An Investigation into the Source of Power for AIRS, an Artificial Immune Classification System

Donald E. Goodman, Jr.[1], Lois Boggess[2], and Andrew Watkins[3]

[1]Department of Psychology, Mississippi State University
Mississippi State, MS 39762

[2]Department of Computer Science and Engineering, Mississippi State University
Mississippi State, MS 39762

[3]Computing Laboratory, University of Kent at Canterbury
Canterbury, Kent, England. U.K. CT2 7NF

Abstract - **The AIRS classifier, based on metaphors from the field of artificial immune systems, has shown itself to be an effective general purpose classifier across a broad spectrum of classification problems. This research examines the new classifier empirically, replacing one of the two likely sources of its classification power with alternative modifications. The results are slightly less effective, but not statistically significantly so. We conclude that the modifications, which are computationally somewhat more efficient, provide fast test versions of AIRS for users to experiment with. We also conclude that the chief source of classification power of AIRS must lie in its replacement and maintenance of its memory cell population.**

INTRODUCTION

Late in 2001, a new classifier was introduced, based on principles from the discipline of Artificial Immune Systems. The classifier was AIRS (Artificial Immune Recognition System), and it was interesting not only because it showed that Artificial Immune Systems could be used for classification but also because it was surprisingly successful as a general purpose classifier.

This paper empirically explores the possible sources of classification power of this new classifier. We believe such an exploration is important for the following reasons:

a) AIRS is effective in a broad array of different classification problems, including problems with large dimensioned feature space, problems with many classes, and problems with real-valued and discrete features.

b) Some general purpose classifiers perform poorly until an appropriate architecture for the classifier is determined by the researcher, and the search for the appropriate architecture may require substantial effort. For the majority of problems to which it has been applied, using the default parameters with which AIRS is delivered produces results which are within a couple of percentage points of the best results that AIRS obtains. And the best results that AIRS obtains are usually highly competitive.

c) AIRS is self-adjusting for the feature of its architecture that is most descriptive of the problem space. In fact, when another classifier, Kohonen's LVQ, was instructed to use the same number of cells that AIRS determined to be a good characterization of the problem space, the classification accuracy of LVQ improved [1]

d) Although we have experience with an array of general-purpose classifiers, none in our experience has performed as consistently strongly as AIRS across the same gamut of classification types.

AIRS was originally conceived in an attempt to demonstrate that Artificial Immune Systems were amenable to the task of classification. The AIRS algorithm was motivated in particular by resource limited Artificial Immune Systems [2][3]. Although the motivation was initially simply to show that classification was possible using this paradigm, the algorithm was tested a broad range of publicly-available classification tasks, and proved to be highly effective. It has been tested on problems with up to 279features and on problems with up to 12 classes. [1][4][5][6][7].

[1] and [5] both investigated multiple-class problems in which the number of classes in the problem space was fairly large, and each encountered a widely-studied publicly available classification problem for which AIRS appears to be the most successful single general-purpose classifier for the problem.

This paper explores possible sources of the power of the AIRS algorithm. First, we briefly describe the basic classification algorithm, particularly the memory cell pool which is used for the actual classification task after training, and the "training" algorithm that derives this memory cell pool. Since this memory cell pool is in a sense the AIRS algorithm's depiction of the important parts of the problem space, we compare the characteristics of the members of the memory cell pool with the "antigens" - the training instances

- from the data for a variety of the classification problems. We then explore the effect of replacing the memory cell derivation algorithm of the original AIRS with a variety of plausible alternatives, observing any resulting changes in effectiveness in the overall AIRS classification performance. Finally, we apply our observations to the goal of deducing the source of power of the AIRS algorithm when applied to the problem of classification.

THE AIRS CLASSIFIER

As mentioned in the introduction, the initial intent in developing AIRS was to demonstrate that a classifier built on the principles of artificial immune systems could be an effective general purpose classifier suitable for a broad variety of classification tasks.

Artificial Immune Systems had already been applied successfully in other domains: in particular, Artificial Immune Systems are an appealing metaphor for applications in Computer Security. The ability to distinguish between Self and Non-self has considerable attraction in that field, and the immune system concept of negative selection of maturing recognizers based on response to Self has been successfully applied to systems whose purpose is to determine whether the integrity of data has been violated [8][9]. There is also considerable research into the application of such evolved recognizers in the field of Intrusion Detection, especially as applied to the question of whether traces of operating system calls correspond to normal behavior or not [10][11][12].

One previous system, [13], had been designed for the purpose of classification. However, it was devised by a medical professional, was highly complex, and was not sufficiently described to replicate. Work by Timmis and others on clustering using Artificial Immune Systems [3][14][15] was appealing, and the further development of that work using the concept of resource limitation [16] became the inspiration for AIRS.

