47 An Introduction to Artificial Immune Systems

Mark Read[1] · *Paul S. Andrews*[2] · *Jon Timmis*[3]
[1]Department of Computer Science, University of York, UK
markread@cs.york.ac.uk
[2]Department of Computer Science, University of York, UK
psa@cs.york.ac.uk
[3]Department of Computer Science and Department of Electronics, University of York, UK
jtimmis@cs.york.ac.uk

1 Introduction *1576*

2 The Immune System *1577*

3 Engineering Artificial Immune Systems *1578*

4 Reflections and Projections *1586*

5 Summary *1592*

G. Rozenberg et al. (eds.), *Handbook of Natural Computing*, DOI 10.1007/978-3-540-92910-9_47,

Abstract

The field of artificial immune systems (AIS) comprises two threads of research: the employment of mathematical and computational techniques in the modeling of immunology, and the incorporation of immune system metaphors in the development of engineering solutions. The former permits the integration of immunological data and sub-models into a coherent whole, which can be of value to immunologists in the facilitation of immunological understanding, hypothesis testing, and the direction of future research. The latter attempts to 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.

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 and Timmis 2002a): the self-organization 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 recognize pathogens; and optimization 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 and Timmis 2002a)

The field of AIS also encompasses modeling and simulation techniques to understand the immune system in general (see Timmis et al. (2008a) for a review), however, this chapter focuses on immune-inspired systems for engineering problems.

This chapter is not intended as an extensive review chapter, but its purpose is to present 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 an inspiration for the development of immune-inspired algorithms. We have chosen not to focus on the modeling aspect of AIS, but rather on the main algorithms that have been developed over recent years. This is undertaken in Sect. 3 where we discuss four main immune-inspired algorithms that dominate the literature, namely, clonal selection, immune networks, negative selection, and dendritic cell algorithms, and highlight their usage in terms of applications. Section 4 follows with a discussion on AIS and how researchers have begun to evaluate current AIS and describes new frameworks and methodologies 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 to, and finally we provide a very brief outline of the modeling

approaches that can be found in the literature that are employed to help further our 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 defense system that has evolved to protect its host from pathogens (harmful microorganisms such as bacteria, viruses, and parasites) (Goldsby et al. 2003). It comprises a variety of specialized cells that circulate and monitor the body, various extracellular molecules, and immune organs that provide an environment within which immune cells 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 very complicated with many 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 defense mechanisms that are nonspecific 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 *shape* of 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 defense 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 extracellular molecules. When a sufficiently strong chemical bond occurs between a receptor and another molecule (a ligand), a cascade of intracellular 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 receptors exist: innate receptors that have evolved to recognize specific molecules; and the unique receptors of lymphocytes that are generated during the life time of the individual to recognize previously unseen molecules. The latter of these molecules are generically known as *antigens*, a term given to any molecular structure that can chemically bind to the unique receptors of adaptive 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 signaling 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 cells (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 recognize as being desirable in their systems. Timmis and Andrews (2007), Timmis et al. (2008a), de Castro and Timmis (2002a) have identified these as the following.

Distribution and self-organization. The behavior 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 organized response emerges as a system-wide property derived from the low-level agent behaviors. These immune agents act concurrently making immune processes naturally parallelized.

Learning, adaption, and memory. The immune system is capable of recognizing previously unseen pathogens, thus exhibiting the ability to learn. Learning implies the presence of memory, and the immune system is able to "remember" previously encountered pathogens, as demonstrated by the phenomenon of primary and secondary immune responses. The first time a pathogen is encountered, an immune response (the primary response) is elicited; the next time that pathogen is encountered, a faster and often more aggressive response is mounted (the secondary response).

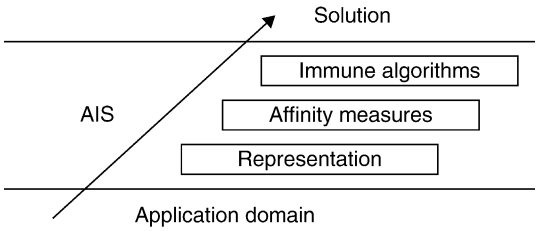
Pattern recognition. Through its various receptors and molecules, the immune system is capable of recognizing a diverse range of patterns. This is accomplished through receptors that perceive antigenic materials in differing contexts (processed molecules, whole molecules, additional signals, etc.). Receptors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

Classification. The immune system is very effective at distinguishing harmful substances (typically viewed as *nonself*) from the body's own tissues (typically viewed as *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). The 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 and Timmis (2002a) have proposed a flexible and generic layered approach to the development of immune-inspired engineering solutions, shown in ● *Fig. 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.

