evaluate artificial immune systems? *Evol. Comput.*, 13(2):145–177, 2005.

[Goldsby *et al.* 2003] Richard A. Goldsby, Thomas J. Kindt, Barbara A. Osborne, and Janis Kuby. *Immunology*. W. H. Freeman and Company, fifth edition, 2003.

[Gonzalez & Dasgupta 2003] F. A. Gonzalez and D Dasgupta. Anomaly detection using real-valued negative selection. *Genetic Programming and Evolvable Machines*, 4(4):383–403, 2003.

[Greensmith *et al.* 2005] Julie Greensmith, Uwe Aickelin, and Steve Cayzer. Introducing dendritic cells as a novel immune-inspired algorithm for anomaly detection. In [Jacob *et al.* 2005], pages 153–167.

[Greensmith *et al.* 2006a] Julie Greensmith, Uwe Aickelin, and Jamie Twycross. Articulation and clarification of the dendritic cell algorithm. In [Bersini & Carneiro 2006], pages 404–417.

[Greensmith *et al.* 2006b] Julie Greensmith, Jamie Twycross, and Uwe Aickelin. Dendritic cells for anomaly detection. In *IEEE Congress on Evolutionary Computation(CEC 2006)*, pages 664–671, 2006.

[Guzella *et al.* 2007] T. S. Guzella, T. A. Mota-Santos, and W. M. Caminhas. Regulatory t cells: Inspiration for artificial immune systems. In [Castro *et al.* 2007b], pages 312–323.

[Hart & Ross 2004] Emma Hart and Peter Ross. Studies on the implications of shape-space models for idiotypic networks. In [Nicosia *et al.* 2004b], pages 413–426.

[Hart & Timmis 2008] Emma Hart and Jon Timmis. Application areas of AIS: The past, the present and the future. *Journal of Applied Soft Computing*, 8(1):191–201, 2008.

[Hart *et al.* 2006] Emma Hart, Hugues Bersini, and Francisco Santos. Tolerance vs intolerance: How affinity defines topology in an idiotypic network. In [Bersini & Carneiro 2006], pages 109–121.

[Hart 2005] Emma Hart. Not all balls are round: An investigation of alternative recognition-region shapes. In [Jacob *et al.* 2005], pages 29–42.

[Hart 2006] Emma Hart. Analysis of a growth model for idiotypic networks. In [Bersini & Carneiro 2006], pages 66–80.

[Hone & van den Berg 2007] A. Hone and H. van den Berg. Modelling a cytokine network (special session: Foundations of artificial immune systems). In *Foundations of Computational Intelligence*, pages 389–393. IEEE, 2007.

[Honorio *et al.* 2007] L. Honorio, A. Leite da Silva, and D. Barbosa. A gradient-based artificial immune system applied to optimal power flow problems. In [Castro *et al.* 2007b], pages 1–12.

[Jacob *et al.* 2004] Christian Jacob, Julius Litorco, and Leo Lee. Immunity through swarms: Agent-based simulations of the human immune system. In [Nicosia *et al.* 2004b], pages 400–412.

[Jacob *et al.* 2005] Christian Jacob, Marcin L. Pilat, Peter J. Bentley, and Jon Timmis, editors. *Artificial Immune Systems, 4th International Conference, ICARIS 2005*, volume 3627 of *Lecture Notes in Computer Science*. Springer, 2005.

[Jerne 1974] Niels K. Jerne. Towards a network theory of the immune system. *Ann. Immunol. (Inst. Pasteur)*, 125C:373–389, 1974.

[Kelsey *et al.* 2008] Johnny Kelsey, Brian Henderson, Rob Seymour, and Andy Hone. A stochastic model of the interleukin (IL)-1$\beta$ network. In [Bentley *et al.* 2008], pages 1–11.

