Effect of Nonlinear Resource Allocation on AIRS Classifier Accuracy

Shahram Golzari [a], Shyamala Doraisamy [b], Md Nasir B Sulaiman [c], Nur Izura Udzir [d]

[a] *Faculty of Computer Science and Information Technology*
University Putra Malaysia, 43400, Serdang, Selangor, Malaysia
and Electrical and Computer Engineering Department,
Hormozgan University, Bandarabbas, Iran
Tel : +60129114236, Fax: +60389466576
Email : golzarihormozi@yahoo.com

[b] *Faculty of Computer Science and Information Technology*
University Putra Malaysia, 43400, Serdang, Selangor, Malaysia
Tel : +60389466548 , Fax : +60389466576
Email:Shyamala@fsktm.upm.edu.my

[c] *Faculty of Computer Science and Information Technology*
University Putra Malaysia, 43400, Serdang, Selangor, Malaysia
Tel: +60389466507, Fax : +60389466576
Email : nasir@fsktm.upm.edu.my

[d] *Faculty of Computer Science andInformation Technology,*
University Putra Malaysia, 43400, Serdang, Selangor, Malaysia
Tel : +60389466521, Fax : +60389466576
Email : izura@fsktm.upm.edu.my

ABSTRACT

Artificial Immune Recognition System (AIRS) is most popular immune inspired classifier. It also has shown itself to be a competitive classifier. AIRS uses linear method to allocate resources. In this paper, two different nonlinear resource allocation methods apply to AIRS. Then new algorithms are tested on 8 benchmark datasets. Based on the results of experiments, one of them increases the accuracy of AIRS in the majority of cases.

Keywords

Artificial Immune System, AIRS, classification, Nonlinear.

1.0 INTRODUCTION

Artificial immune system (AIS) is a computational method inspired by the biology immune system. It is progressing slowly and steadily as a new branch of computational intelligence and soft computing (de Castro & Timmis, 2002; de Castro & Timmis, 2003). One of AIS based algorithms is Artificial Immune Recognition System (AIRS). AIRS is a supervised immune-inspired classification system capable of assigning data items unseen during training to one of any number of classes based on previous training experience. AIRS is probably the first and best known AIS for classification, having been developed in 2001 (Watkins, 2001).

AIRS has four main steps: Initialization, ARB generation, Competition for resources and nomination of candidate memory cell, and finally promotion of candidate memory cell into memory pool. The aim of this study is applying some changes in the resource allocation and competition step of the algorithm. AIRS uses linear method for resource allocation and we use the nonlinear resource allocation methods in this research. Then the algorithms are implemented with modifications and the resulting algorithms tested against benchmark data to determine

the effect of changes on AIRS, specially on its accuracy.
The following section introduces the AIRS algorithm in briefly. Section 3.0 details the changes made to the resource competition of AIRS algorithm and these are tested and evaluated in Section 4.0.

