TOMASZ PRACZYK
Polish Naval Academy

THE CONCEPT OF THE SHIP IMMUNE SYSTEM

ABSTRACT

The paper presents the concept of the ship immune system. The task of the system mentioned is to differentiate self objects, i.e. objects that are not dangerous to our ship, from other objects that can be a potential threat. To perform the task the system uses mechanisms adapted from artificial immune systems. Since, the traditional model of artificial immune system assumes objects represented in the form of binary strings, in the paper modifications to this model are proposed. The modifications mentioned either makes it possible to represent objects in the form of real valued vectors or transform real valued representations to simpler binary or integer ones.

Keywords: Immune Systems, Differentiate Self Objects.

INTRODUCTION

The identification of the type "self-alien" is the very often problem during a war. To date, mistakes occur on a battlefield, in which self units are destroyed as a result of fire carried out by other self units. At sea, it can be a problem to identify a submerged submarine. In this case, the potential to identify a ship on the one hand can protect us from destroying a self ship but on the other hand gives us a possibility to avoid an attack performed by a ship pretending to be a self ship.

The existing "self-alien" systems work in the active way. Thus, to identify a ship it is necessary to exchange the information verifying its identity. The paper suggests another solution based on the idea adapted from artificial immune systems (AISs) [[1],[3],[4],[6],[7],[9]]. The solution proposed was called Ship Immunological System (SIS). In the case of SIS, the identification of a ship takes place in the passive way. Moreover, the identification is performed based on a signature of a ship which can be treated as fingerprints very difficult to counterfeit. The next advantage of SIS is a possibility to create it exclusively based on signatures of self ships. To build the system it is not necessary the information about alien ships. The system should detect all that differs from signatures of self ships memorized in the system.

The next beneficial feature of SIS is its adaptation to changing exterior conditions. This means that any change in the own fleet does not entail the necessity to built the new system. The set of detectors imitating alien ships should always adjust to changing signatures of self ships.

Models of AISs created so far use signatures of objects in the form of binary strings. However, ships are usually represented in the form of real valued vectors (e.g. radio signals generated by ships). For this reason to detect nonself ships other than classical detection schemes have to be used. In the paper detection schemes are proposed which reduce real valued vectors to either binary strings or integer vectors. The schemes proposed can be used in SIS to detect alien ships.

The paper is organized as follows: section 2 outlines AIS; section 3 is a presentation of the concept of SIS; section 4 includes description of detection schemes used in SIS; and section 5 is a summary.

ARTIFICIAL IMMUNE SYSTEM

The task of our immune system is to protect us from large variety of viruses, bacteria and other infectious organisms collectively called antigens [[4],[9]]. In order to perform this mission, the immune system must be able to distinguish self cells and molecules, which should protected, from foreign antigens, which should be eliminated. A pathogen is detected if it binds to receptor of some antibody (in fact, the natural immune system consists of large variety of different class of cells and molecules, e.g. B-cells, T-cells and antibodies). Binding is a chemical process that depends on a shape of antibody and pathogen receptors (pathogen receptors are called epitopes). The more similar both receptors the greater chance their owners to bind. Once pathogens are detected, the immune system eliminates them in some manner. Usually, AIS models both self and foreign molecules and cells (or more strictly receptors and epitopes) as binary strings of some fixed length. Chemical binding between them is implemented as a partial string matching. There are many different matching rules that are used in various immune system models, for example Hamming distance, edit distance, *n*-grams, *r*-chunks and *r*-contiguous-bits rule [[4]].

Generally, AIS consists of three sets of binary strings. The first set includes mature detectors imitating antibodies from the natural immune system. The task of mature detectors is to detect all dangerous objects different from self objects. The detection is performed by means of matching rules mentioned above. The second set includes immature detectors generated at random. The immature detectors do not take part in the detecting process.

Each immature detector is tested whether it identifies self objects as nonself ones. If yes, the detector is eliminated. If no, the detector becomes mature. The last set contains binary strings representing self objects. The set is used to test immature detectors.