LOOKING FOR THE SOURCE OF CLASSIFICATION POWER

The goal was to show that an Artificial Immune System could be an effective general purpose classifier; we did not actually expect it to be quite as effective as it proved to be. Table 1 on the next page shows the comparative effectiveness of the original AIRS algorithm [4] in terms of reported accuracy rates in the literature. The number reported in the table for AIRS is its average accuracy over multiple runs of 10-way cross-validation. The figures for the other classifiers are derived from [17] and [18] (which are themselves also normally averages of multiple runs of cross-validation), and indicate whether the accuracy was greater (positive) or less (negative) than the average accuracy for AIRS on the same problem.

In broad terms, training the AIRS classifier involves two interrelated processes. In nature, lymphocytes (B-cells and T-cells) respond to an invading antigen and those B-cells which match the invader closely enough begin mutating to generate even closer matches, as part of the process of attacking the invader. As a classifier-in-training, for any presenting antigen (training instance from the training data), AIRS uses a pool of B-cells, some of which are mutations of an existing B-cell which most closely resembles the presenting antigen, and some of which are simply randomly generated cells. In the current version of AIRS, stimulation is inversely proportional to distance in feature space: the smaller the Euclidean distance between antigen and B-cell, the greater the stimulation. The B-cells which are most highly stimulated by exposure to the antigen begin cloning and mutating. The least stimulated B-cells die out. The process continues over multiple generations, with the most-stimulated B-cells being retained and the least stimulated ones being eliminated. Once the average stimulation level of the entire B-cell population reaches a threshold, which can be determined by the user, the process stops, and the cell which is most stimulated by the antigen becomes a candidate for promotion to a memory cell.

The second training process referred to above is the selection and maintenance of the set of memory cells. In nature, most of the enormous number of lymphocytes generated in combating an invader die. But a relatively small number of the cells remain in the immune system indefinitely, with the result that any successive invasions by the same or similar antigens are met with an immediate response by the immune system. Although there is controversy over whether the cells themselves have an indefinite lifespan or whether the immune system has a method of replicating the actual cells, the net effect is the same - there are cells that represent a very long-term memory of invaders to the system. In AIRS, when a B-cell becomes a candidate for inclusion in the long-term memory cell population, it must first prove to be more stimulated by the current antigen than any other existing memory cell which responds to the classification class to which the antigen belongs. Provided that this is the case, the candidate cell is included in the memory cell population. Additionally, if the new memory cell is sufficiently similar to the memory cell which originally was most stimulated by the invading antigen (where the definition of "sufficiently similar" is controlled by a parameter which the user can adjust), the new memory cell actually replaces this other cell. This mechanism contributes both to the generalization capabilities of the AIRS classifier and to the data reduction capabilities of the classifier. Typically there are fewer than half as many memory cells in a trained AIRS system as there were training instances, and only a small fraction of the memory cell population is identical to the training instances which engendered them. For more details of the current AIRS algorithm, see [19].

Table 1.

Comparison of classifiers on the five classification tasks. Except for AIRS data, these results are taken from Duch [17][18].When Duch's reported accuracy seems to disagree with Duch's ranking, both the ranking and the reported accuracy are retained.