Fig. 1
The layered framework approach to constructing AIS solutions. (Taken from de Castro and Timmis (2002a).)



Given a particular application domain, an appropriate representation of the data must be chosen. In AIS, this typically follows the notion of *shape-space* (Perelson and Oster 1979). Here, molecules m (such as receptors and antigen) exist as points in a shape space S, and can be represented as a string of attributes $m = <m_1, m_2, \ldots, m_L>$ in an L dimensional space, $m \in S^L$. The attributes m_i will represent aspects of a problem domain: patterns to be recognized, functional values to be optimized, combinations of input and proposed actions, etc. de Castro and Timmis (2002a) suggest four data types of which these attributes may belong: real valued; integer valued; hamming valued, finite length strings composed of digits with a finite alphabet; and categorically valued, where values include items such as "name," or "color." 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 convincing critique of the shape-space paradigm and discusses the limitations of such an approach. Their paper is discussed in more detail in Sect. 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 algorithms exist, which can operate independently of the choice of representation and affinity measure, adding dynamics to the algorithm populations based on measurements that the affinity functions provide. Despite this, immune algorithms should be chosen with care based on the problem's data (Freitas and Timmis 2007).

In what is considered to be one of the 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 optimization problems (Hart and Timmis 2008). Four main classes of AIS algorithm have been applied to these problems and each is outlined below.

3.1 Clonal Selection Theory-Inspired AIS

Clonal selection-based algorithms attempt to capture mechanisms of 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-cells 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 and Von Zuben 2002, 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 recognizing unseen patterns. A randomly initialized 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 toward optimization problems by removing the antigen set S, and directly representing the function or domain to be optimized 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 and Stepney 2005).

In *Algorithm 1*, a *generic* clonal selection algorithm is outlined, 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 and Freitas 2007). In the context of dynamic learning, work by Kim and Bentley (2002a, b, c) 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 online manner and then be able to, in a limited manner, adapt to new types of attacks. As a final example, work by Kelsey and Timmis (2003), and Cutello et al. (2004a, b, 2005) have developed particularly effective optimization algorithms based on variants of clonal selection

Algorithm 1 A generic clonal selection algorithm, based on CLONALG (de Castro and Von Zuben 2000, 2002)

input: S = a set of antigens, representing data elements to be recognized.
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

by making use of novel selection and mutation mechanisms tailored specifically for certain types of optimization 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 recognize each other which acts in a self-organizing manner to produce memory, even in the absence of antigen. B-cells interact via receptors to stimulate and suppress each other. This forms a regulatory network that represents 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 optimization to machine learning (de Castro and Timmis 2002b; Honorio et al. 2007; Timmis and Neal 2000; de Castro and Von Zuben 2001; Bezerra et al. 2004). From a machine learning perspective, many of the systems are unsupervised and produce an instant 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 and reduce the dimensionality and complexity of the data (Neal 2003). A generic immune network algorithm, based on aiNet (de Castro and 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) recognize. 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 and Neal 2000). To address this, work by Timmis and 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

Algorithm 2 A generic immune network algorithm, based on aiNet (de Castro and 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

renders the algorithm robust regarding exact parameter values. The algorithm captures well the properties of self-organization 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–nonself discrimination provided the basis for the development of various security-based AIS algorithms. Specifically, a process called negative selection was used, as inspiration, to derive a set of detectors capable of recognizing only nonself. An example of an algorithm based on Forrest et al. (1994) is shown in *Algorithm 3*.

This algorithm was applied to protecting a computer from unauthorized 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 D are matched against elements in S, and those $d \in D$ that match (based on an affinity function) are removed. In Forrest et al. (1994) (this work was the

Algorithm 3 Generic negative selection algorithm (Based on Forrest et al. (1994).)

```
input: S = set of self strings characterizing 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 C.
  for all random strings c ∈ C do
    for all self strings s ∈ S do
      if c matches s then
        Discard c
      else
        Place c 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
```

first instance of negative selection being employed in the context of computer security) 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 from 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 and 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 and Dasgupta 2003; Esponda et al. 2004).

Despite a considerable amount of examples in the literature, it has been argued that negative selection suffers several 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 and 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 required 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 and Cayzer 2002). Rather than monitor for the explicit presence of the intruder, danger-theory-inspired systems could be alerted by the anomalous intruder behavior. The shift in emphasis is subtle, but significant. 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 and 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 neighborhood 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- and nonself-inspired approaches. Danger signals restrict the domain of nonself to a manageable size, remove the requirement to observe all self, and instill adaptability regarding scenarios where self and nonself boundaries are dynamic (Aickelin and Cayzer 2002).