THE CONCEPT OF THE SHIP IMMUNE SYSTEM

It is assumed that SIS should work as AIS. That is, at first the set of signatures of self ships are created. This set should contain representations of ships in the form of binary strings, integer vectors or in the worst case in the form of real valued vectors. The signatures from this set are used to create the set of mature detectors of nonself ships. Once the set of self signatures is created the system starts to generate immature detectors. The immature detectors are generated at random. Each immature detector is compared to all self signatures. To survive and to become mature an immature detector has to be different from all self signatures. Otherwise, it is eliminated and replaced with other randomly generated immature detector. The process of generating immature detectors is continued during all the "life" of SIS. This makes it possible to adapt the system to continuous changes of signatures. Immature detectors which passed the test become mature detectors. The mature detectors participate in the identification of objects. To detect nonself objects the mature detectors use detecting schemes (or matching rules) measuring similarity between a detector and a signature of an unknown object. The process of detecting nonself objects by means of mature detectors, in detail, is described in the following section. The lifetime of the mature detectors, like their immature counterparts, is not infinite. The mature detectors can also be eliminated. This can happen in two situations. First, when they are responsible for misclassification of a number of objects in turn. Second, when they are selected to be replaced. The replacement of mature detectors with new immature detectors is performed periodically and is necessary in order for the set of mature detectors to include, all the time, up-to-date detectors. The detectors to replace are selected at random, based on their lifetime or based on frequency of detections performed by detectors. The simplified model of SIS is presented in Fig. 1.

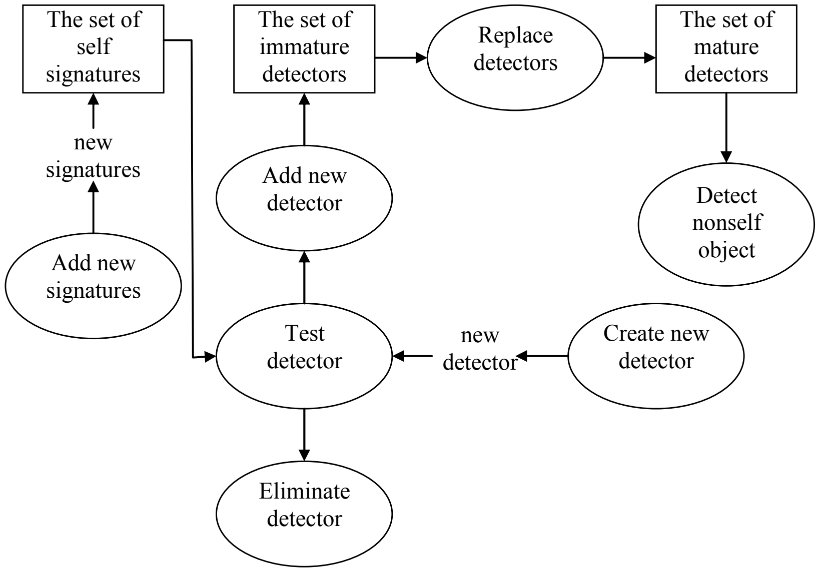

Fig. 1. Model of SIS

DETECTION SCHEMES

To detect nonself objects the mature detectors use detecting schemes. The detection schemes measure similarity between a detector and a signature of an object being identified. Models of AIS created so far use detectors and signatures of identified objects in the form of binary strings. To identify nonself objects the models mentioned usually use the following detecting schemes: Hamming distance, *r*-contiguous-bits rule.

In the case of Hamming distance a nonself object is detected if the distance between any mature detector and the signature of identified object is fewer or equal to *r* where *r* is a parameter of detection scheme. In other words, the nonself object is detected if both compared binary strings, i.e. the string representing one of detectors and the string representing the identified object, agree on at least *r* bits. Positions of bits, where both strings agree, are insignificant.

11***11***10***01***0001

10***11***01***01***1110

Fig. 2. Using Hamming distance with *r*=5; the strings above agree on five bits

In the case of *r*-contiguous-bits (*rcb*) rule [[4]], two binary strings of equal size match if they agree on at least *r* contiguous bits.