2.0 AIRS

Artificial Immune Recognition System(AIRS) is investigated by Watkins[3]. AIRS can be applied to classification problems, which is a very common real world data mining task. Most other artificial immune system research concerns unsupervised learning and clustering. The only other attempt to use immune systems for supervised learning is the work of (Carter, 2000). The AIRS design refers to many immune system metaphors including resource competition, clonal selection, affinity maturation, memory cell retention and also used the resource limited artificial immune system concept investigated by(Timmis & Neal, 2001).
AIRS has four stages. The first is performed once at the beginning of the process (normalization and initialization), and other stages constitute a loop and are performed for each antigen in the training set: ARB generation, Competition for resources and nomination of candidate memory cell, promotion of candidate memory cell into memory pool. The mechanism to develop a candidate memory cell is as follows:
1. A training antigen is presented to all the memory cells belonging to the same class as the antigen. The memory cell most stimulated by the antigen is cloned. The memory cell and all the just generated clones are put into the ARB pool. The number of clones generated depends on the affinity between the memory cell and antigen, and affinity in turn is determined by Euclidean distance between the feature vectors of the memory cell and the training antigen. The smaller the Euclidean distance, the higher the affinity, the more is the number of clones allowed.
2. Next, the training antigen is presented to all the ARBs in the ARB pool. All the ARBs are appropriately rewarded based on affinity between the ARB and the antigen as follows: An ARB of the same class as the antigen is rewarded highly for high affinity with the antigen. On the other hand, an out of class ARB is rewarded highly for a low value of affinity measure. The rewards are in the form of number of resources. After all the ARBs have been rewarded, the sum of all the resources in the system typically exceeds the maximum number allowed for the system. The excess number of resources held by ARBs are removed in order starting from the ARB of lowest affinity and moving higher until the number of resources held does not exceed the number of resources allowed for the system. Those ARBs, which are not left with any resources, are removed from the ARB pool. The remaining ARBs are tested for their affinities towards the training antigen. If for any class of ARB the total affinity over all instances of that class does not meet a user defined stimulation threshold, then the ARBs of that class are mutated and their clones are placed back in the ARB pool. Step 2 is repeated until the affinity for all classes meet the stimulation threshold.
3. After ARBs of all classes have met the stimulation threshold, the best ARB of the same class as the antigen is chosen as a candidate memory cell. If its affinity for the training antigen is greater than that of the original memory cell selected for cloning at step 1, then the candidate memory cell is placed in the memory cell pool. If in addition to this the difference in affinity of these two memory cells is smaller than a user defined threshold, the original memory cell is removed from the pool.
These steps are repeated for each training antigen. After completion of training the test data are presented only to the memory cell pool, which is responsible for actual classification. The class of a test antigen is determined by majority voting among the k most stimulated memory cells, where k is a user defined parameter.
Some researches have been done to evaluate the performance of AIRS (Watkins & Boggess, 2002; Watkins & Timmis, 2002; Watkins, Timmis & Boggess, 2004; Watkins, 2005). The results show that AIRS is comparable with famous and powerful classifiers.

3.0 NONLINEAR RESOURCE ALLOCATION

Resource competition is one stage of AIRS. The purpose of resource competition in AIRS is improving the selection probability of high-affinity ARBs for next steps. Resource competition is done based on the number of allocated resources for each ARB. According to this resource allocation mechanism, half of resources is allocated to the ARBs in the class of Antigen while the remaining half is distributed to the other classes. The distribution of resources is done by multiplying stimulation rate with clonal rate that shown in (1). Mervah and Boggess (Mervah & Boggess, 2002) have used a different resource allocation mechanism. In their mechanism, the Ag classes occurring more frequently get more resources. Classical AIRS and Mervah study use the linear resource allocation and the number of allocated resources has linearly relation with affinities.

$$Rsources = StimulationRate \times ClonalRate \quad (1)$$

Another approach is nonlinear resource allocation. In this approach, resource allocation is done in nonlinearly with affinities. The difference in resources number between high-affinity ARBs and low affinity ARBs is bigger in this approach than Linear approach. Researchers in (Polat, Kara, Latifoglu & Günes, 2006) have been used fuzzy function for nonlinear resource allocation, but they have not tested their method on bench mark datasets.

The aim of this study is applying nonlinear resource allocation methods on AIRS and comparing of linear and nonlinear resource allocation methods in AIRS. We study the effect of nonlinear resource allocation methods on some characteristics of AIRS such as accuracy, number of final memory cells and building model time. Our focus in this study is on accuracy of algorithm. The comparison of nonlinear methods that have different resource allocation strategies is another purpose of this study.