The main danger-theory-inspired algorithm that has been developed is the dendritic cell algorithm (DCA) (Greensmith et al. 2005, 2006a). The DCA is a signal-processing algorithm, inspired by the behavior 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 dendritic 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 pathogen associated molecular pattern (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 known signature of abnormal behavior. 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 behavior.
- *Safe signal.* Indicative of normal system operation.
- *Pro-inflammatory signal.* A general sign of system distress.

The main challenge in 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 (DCs). 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 signaling 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 behavioral consequences (derived from the signals described above) as opposed to physical presence (in the case of negative selection algorithms), the

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

input: S = a set of antigens, representing data elements 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 $ag \in S$ **do** ▷ Perform signal processing on *ag*
for 10 randomly selected dendritic cells $dc \in DC$ **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 $dc \in \boldsymbol{M}$ **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 $ag \in S$ **do** ▷ Classification of antigens
for all dendritic cells $dc \in M$ 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

DCA is able to operate in the presence of dynamically changing environments. However, in its current state (Greensmith et al. 2005, 2006a, b), 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 overcome the limitation of operating in an off-line manner, Lay and Bate (2007) have

developed a real-time, online DCA that is capable of altering schedule overruns in real-time operating systems. The DCA has also been used for behavior classification on a robotics platform (de Castro et al. 2007a).

4 Reflections and Projections

Artificial immune systems has matured into a well recognized 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 (Timmis et al. 2003; Nicosia et al. 2004; Jacob et al. 2005; Bersini and Carneiro 2006; de 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. A number of these reflections and proposed future directions for AIS are assessed here.

4.1 Evaluation of Current AIS

Hart and Timmis (2008) analyze a large collection of AIS engineering applications and categorize these into three classes of problem: anomaly detection, optimization, 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 and 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 optimization problems, it is concluded that although optimization-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 parallelization 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 that this is partly due to several shortcomings that have characterized AIS design and application to date (Hart and 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.

In a similar vein to Hart and 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, an 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 criticized 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 focuses 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 and Timmis 2008; Timmis et al. 2008a). The effectiveness criteria do not reflect the need for AIS to carve its own niche (Hart and 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 data sets 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 in 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 cooperative nature in which many varieties of immune cells with differing specificities and recognition targets are able to cooperatively mount an effective immune response that hones on a specific target (Cohen's *correspondence* (Cohen 2000)).

4.2 Inspiration, Frameworks, and Methodologies

In recent years, there has been a gradual shift in some AIS toward paying more attention to the underlying biological system that serves as inspiration. For example, the development of the DCA (see ❯ Sect. 3.4) involved the input from real biological experimentation as inspiration. However, there was no reported sophisticated biological modeling 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 and Garrett (2004) and Jacob et al. (2004), who have used modeling techniques to build AIS in order to understand underlying immune properties.

The majority of AIS such as those detailed in ❯ Sects. 3.1–3.3 have taken their inspiration from well-established immunological perspectives. In contrast, Andrews and Timmis (2005, 2007) advocate exploiting conflicting immune theories as a rich source of potential ideas for the engineer. This is an approach that has been successfully carried out by Aickelin and

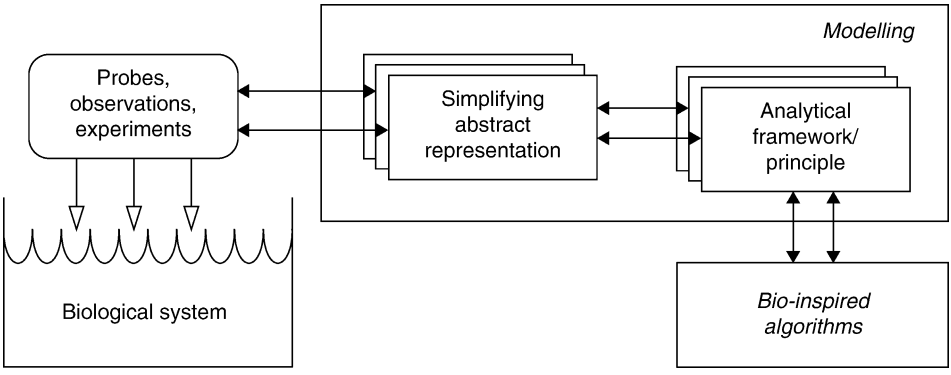
Cayzer (2002), Secker et al. (2003), and Greensmith et al. (2005) in exploiting danger theory and the development of the DCA. AIS can draw significantly more inspiration from the immune system, and the immunological debate surrounding its higher functions, than the relatively simplistic subset of concepts 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 and 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 *Fig. 2*. This provides a structure and methodology for biological investigation, abstraction, modeling, and ultimately the construction of algorithms. The process should be 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 approach is to probe the biological system through observation and experimentation. These probes are biased toward extracting information concerning the particular biological phenomena of interest. From the information gained, careful abstraction and mathematical modeling will highlight the central processes responsible for the observed biological phenomena. Analytical computational models may be constructed, which allow for the execution and animation of any underlying model, and can