11111***00100***01

10110***00100***11

Fig. 3. Using *rcb* rule with *r*=5; the strings above agree on five contiguous bits

In the case of both detecting schemes presented above, the result of matching between two strings depends on the value of *r*. If *r* equals length of both strings then any antibody string can only recognize a single antigen string. In turn, if *r*=0 then each detector string matches each antigen string. Generally, higher values of *r* make a detecting scheme more specific. In turn, lower values make it more general.

The value of *r* has influence on discrimination errors, which AIS can make during work. If a self string is identified as foreign then we deal with the so-called false positive. In turn, false negative occurs when a nonself string is classified as normal. Both errors are dangerous. In the first case, the system attacks oneself while in the second case it does nothing in order to defend oneself against outside threat.

Above, the detection schemes are presented which can be used to compare binary strings. However, in the case of ships, we deal with signatures in the form of real valued vectors. There are two solutions to this problem. First, we can try to built the system based on real valued vectors. Second, real valued vectors representing ships are reduced to binary strings or integer vectors and to compare them we use detecting schemes presented above. Both solutions are described in the following sections.

Detecting objects in the form of real valued vectors

In the case of SIS and generally in the case of all types of AISs the very important issue is to completely surround areas containing self objects by detectors (Fig. 4). Using real valued vectors as detectors and signatures of ships can make it difficult. To enable SIS based on real valued vectors to work two operations are necessary. First, we have to maximally compress the vectors representing antigens and antibodies (in the case of radio signals we can use e.g. FFT). Second, we have to appropriately organize the set of mature detectors. In order for real valued detectors to surround all self signatures, the number of detectors has to be very large. This, in turn, can cause the system to be very slow and in consequence useless. The panacea to this problem is to appropriately organize the set of detectors.

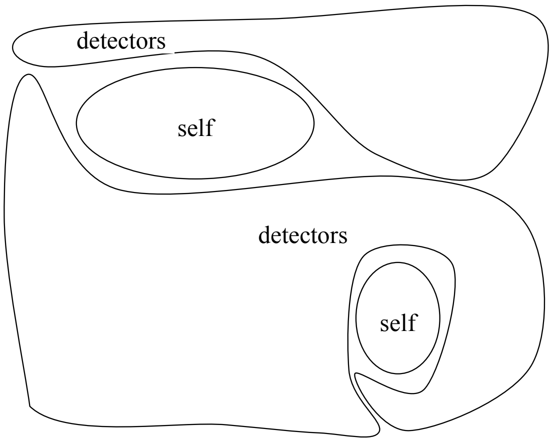

Fig. 4. The set of detectors surrounding two concentrations of self signatures

The whole set of detectors should be divided into parts. Each part should be represented by a single average detector. To determine average detectors, for example, Kohonen neural network can be used. In this case, the detection process starts from comparing an unknown object with all average detectors. In the following phase of detection process only detectors included in the zone of responsibility of the average detector the closest to the identified object are used. The remaining detectors do not take part in the detection process. If some detector from the set of selected detectors is similar to the identified object the object mentioned is treated as nonself.

In the case of real valued vectors the following detection schemes can be used [[10],[11]]:

1. Euclidean distance

$$\mathbf{x}\,\mathrm{M}^{\delta(1)}\mathbf{y} \Leftrightarrow d^{E}(\mathbf{x},\mathbf{y}) \le \delta \tag{1}$$

where: **x**,**y** are real valued vectors;
d^E is Euclidean distance;
δ is the parameter;
$\mathbf{x}\mathrm{M}^{\delta}\mathbf{y}$ means that the vectors **x**,**y** match each other;

2. Partial Euclidean distance

$$\mathbf{x}\,\mathrm{M}_r^{\delta(2)}\mathbf{y} \Leftrightarrow \underset{i}{\exists}\quad \mathbf{x}[i,r]\,\mathrm{M}^{\delta(1)}\mathbf{y}[i,r] \tag{2}$$

where: **x**[*i,r*] is a window of size *r* included in the vector **x**; the window begins from the position *i*;

3.

$$\mathbf{x}\,\mathrm{M}_r^{\delta(3)}\mathbf{y} \Leftrightarrow \underset{i}{\exists}\ \underset{j\in i\ldots i+r}{\forall} \left|\mathbf{x}[j]-\mathbf{y}[j]\right| < \delta \tag{3}$$

where **x**[*i*] is i^{th} element of the vector **x**.