We use two simple nonlinear functions to do nonlinear resource allocation. Here we have three versions of AIRS: Original version (AIRS), EAIRS and PAIRS. These use different resource allocation strategies. AIRS, PAIRS and EAIRS use strategies that have been shown in (1), (2), (3) Respectively.

$$Rsources = Exp(StimulationRate) \times ClonalRate \quad (2)$$

$$Rsources = \begin{cases} (StimulationRate)^{\frac{1}{2}} \times ClonalRate \ \ if \ SR \geq 0.5 \\ (StimulationRate)^{2} \times ClonalRate \ \ if \ SR < 0.5 \end{cases} \quad (3)$$

In both PAIRS and EAIRS, the difference between the number of resources allocated for high affinity and low affinity ARBs are wider than AIRS. In PAIRS the coefficient of clonal rate is in range [0,1], but in EAIRS this coefficient is greater than 1. Fore same affinities, the number of resources allocated in EAIRS is more than AIRS; But the number of resources allocated in PAIRS depends on amount of affinity (stimulation Rate). If affinity is greater than 0.5, the number of resources allocated in PAIRS is more than AIRS and this relation is vise versa for others.

EAIRS has strongest selection pressure among the algorithms, Because the allocation method of EAIRS allocates many resources for ARBs with high affinity and the competition strategy remove almost of ARBs.

PAIRS do not make the fundamental change on selection pressure. This algorithm changes the coefficient of claonl rate, but remains it between 0 to 1 yet. We choose these two different nonlinear resource allocation methods because of their different characteristics in allocation and selection.

4.0 EXPERIMENTS AND RESULTS

Experiments were carried out in order to determine how PAIRS and EAIRS performed compared to AIRS. One advantage of AIRS is that it is not necessary to know the appropriate settings and parameters for the classifier. The most important element of the classifier is its ability to be self-determined. The used values of the parameters can be found in Table 1.

For this study, a number of datasets were retrieved from the well-known UCI machine learning repository (Newman, 1998). Due to the inability of AIRS to handle datasets in which continuous and discrete attributes are present, the chosen datasets used continuous attributes only. We selected datasets with varying number of attributes, instances and classes, from simple toy datasets to difficult real world learning problems.

Table 1: Algorithm Parameters

Used Parameter	Value
Clonal rate	10
Mutation rate	0.1
ATS	0.2
Stimulation threshold	0.99

Resources	150
Hypermutation rate	10
K value in KNN classifier	1

A 10-fold cross validation approach was used to estimate the predictive accuracy of the algorithms. In this approach, data instances are randomly assigned to one of 10 approximately equal size subsets. At each iteration, all but one of these sets are merged to form the training set while the classification accuracy of the algorithm is measured on the remaining subset. This process is repeated 10 times, choosing a different subset as the test set each time until all data instances have been used 9 times for training and once for testing. The final predictive accuracy is computed over all folds in the usual manner but dividing the number of correct classifications taken over all folds by the number of data instances in all folds.

Table 2 shows the mean classification accuracy obtained when running three algorithms on the selection datasets. The AIRS, PARS and WAIRS columns show the mean predictive accuracy of the respective algorithm.

Table 2: Classification Accuracies comparison (%)

Dataset	AIRS	PAIRS	EAIRS
sonar	80.29	**81.25**	70.43
Breast cancer	96.42	**96.87**	96.13
Wave form	77.1	**77.66**	76.66
Iris	**96**	94.67	94
Ionosphere	84.9	86.32	**87.18**
Pima Diabetes	70.83	**71.18**	69.01
German Credit	**71.1**	69.7	69.2
TIC-TAC	83.1	**84.76**	83.3

PAIRS, AIRS and AIRS have achieved best accuracy on 5, 2 and one datasets respectively. PAIRS is better than AIRS on 6 datasets, but EAIRS is better than AIRS on only one dataset. These results shows that PAIRS has increased the accuracy. PAIRS Improves the selection probability of high affinity ARBs and selects more high affinity ARBs and less low affinity ARBs for next generation. This algorithm remains more number of high affinity ARBs in environment to compete at next steps. In contrast, EAIRS allocates many resources for high affinity ARBs and removes all of low affinity ARBs and almost of high affinity ARBs too. This algorithm selects limited number of ARBs for next generation; Therefore, this algorithm loses the diversity and supports the premature and fast convergence of process and maybe not accurate results.

We repeated the experiments in same conditions and calculate the number of memory cells remained in the algorithms after training phase. Results have been shown in Table 3.