Fig. 2
The conceptual framework approach to deriving biologically inspired algorithms (Stepney et al. 2005).



provide a deeper insight into its workings. 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 and 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 and Timmis 2006) and adaptable degenerate immune cell receptors (Andrews and Timmis 2008); and an instantiation of an artificial cytokine network (Hone and van den Berg 2007), which examined the behavior of the network to elicit any useful properties that could be applied to solving engineering problems (Read et al. 2008). Newborough and 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 optimization, and ant colony optimization.

The conceptual framework of Stepney et al. (2005) also influenced Twycross and 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 realize 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 and 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 (e.g., 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 and 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 favored 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 and Ross (2004) and Hart (2005).

4.3 Application Domains

It has been suggested by Hart and 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*.
- They will exhibit *homeostasis*.
- They will benefit from interactions between *innate* and *adaptive* immune models.
- They 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 are reviewed here.

A central function of the immune system is its cooperation with the endocrine and neural systems in the provision of homeostasis to the host (Hart and 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" American Psychological Association (APA). Hence, since the domain in which in vivo immune systems operate is 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 (e.g., an overheating motor) to be corrected locally (e.g., by turning on a local fan), while 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 and 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, while simultaneously reducing development time and costs.

Embodiment in bio-inspired engineering has been investigated by Stepney (2007), who examines the intimate coupled nature of a system and its environment. This includes their perceptions and consequent reactions in perturbing one another through complex high bandwidth 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 behavioral modifications. A consequence of embodiment is the coevolution 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 (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 its artificial environment, rather than deliberately engineer the interfaces, sensors, and actuators through which the 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 in addressing similar issues by suggesting that a layer is missing from AIS design. They outline the concept of an artificial tissue layer acting as an interface between a problem space and an AIS. The tissue layer performs some data preprocessing 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 a similar work, Twycross and Aickelin (2006) present a framework that facilitates the complete encapsulation of an AIS, providing: artificial anatomical compartments within which the AISs immune elements may operate; and generic receptors for 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 preprocessing is interesting from the perspective of embodiment.

4.4 Modeling 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 toward producing engineering applications using methodologies such as the conceptual framework approach of Stepney et al. (2005). Forrest and Beauchemin (2007) note that there is a vast range of modeling approaches applicable to modeling the immune system, each with their own advantages and disadvantages operating at different levels of abstraction.

An overview of many mathematical techniques used for modeling the immune system is provided by Perelson and 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 and Atlan 1988) and the work of Kelsey et al. (2008) which present and analyze 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 modeling.

Forrest and Beauchemin (2007) provide a review of many of the modeling approaches in immunology, with a focus on agent based modeling (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 cell 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 an explicit notion of space, such as that used in cellular automata-like models (Kleinstein and 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 and Ross (2004), Hart (2005, 2006), and 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, emphasizing 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 modeling language (UML) (Fowler 2000), which consists of a set of 13 different types of diagram that can model different aspects of structure and behavior. 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 modeling 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 modeling. 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 modeled with a simple state diagram.

5 Summary

This chapter is intended as an overview of the area of artificial immune systems (AIS), predominantly from an engineering solutions perspective. It is not meant to be an exhaustive bibliography, but serves to illustrate that AIS is an area of great diversity, actively reflecting upon itself, and expanding into new areas and meeting new challenges. The spectrum of AIS research ranges from the modeling of immune systems in aid of immunological study, to the development of algorithms for specific engineering applications. While the predominant focus of this chapter has been on the algorithmic aspect of AIS, the other aspects of AIS research are no less significant. The principled development of immune-inspired algorithms that capture, in a more than superficial manner, the properties and characteristics of the immune system are equally valuable to the discipline's continuing success. 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 modeling to help understand that system, and the principled abstraction of algorithms and general frameworks from those models.

It is worth noting that there are many different aspects of the immune system that have served as inspiration for AIS. Only the main strands have been reviewed in this chapter, namely: clonal selection, immune networks, negative selection, and danger theory. However, there remain many untapped possibilities that are worthy of consideration and study within

principled frameworks, such as the conceptual framework of Stepney et al. (2005). In addition, the area of AIS should seek out challenging application areas that exploit the immune metaphor further than it has to date.