Detecting objects in the form of binary strings or integer vectors

In this case we deal with simpler detectors and signatures of self objects. The set of detectors does not have to be organized in clusters what is the great advantage of the current approach in comparison to the solution presented above. To use binary or integer vectors as detectors and signatures of objects, first, it is necessary to create them from real valued vectors. In the paper the solution is proposed in which the vectors are created in two phases. First, real valued vectors are transformed into integer ones. Afterwards, if necessary, integer vectors are transformed into binary ones. Since, the process of transforming integers into binary strings rather does not require a greater comment (each integer corresponds to binary string, e.g. integer 10 equals binary 0101), in the paper we restrict oneself only to describe the process of transforming real valued vectors into integer ones. Generally, to obtain integer vectors from real valued ones it is necessary to divide the set of values of real valued vectors into equal length segments. For example, if the real valued vectors contain elements from the range <0,1000>, the segments mentioned can be of the size 100. In this case, we would deal with ten segments.

The next operation in transforming real valued vectors into integer ones is to assign numbers to individual segments. The numbers should be assigned in turn, i.e. the first segment <0,100) should obtain the number 0, the segment <100,200) should obtain the number 1 and so on. The last activity in the transformation process is the conversion of real values into numbers assigned to individual segments. The example transformation process is presented in Fig. 5.

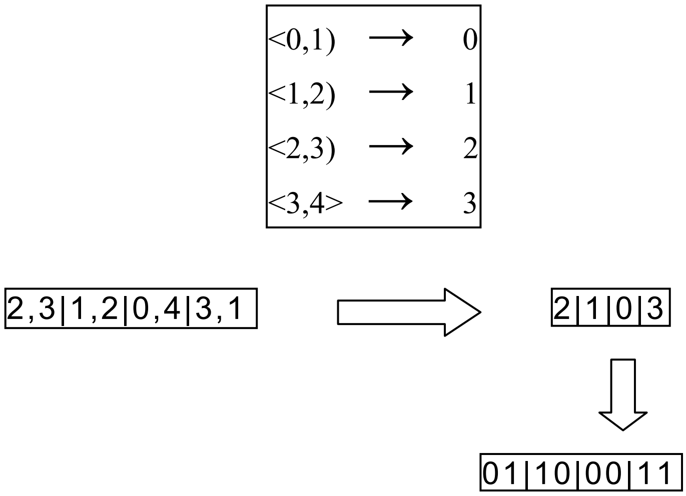

Fig. 5. Transforming real valued vectors into integer and binary ones

To detect nonself objects when both detectors and signatures of self objects are binary strings SIS can use Hamming distance or *rcb* rule presented at the beginning of section 4. In the case of detectors and signatures of self objects represented as integer vectors SIS has to use other detection schemes. Examples of the schemes that can be used to compare integer vectors are presented below:

1. Equivalent of Hamming distance for binary strings

$$\mathbf{x}\,\mathrm{M}^{\delta(4)}\mathbf{y} \Leftrightarrow d^{I}(\mathbf{x},\mathbf{y}) \le \delta \tag{4}$$

where

$$d^{I}(\mathbf{x},\mathbf{y}) = \sum_{i} inequal(\mathbf{x}[i],\mathbf{y}[i]) \tag{5}$$

$$inequal(a,b) = \begin{cases} 1\ if\ \ a \neq b \\ 0\ otherwise \end{cases} \tag{6}$$

2. Equivalent of r-contiguous-bits rule

$$\mathbf{x}\,\mathrm{M}_r^{(5)}\mathbf{y} \Leftrightarrow \underset{i}{\exists} \underset{j \in i \ldots i+r}{\forall} \mathbf{x}[j] = \mathbf{y}[j] \tag{7}$$