About EAIRS, the results support our previous findings. EAIRS has more data reduction on all datasets. This results show EAIRS has strong selection pressure and loses diversity and does not allow most of ARBs to remain in environment and competition in next steps of algorithm and finally selects some premature candidates as memory cells.

Table3: The number of memory cells remained in the system after training

Dataset	AIRS	PAIRS	EAIRS
Sonar	167	170	158
Breast cancer	304	304	278
Wave form	4693	4695	4232
Iris	51	45	52
Ionosphere	219	219	205
Pima Diabetes	540	543	481
German Credit	963	953	924
TIC-TAC	954	951	935

The difference between numbers of memory cells is not significant in AIRS and PAIRS.

Necessary time for building model is another parameter that obtained from experiments. The results for this parameter have been shown in table 4.

Table 4: Model building time (s)

Dataset	AIRS	PAIRS	EAIRS
Sonar	4.92	4.8	1.05
Breast cancer	2.3	3	0.83
Wave form	84.8	95.97	44.75
Iris	0.38	0.47	0.2
Ionosphere	5.38	5.83	1.36
Pima Diabetes	2.09	2.92	1.03
German Credit	8.69	10.05	2.77
TIC-TAC	2.61	3.06	1.45

As we expected, based on the previous results, EAIRS algorithm takes minimum time for all datasets. However we faced with surprising results for PAIRS. PAIRS take more time than AIRS except on one dataset. The reason of this issue could be found at future researches.

5.0 Conclusion

In this paper, we applied two nonlinear resource allocation methods on AIRS. One of them was based on the square root of affinity and another was based on the affinity exponential. Experiments were conducted on benchmark datasets. The results show that the exponential method decreases the model building time and number of memory cells in all cases. However, the square root based method increases the accuracy of algorithm in the majority of cases.

References

Carter, J. H. (2000). The immune systems as a model for pattern recognition and classification. *Journal of the American Medical Informatics Association, 7(1)*, 28-41.

de Castro, L. N., & Timmis, J. (2003). Artificial Immune Systems as a novel Soft Computing Paradigm. *Soft Computing Journal*, 7(7).

de Castro, L. N., & Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach:* Springer.

Marwah, G., & Boggess, L. (2002). Artificial immune systems for classification: Some issues. *In Proceedings of the first international conference on artificial immune systems, University of Kent at Canterbury, England*, 149–153.

Newman, D. J. (1998). UCI Repository of machine learning databases. Retrieved September 2007 from: http://www.ics.uci.edu/~mlearn/MLRepository.html.

Polat, K., Kara, S., Latifoglu, F., & Günes, S. (2006). A Novel Approach to Resource Allocation Mechanism in Artificial Immune Recognition System: Fuzzy Resource Allocation Mechanism and Application to Diagnosis of Atherosclerosis Disease. *ICARIS 2006*, 244-255.

Timmis, J., & Neal, M. (2001). A Resource Limited Artificial Immune System. *Knowledge Based Systems, 14(3)*, 121-130.

Watkins, A. (2001). *AIRS: A Resource Limited Artificial Immune Classifier*. M.S. thesis, Department of Computer Science, Mississippi State University.

Watkins, A. (2005). *Exploiting Immunological Metaphors in the Development of Serial, Parallel, and Distributed Learning Algorithms*. PhD. Thesis, Computer Science, University of Kent, Canterbury, England.

Watkins, A., Timmis, J., & Boggess, L. (2004). Artificial Immune Recognition System (AIRS): An Immune-Inspired Supervised Learning Algorithm. *Genetic Programming and Evolvable Machines, 5(3)*, 291-317.

Watkins, A., & Boggess, L. (2002). A new classifier based on resource limited artificial immune systems. *In Congress on Evolutionary Computation. Part of the World Congress on Computational Intelligence,,Honolulu, HI.*, 1546– 1551.

Watkins, A., & Timmis, J. (2002). Artificial Immune Recognition System (AIRS): Revisions and Refinements. *In 1st International Conference on Artificial Immune Systems (ICARIS 2002), Canterbury, UK*,173-181.