As pointed out by Timmis et al. (2008a), a recent paper by Cohen (2007) identifies three types of AIS researcher. The first type he calls the literal school, they 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 nonself); the second type are those of the metaphorical school who 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 computational and mathematical models. This goes to illustrate the diversity of research that lies within the discipline of AIS, and renders it a promising avenue for truly interdisciplinary research.

Acknowledgments

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 U, Cayzer S (2002) The danger theory and its application to artificial immune systems. In: Timmis J, Bentley PJ (eds) ICARIS 2002: Proceedings of the 1st international conference on artificial immune systems, University of Kent Printing unit, Canterbury, UK, September 2002, pp 141–148

Aickelin U, Bentley PJ, Cayzer S, Kim J, McLeod J (2003) Danger theory: The link between AIS and IDS? In: Bentley PJ, Hart E (eds) ICARIS 2003: 2nd international conference on artificial immune systems, Edinburgh, Scotland, September 2003. Lecture notes in computer science, vol 2787. Springer, New York, pp 147–155

American Psychological Association (APA): Homeostasis. (n.d.). Dictionary.com Unabridged (v 1.1). Retrieved June 25, 2008, from Dictionary.com Web site: http://dictionary.reference.com/browse/homeostasis

Andrews PS, Timmis J (2005) Inspiration for the next generation of artificial immune systems. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 126–138

Andrews PS, Timmis J (2006) A computational model of degeneracy in a lymph node. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 164–177

Andrews PS, Timmis J (2007) Alternative inspiration for artificial immune systems: exploiting Cohen's cognitive immune model. In: Flower D, Timmis J (eds) In silico-immunology. Springer, New York, Chap 7 (2007)

Andrews PS, Timmis J (2008) Adaptable lymphocytes for artificial immune systems. In: Bentley PJ, Lee D, Jung S (eds) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 376–386

Balthrop J, Esponda F, Forrest S, Glickman M (2002) Coverage and generalisation in an artificial immune system. In: Genetic and evolutionary computation. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, July 2002, pp 3–10

Beauchemin C, Forrest S, Koster FT (2006) Modeling influenza viral dynamics in tissue. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 23–36

Bentley PJ, Greensmith J, Ujjin S (2005) Two ways to grow tissue for artificial immune systems. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune

systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 139–152

Bentley PJ, Lee D, Jung S (eds) (2008) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 5132. Springer, New York http://www.artificial-immune-systems.org/icaris.shtml

Bersini H (2006) Immune system modeling: the OO way. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 150–163

Bersini H, Carneiro J (eds) (2006) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin http://www.artificial-immune-systems.org/icaris.shtml

Bezerra GB, de Castro LN, Zuben FJV (2004) A hierachical immune network applied to gene expression data. In: ICARIS 2004: Proceedings of the 3rd international conference on artificial immune systems. Catania, Springer, Berlin/Heidelberg, September 2004, pp 14–27

Cohen IR (2000) Tending Adam's garden: evolving the cognitive immune self. Elsevier Academic Press, London, UK

Cohen IR (2007) Real and artificial immune systems: computing the state of the body. Nat Rev Immunol 7:569–574

Cutello V, Nicosia G, Pavone M (2004a) Exploring the capability of immune algorithms: a characterization of hypermutation operators. In: Nicosia G, Cutello V, Bentley PJ, Timmis J (eds) ICARIS 2004: 3rd international conference on artificial immune systems? Catania, Italy, September 2004. Lecture notes in computer science, vol 3239. Springer, Berlin, pp 263–276

Cutello V, Nicosia G, Pavone M (2004b) An immune algorithm with hyper-macromutations for the Dill's 2D hydrophobic-hydrophilic model. IEEE congress on evolutionary computation, CEC 2004, Portland, Oregon, USA, June 19–23, 2004. IEEE Press, 1:1074–1080

Cutello V, Narzisi G, Nicosia G, Pavone M (2005) Clonal selection algorithms: A comparative case study using effective mutation potentials. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 263–276

de Castro LN, Timmis J (2002a) Artificial immune systems: a new computational approach. Springer-Verlag, London

de Castro L, Timmis J (2002b) An artificial immune network for multi modal optimisation. In: WCCI: Proceedings of the world congress on computational intelligence, Honolulu, HI, May 2002. IEEE, New York, NY, USA, pp 699–704

de Castro LN, Von Zuben FJ (2000) The clonal selection algorithm with engineering applications. In: Proceedings of GECCO'00, workshop on artificial immune systems and their applications. Las Vegas, NV

de Castro LN, Von Zuben FJ (2001) aiNet: an artificial immune network for data analysis. Idea Group Publishing, Hershey, PA, pp 231–259