When signatures of self objects are represented as integer or binary vectors detectors can take two forms. First, they can also be represented as integer or binary vectors. Second, they can take the form of vectors including integers or binary values and additionally the so-called "don't care symbols" denoted as "#" (don't care symbol is interpreted as zero, one or any integer value). In the former case, the detection process can be performed by means of Hamming distance, *rcb* rule or by means of the detection schemes (4), (7). In the latter case detectors have the same or similar construction as classifiers from Learning Classifier Systems [[2],[5],[8]] and the detection is performed by means of the following detection scheme (Fig 6.):

$$\mathbf{d}\,\mathrm{M}^{(6)}\mathbf{y} \Leftrightarrow \underset{i}{\forall}\left[(\mathbf{d}[i] = \mathbf{y}[i]) \vee (\mathbf{d}[i] \neq \mathbf{y}[i] \wedge \mathbf{d}[i] = \text{"\#"})\right] \tag{8}$$

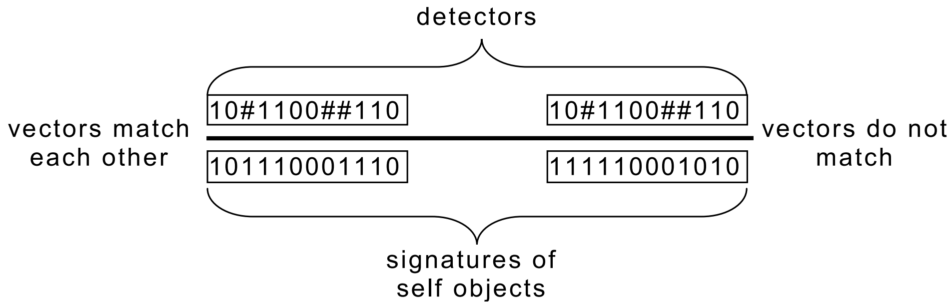

Fig. 6. Using detectors including "#"

CONCLUSIONS

The paper outlines the concept of the ship immune system. In the paper the most important elements of the system are presented, i.e. salient features of the system, the construction of the system, and the course of action of the system. In addition to the general description of the system, the paper also presents different possibilities in the field of detection of ships. Detection schemes are defined that can be used to detect ships represented in the form of real valued vectors. Moreover, the schemes are also presented which require binary or integer representations of ships.

REFERENCES

[1] Balthrop J., Esponda F., Forrest S., Glickman M. Coverage and Generalization in Artificial Immune System. in Proc. Genetic Evolutionary Computation Conf., 2002.

[2] Butz M. V. Rule-based Evolutionary Online Learning Systems: Learning Bounds, Classification, and Prediction. University of Illinois, IlliGAL Report No. 2004034, 2004.

[3] D'haeseleer P., Forrest S., Helman P. An Immunological Approach to Change Detection: Algorithms, Analysis and Implications, Scientific Literature Digital Library - http://citeseer.ist.psu.edu

[4] Esponda F., Forrest S., Helman P. A Formal Framework for Positive and Negative Detection Schemes. IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics, 34(1), pp. 357-373, 2004.

[5] Goldberg D. E. Genetic algorithms in search, optimization and machine learning. Addison Wesley, Reading, Massachusetts. 1989.

[6] Hightower R., Forrest S., Perelson A. The Baldwin Effect in the Immune System: Learning by Somatic Hypermutation. In R. K. Belew and M. Mitchell, editors, Individual Plasticity in Evolving Populations: Models and Algorithms. Addison-Wesley, pp. 159-167, 1996.

[7] Hofmeyr S., Forrest S., Somayaji A. Intrusion Detection using Sequences of System Calls, Scientific Literature Digital Library - http://citeseer.ist.psu.edu

[8] Holland J. H. Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, Michigan. 1975.

[9] Forrest S., Hofmeyr S. Immunology as Information Processing, Design Principles for Immune Systems and Other Distributed Autonomous Systems, eds. Oxford Univ. Press, pp. 361-387, 2000

[10] Praczyk T., Detection of self navigational aids on radar image using ideas from immune systems, Archives of Control Science, Volume 17(LIII), No. 3, pages 241-259, 2007

[11] Praczyk T., Adaptation of r-contiguous-bits scheme borrowed from immune systems to characteristic points of radar image identification, Theoretical and Applied Informatics, vol. 19(2007), pp. 37-56, 2007

Received July 2008
Reviewed May 2009