[Kim & Bentley 2002a] J. Kim and P. J. Bentley. A model of gene library evolution in the dynamic clonal selection algorithm. In [Timmis & Bentley 2002], pages 182–189.

[Kim & Bentley 2002b] Jungwon Kim and Peter Bentley. Immune memory in the dynamic clonal selection algorithm. In [Timmis & Bentley 2002], pages 59–67.

[Kim & Bentley 2002c] Jungwon Kim and Peter J. Bentley. Towards an artificial immune system for network intrusion detection: An investigation of dynamic clonal selection. *Proceedings of the 2002 Congress on Evolutionary Computation CEC2002*, 2002.

[Kleinstein & Seiden 2000] Steven H. Kleinstein and Philip E. Seiden. Simulating the immune system. *Computing in Science and Engineering*, 2(4):69–77, 2000.

[Lay & Bate 2007] N. Lay and I. Bate. Applying artificial immune systems to real-time embedded systems. In *IEEE Congress on Evolutionary Computation, 2007.*, pages 3743–3750, 2007.

[Matzinger 1994] Polly Matzinger. Tolerance, danger, and the extended family. *Annual Review of Immunology*, 12:991–1045, April 1994.

[Matzinger 2002] Polly Matzinger. The danger model: A renewed sense of self. *Science*, 296(5566):301–305, April 2002.

[McEwan *et al.* 2008] Chris McEwan, Emma Hart, and Ben Paechter. Boosting the immune system. In [Bentley *et al.* 2008], pages 316–327.

[Neal *et al.* 2006] Mark Neal, Jan Feyereisl, Rosario Rascunà, and Xiaolei Wang. Don't touch me, i'm fine: Robot autonomy using an artificial innate immune system. In [Bersini & Carneiro 2006], pages 349–361.

[Neal 2003] Mark Neal. Meta-stable memory in an artificial immune network. In [Timmis *et al.* 2003], pages 168–180.

[Newborough & Stepney 2005] John Newborough and Susan Stepney. A generic framework for population-based algorithms, implemented on multiple FPGAs. In [Jacob *et al.* 2005], pages 43–55.

[Nicosia *et al.* 2004a] G. Nicosia, M. Pavone, and V. Cutello. An immune algorithm with hypermacromutations for the 2d hydrophilic-hydrophobic model. In *Congress on Evolutionary Computation*, pages 1074–1080, Portland, Oregon, June 2004.

[Nicosia *et al.* 2004b] Guiseppe Nicosia, Vincenzo Cutello, Peter J. Bentley, and Jon Timmis, editors. *Artificial Immune Systems, Third International Conference, ICARIS 2004*, volume 3239 of *Lecture Notes in Computer Science*. Springer, 2004.

[Owens *et al.* 2007] Nick D. Owens, Jon Timmis, Andrew J. Greensted, and Andy M. Tyrell. On immune inspired homeostasis for electronic systems. In [Castro *et al.* 2007b], pages 216–227.

[Owens *et al.* 2008] Nick D. L. Owens, Jon Timmis, Andrew Greensted, and Andy Tyrrell. Modelling the tunability of early t cell signalling events. In [Bentley *et al.* 2008], pages 12–23.

[Perelson & Oster 1979] Alan S. Perelson and George F. Oster. Theoretical studies of clonal selection: Minimal antibody repertoire size and reliability of self-non-self discrimination. *Journal of Theoretical Biology*, 81(4):645–670, December 1979.

[Perelson & Weisbuch 1997] Alan S. Perelson and Gerard Weisbuch. Immunology for physicists. *Reviews of Modern Physics*, 69(4):1219–1267, 1997.

[Read *et al.* 2008] Mark Read, Jon Timmis, and Paul S. Andrews. Empirical investigation of an artificial cytokine network. In [Bentley *et al.* 2008], pages 340–351.

[Secker & Freitas 2007] A. Secker and A. Freitas. Wairs: Improving classification accuracy by weighting attributes in the airs classifier. In *Proceedings of the Congress on Evolutionary Computation*, pages 3759–3765, 2007.