de Castro LN, Von Zuben FJ (2002) Learning and optimization using the clonal selection principle. IEEE Trans Evol Comput 6(2):239–251

de Castro LN, Von Zuben FJ, Knidel H (eds) (2007a) The application of a dendritic cell algorithm to a robotic classifier. In: ICARIS 2007: Proceedings of 6th international conference on artificial immune systems, Santos, Brazil, August 2007. Lecture notes in computer science, vol 4628. Springer, Berlin

de Castro LN, Von Zuben FJ, Knidel H (eds) (2007b) ICARIS 2007: Proceedings of 6th international conference on artificial immune systems, Santos, Brazil, August 2007. Lecture notes in computer science, vol 4628. Springer, Berlin http://www.artificial-immune-systems.org/icaris.shtml

Efroni S, Harel D, Cohen IR (2003) Towards rigorous comprehension of biological complexity: modeling, execution, and visualization of thymic t-cell maturation. Gen Res 13:2485–2497

Esponda F, Forrest S, Helman P (2004) A formal framework for positive and negative detection schemes. IEEE Trans Syst Man Cybern B Cybern 34(1):357–373

Farmer JD, Packard NH, Perelson AS (1986) The immune system, adaptation, and machine learning. Phys D 2(1–3):187–204

Flower D, Timmis J (eds) (2007) In silico immunology. Springer, New York

Forrest S, Beauchemin C (2007) Computer immunology. Immunol Rev 216(1):176–197

Forrest S, Perelson AS, Allen L, Cherukuri R (1994) Self-nonself discrimination in a computer. In: SP '94: Proceedings of the 1994 IEEE symposium on security and privacy, Oakland, CA, May 1994. IEEE Computer Society, Washington DC, pp 202–212

Fowler M (2000) UML distilled: a brief guide to the standard object modeling language, 2nd edn. Addison-Wesley, Reading, MA

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

Galeano JC, Veloza-Suan A, González FA (2005) A comparative analysis of artificial immune network

models. In: GECCO 2005: Proceedings of the genetic and evolutionary computation conference, Washington, DC, June 2005. Springer, Berlin

Gamma E, Helm R, Johnson R, Vlissides J (1995) Design patterns: elements of reusable object-oriented software. Addison-Wesley, Reading, MA

Garrett S (2005) How do we evaluate artificial immune systems? Evol Comput 13(2):145–177

Goldsby RA, Kindt TJ, Osborne BA, Kuby J (2003) Immunology, 5th edn. W. H. Freeman and Company, New York

Gonzalez FA, Dasgupta D (2003) Anomaly detection using real-valued negative selection. Genet Programming Evolvable Mach 4(4):383–403

Greensmith J, Aickelin U, Cayzer S (2005) Introducing dendritic cells as a novel immune-inspired algorithm for anomaly detection. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference an artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 153–167

Greensmith J, Aickelin U, Twycross J (2006a) Articulation and clarification of the dendritic cell algorithm. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 404–417

Greensmith J, Twycross J, Aickelin U (2006b) Dendritic cells for anomaly detection. In: CEC 2006: IEEE congress on evolutionary computation, Vancouver, Canada, July 2006, pp 664–671

Guzella TS, Mota-Santos TA, Caminhas WM (2007) Regulatory t cells: inspiration for artificial immune systems. In: de Castro LN, Von Zuben FJ, Knidel H (eds) ICARIS 2007: of 6th international conference on artificial immune systems, Santos, Brazil, August 2007. Lecture notes in computer science, vol 4628. Springer, Berlin, pp 312–323

Hart E (2005) Not all balls are round: an investigation of alternative recognition-region shapes. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 29–42

Hart E (2006) Analysis of a growth model for idiotypic networks. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 66–80

Hart E, Ross P (2004) Studies on the implications of shape-space models for idiotypic networks. In: Nicosia G, Cutello V, Bentley PJ, Timmis J (eds) ICARIS 2004: 3rd international conference on artificial immune systems, Catania, Italy, September 2004. Lecture notes in computer science, vol 3239. Springer, Berlin, pp 413–426

Hart E, Timmis J (2008) Application areas of AIS: the past, the present and the future. J Appl Soft Comput 8(1):191–201

Hart E, Bersini H, Santos F (2006) Tolerance vs intolerance: How affinity defines topology in an idiotypic network. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 109–121

Hone A, van den Berg H (2007) Modelling a cytokine network (special session: Foundations of artificial immune systems). In: Foundations of computational intelligence, Honolulu, HI, April 2007. IEEE, New York, pp 389–393

Honorio L, Leite da Silva A, Barbosa D (2007) A gradient-based artificial immune system applied to optimal power flow problems. In: de Castro LN, Von Zuben FJ, Kneidel H (eds) ICARIS 2007: 6th international conference on artificial immune systems, Santos, Brazil, August 2007. Lecture notes in computer science, vol 4628. Springer, Berlin, pp 1–12

Jacob C, Litorco J, Lee L (2004) Immunity through swarms: Agent-based simulations of the human immune system. In: Nicosia G, Cutello V, Bentley PJ, Timmis J (eds) ICARIS 2004: 3rd international conference on artificial immune systems, Calania, Italy, September 2004. Lecture notes in computer science, vol 3239. Springer, Berlin, pp 400–412

Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) (2005) ICARIS 2005: 4th international conference on Artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg http://www.artificial-immune-systems.org/icaris.shtml

Jerne NK (1974) Towards a network theory of the immune system. Ann Immunol (Inst Pasteur) 125C:373–389

Kelsey J, Timmis J (2003) Immune inspired somatic contiguous hypermutation for function optimisation. In: GECCO 2003: Genetic and evolutionary computation conference, Chicago, IL, July 2003. Springer, New York, pp 207–218

Kelsey J, Henderson B, Seymour R, Hone A (2008) A stochastic model of the interleukin (IL)-1β network. In: Bentley PJ, Lee D, Jung S (eds) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 5132. Springer, New York, pp 1–11

Kim J, Bentley PJ (2002a) A model of gene library evolution in the dynamic clonal selection algorithm. In: Timmis J, Bentley PJ (eds) ICARIS 2002: Proceedings of the 1st international conference on artificial immune systems. University of Kent Printing Unit, Canterbury, UK, September 2002, pp 182–189

Kim J, Bentley P (2002b) Immune memory in the dynamic clonal selection algorithm. In: Timmis J, Bentley PJ (eds) ICARIS 2002: Proceedings of the 1st international conference on artificial immune systems. University of Kent Printing Unit, Canterbury, UK, September 2002, pp 59–67

Kim J, Bentley PJ (2002c) Towards an artificial immune system for network intrusion detection: an investigation of dynamic clonal selection. In: CEC2002: Proceedings of the 2002 congress on evolutionary computation. Honolulu, HI, May 2002

Kleinstein SH, Seiden PE (2000) Simulating the immune system. Comput Sci Eng 2(4):69–77

Lay N, Bate I (2007) Applying artificial immune systems to real-time embedded systems. In: IEEE congress on evolutionary computation 2007, Singapore, September 2007, pp 3743–3750

Matzinger P (1994) Tolerance, danger, and the extended family. Annu Rev Immunol 12:991–1045

Matzinger P (2002) The danger model: a renewed sense of self. Science 296(5566):301–305

McEwan C, Hart E, Paechter B (2008) Boosting the immune system. In: Bentley PJ, Lee D, Jung S (eds) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 5132. Springer, New York, pp 316–327

Neal M (2003) Meta-stable memory in an artificial immune network. In: Timmis J, Bentley PJ, Hart E (eds) ICARIS 2003: 2nd international conference on artificial immune systems, Edinburgh, Scotland, September 2003. Lecture notes in computer science, vol 2787. Springer, New York, pp 168–180

Neal M, Feyereisl J, Rascunà R, Wang X (2006) Don't touch me, I'm fine: robot autonomy using an artificial innate immune system. In: Bersini H, Carneiro J (eds) ICARIS 2006: 5th international conference on artificial immune systems, Oeiras, Portugal, September 2006. Lecture notes in computer science, vol 4163. Springer, Berlin, pp 349–361

Newborough J, Stepney S (2005) A generic framework for population-based algorithms, implemented on multiple FPGAs. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 43–55

Nicosia G, Cutello V, Bentley PJ, Timmis J (eds) (2004) ICARIS 2004: 3rd international conference on artificial immune systems, Catania, Italy, September 2004. Lecture notes in computer science, vol 3239. Springer, Berlin http://www.artificial-immune-systems.org/icaris.shtml

Owens ND, Timmis J, Greensted AJ, Tyrell AM (2007) On immune inspired homeostasis for electronic systems. In: de Castro CN, Von Zuben FJ, Knidel H (eds) ICARIS 2007: 6th international conference on artificial immune systems, Santos, Brazil, August 2007. Lecture notes in computer science, vol 4628. Springer, Berlin, pp 216–227

Owens NDL, Timmis J, Greensted A, Tyrrell A (2008) Modelling the tunability of early t cell signalling events. In: Bentley PJ, Lee D, Jung S (eds) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 5132. Springer, New York, pp 12–23