[Secker *et al.* 2003] Andrew Secker, Alex Freitas, and Jon Timmis. A danger theory inspired approach to web mining. In [Timmis *et al.* 2003], pages 156–167.

[Stepney *et al.* 2005] Susan Stepney, Robert E. Smith, Jon Timmis, Andy M. Tyrrell, Mark J. Neal, and Andrew N. W. Hone. Conceptual frameworks for artificial immune systems. *International Journal of Unconventional Computing*, 1(3):315–338, July 2005.

[Stepney 2007] Susan Stepney. *Embodiment*, chapter 12. In Flower and Timmis [Flower & Timmis 2007], 2007.

[Stibor *et al.* 2005] Thomas Stibor, Philipp Mohr, Jon Timmis, and Claudia Eckert. Is negative selection appropriate for anomaly detection? In *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*, pages 321–328, New York, NY, USA, 2005. ACM.

[Timmis & Andrews 2007] Jon Timmis and Paul S. Andrews. *A Beginners Guide to Artificial Immune Systems*, chapter 3. In Flower and Timmis [Flower & Timmis 2007], 2007.

[Timmis & Bentley 2002] J. Timmis and P. J. Bentley, editors. *Proceedings of the First International Conference on Artificial Immune Systems, ICARIS 2002*. University of Kent Printing Unit, 2002.

[Timmis & Neal 2000] J. Timmis and M. J. Neal. A Resource Limited Artificial Immune System for Data Analysis. *Research and Development in Intelligent Systems XVII*, pages 19–32, December 2000. Proceedings of ES2000, Cambridge, UK.

[Timmis *et al.* 2003] Jon Timmis, Peter Bentley, and Emma Hart, editors. *Artificial Immune Systems, Second International Conference, ICARIS 2003*, volume 2787 of *Lecture Notes in Computer Science*. Springer, 2003.

[Timmis *et al.* 2006] Jon Timmis, Martyn Amos, Wolfgang Banzhaf, and Andy Tyrrell. "Going back to our roots": second generation biocomputing. *International Journal on Unconventional Computing*, 2(4):349–382, 2006.

[Timmis *et al.* 2008a] J. Timmis, P. Andrews, N. Owens, and E. Clark. An interdisciplinary perspectives on artificial immune systems. *Evolutionary Intelligence*, 1(1):5–26, March 2008.

[Timmis *et al.* 2008b] J. Timmis, A. Hone, T. Stibor, and E. Clark. Theoretical advances in artificial immune systems. *Journal of Theoretical Computer Science*, In press(doi:10.1016/j.tcs.2008.02.011), 2008.

[Timmis 2007] Jon Timmis. Artificial immune systems - today and tomorrow. *Natural Computing*, 6(1):1–18, 2007.

[Twycross & Aickelin 2005] Jamie Twycross and Uwe Aickelin. Towards a conceptual framework for innate immunity. In [Jacob *et al.* 2005], pages 112–125.

[Twycross & Aickelin 2006] Jamie Twycross and Uwe Aickelin. Libtissue - implementing innate immunity. In *IEEE Congress on Evolutionary Computation*, pages 499–506, July 2006.

[Watkins *et al.* 2004] Andrew Watkins, Jon Timmis, and Lois Boggess. Artificial immune recognition system (AIRS): An immune-inspired supervised machine learning algorithm. *Genetic Programming and Evolvable Machines*, 5(3):291–317, September 2004.

[Weisbuch & Atlan 1988] Gerard Weisbuch and Henri Atlan. Control of the immune response. *Journal of Physics A: Mathematical and General*, 21(3):189–192, 1988.

[Wilson & Garrett 2004] W. O. Wilson and S. M. Garrett. Modelling immune memory for prediction and computation. In [Nicosia *et al.* 2004b], pages 386–399.