Perelson AS, Oster GF (1979) Theoretical studies of clonal selection: Minimal antibody repertoire size and reliability of self-non-self discrimination. J Theor Biol 81(4):645–670

Perelson AS, Weisbuch G (1997) Immunology for physicists. Rev Mod Phys 69(4):1219–1267

Read M, Timmis J, Andrews PS (2008) Empirical investigation of an artificial cytokine network. In: Bentley PJ, Lee D, Jung S (eds) ICARIS 2008: 7th international conference on artificial immune systems, Phuket, Thailand, August 2008. Lecture notes in computer science, vol 5132. Springer, New York, pp 340–351

Secker A, Freitas A (2007) WAIRS: Improving classification accuracy by weighting attributes in the AIRS classifier. In: Proceedings of the congress on evolutionary computation, Singapore, September 2007. IEEE Press, Singapore, pp 3759–3765

Secker A, Freitas A, Timmis J (2003) A danger theory inspired approach to web mining. In: Timmis J, Bentley PJ, Hart E (eds) ICARIS 2003: 2nd international conference on artificial immune systems, Edinburgh, Scotland, September 2003. Lecture notes in computer science, vol 2787. Springer, New York, pp 156–167

Stepney S (2007) Embodiment. In: Flower D, Timmis J (eds) In silico immunology. Springer, New York, Chap 12

Stepney S, Smith RE, Timmis J, Tyrrell AM, Neal MJ, Hone ANW (2005) Conceptual frameworks for artificial immune systems. Int J Unconventional Comput 1(3):315–338

Stibor T, Mohr P, Timmis J, Eckert C (2005) Is negative selection appropriate for anomaly detection? In: GECCO '05: Proceedings of the 2005 conference on genetic and evolutionary computation, Washington, DC, June 2005. ACM, New York, pp 321–328. doi: http://doi.acm.org/10.1145/1068009.1068061

Timmis J (2007) Artificial immune systems – today and tomorrow. Nat Comput 6(1):1–18

Timmis J, Andrews PS (2007) A beginners guide to artificial immune systems. In: Flower D, Timmis J (eds) In silico immunology. Springer, New York, Chap 3 (2007)

Timmis J, Bentley PJ (eds) (2002) ICARIS 2002: Proceedings of the 1st international conference on artificial immune systems, University of Kent Printing Unit, Canterbury, UK, September 2002

Timmis J, Neal MJ (2000) A resource limited artificial immune system for data analysis. In: Proceedings of ES2000 – Research and development in intelligent systems XVII, Cambridge, UK, December 2000. URL http://www.cs.kent.ac.uk/pubs/2000/1121, pp 19–32

Timmis J, Bentley P, Hart E (eds) (2003) ICARIS 2003: 2nd international conference on artificial immune systems, Edinburgh, Scotland, September 2003. Lecture notes in computer science, vol 2787. Springer, New York http://www.artificial-immune-systems.org/icaris.shtml

Timmis J, Amos M, Banzhaf W, Tyrrell A (2006) "Going back to our roots": second generation biocomputing. Int J Unconventional Comput 2(4):349–382

Timmis J, Andrews P, Owens N, Clark E (2008a) An interdisciplinary perspectives on artificial immune systems. Evol Intell 1(1):5–26

Timmis J, Hone A, Stibor T, Clark E (2008b) Theoretical advances in artificial immune systems. J Theor Comput Sci. doi: 10.1016/j.tcs.2008.02.011

Twycross J, Aickelin U (2005) Towards a conceptual framework for innate immunity. In: Jacob C, Pilat ML, Bentley PJ, Timmis J (eds) ICARIS 2005: 4th international conference on artificial immune systems, Banff, Canada, April 2005. Lecture notes in computer science, vol 3627. Springer, Heidelberg, pp 112–125

Twycross J, Aickelin U (2006) Libtissue – implementing innate immunity. In: IEEE congress on evolutionary computation, Vancouver, Canada, July 2006. pp 499–506

Watkins A, Timmis J, Boggess L (2004) Artificial immune recognition system (AIRS): an immune-inspired supervised machine learning algorithm. Genet Programming Evolvable Mach 5(3):291–317. URL citeseer.ist.psu.edu/watkins04artificial.html

Weisbuch G, Atlan H (1988) Control of the immune response. J Phys A Math Gen 21(3):189–192

Wilson WO, Garrett SM (2004) Modelling immune memory for prediction and computation. In: Nicosia G, Cutello V, Bentley PJ, Timmis J (eds) ICARIS 2004: 3rd international conference on artificial immune systems, Catania, Italy, September 2004. Lecture notes in computer science, vol 3239. Springer, Berlin, pp 386–399