	Cleveland heart disease		**Iris**		**Ionosphere**		**Diabetes**		**Sonar**	
1	IncNet	+6.8%	Grobian (rough)	+3.3%	3-NN + simplex	+3.8%	Logdisc	+3.6%	TAP MFT Bayesian	+8.3%
2	28-NN, stand, Euclid, 7 features	+2.6 - +1.6%	SSV	+1.3%	3-NN	+1.8%	IncNet	+3.5%	Naïve MFT Bayesian	+6.4%
3	Fisher discriminant analysis	+1%	C-MLP2LN	+1.3%	IB3	+1.8%	DIPOL92	+3.5%	SVM	+6.4%
4	LDA	+1.3%	PVM 2 rules	+1.3%	MLP + BP	+1.1%	Linear Discr. Anal.	+3.4 - +3.1%	Best 2-layer MLP + BP, 12 hidden	+6.4%
5	16-NN, stand, Euclid	+1.4 - +0.2%	PVM 1 rule	+0.6%	**AIRS**	**94.9%**	SMART	+2.7%	MLP+BP, 12 hidden	+0.7%
6	FSM, 82.4-84% on test only	+0.8%	**AIRS**	**96.7%**	C4.5	0%	GTO DT (5xCV)	+2.7%	MLP+BP, 24 hidden	+0.5%
7	Naïve Bayes	+0.2 - -0.7%	FuNe-I	0.0%	RIAC	-0.3%	ASI	+2.5%	1-NN, Manhatten	+0.2%
8	**AIRS**	**83.2%**	NEFCLASS	0.0%	SVM	-1.7%	Fischer discr. anal	+2.4%	**AIRS**	**84.0%**
9	SNB	-0.1%	CART	-0.7%	Non-linear perceptron	-2.9%	MLP+BP	+2.3%	MLP+BP, 6 hidden	-0.5%
10	LVQ	-0.3%	FUNN	-1.0%	FSM + rotation	-2.1%	LVQ	+1.7%	FSM - methodology?	-0.4%
11	kNN, k=27, Manh	+0.2 - - 1.0%			1-NN	-2.8%	LFC	+1.7%	1-NN Euclidean	-1.8%
12	GTO DT (5xCV)	-0.7%			DB-CART	-3.6%	RBF	+1.6%	DB-CART, 10xCV	-2.2%
13	kNN, k=19, Euclidean	-0.3 - -1.9%			Linear perceptron	-4.2%	NB	+1.4- -0.3%	CART, 10xCV	-16.1%
14	LDA (all vectors, 85% on train)	-1.4%			OC1 DT	-5.4%	kNN, k=22, Manh	+1.4%		
15	SVM (5xCV)	-1.7%			CART	-6.0%	MML	+1.4%		
16	kNN (k=1?)	-1.7%			GTO DT	-8.9%	SNB	+1.3%		
…							**. . .**			
22							**AIRS**	**74.1%**		
23							C4.5	-0.9%		
	others below 16th rank include MLP with Backprop, CART, RBF, Gaussian EM, ASR, C4.5, and a number of WEKA tools, among others						11 others reported with lower scores, including Bayes, Kohonen, kNN, ID3 …			

The experiments described in this paper test the hypothesis that the power of the AIRS classifier resides in the method of deriving a candidate memory cell from the B-cells. An empirical means of testing that conjecture is to replace that part of the AIRS algorithm with a function which directly generates a candidate memory cell drawn from some suitably constrained probability distribution, rather than filtering one from the random mutations of a population. (While the functions explored are not based on a natural metaphor, there may be natural justification for not using random mutation, since it appears that in nature the mutation of lymphocytes in response to an invader is not a random process.)

Frankly, we expected the performance of AIRS to drop off sharply when a function was substituted. When that was not the case with the first function that we experimented with, an elliptical probability distribution, we looked at how close in feature space the regular AIRS candidates for memory cell status were to the training vectors that caused them to be generated. That is, for each training vector presented to AIRS, a memory candidate cell is evolved. In general the feature space is high-dimensional, so it is not possible to look at a spatial representation of the training vectors and the resulting memory candidates. But it is possible to compute the distance between the training vectors and their respective candidate cells, and to plot a histogram of how many cells fall into respective distances from their training vectors.

Figure 1 is one of the resulting histograms. It shows the distribution of the distances of the memory cells from their respective training vectors for ten-way cross-validation, applying the normal AIRS algorithm (not the functional replacement) to the Pima Indian Diabetes classification problem [20]. AIRS normalizes the feature vector space so that all feature dimensions have the same range and so that the maximum Euclidean distance between any two vectors is 1.0. The farthest outliers of this set were slightly more than one quarter of this distance.

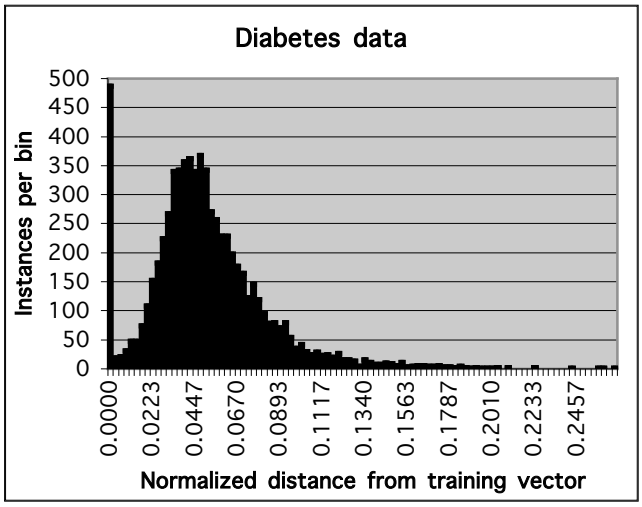

Figure 1. Distance of candidate memory cells from training antigen. Maximum normalized distance possible is 1.0. Y axis is number of antigens out of a total of 6920.

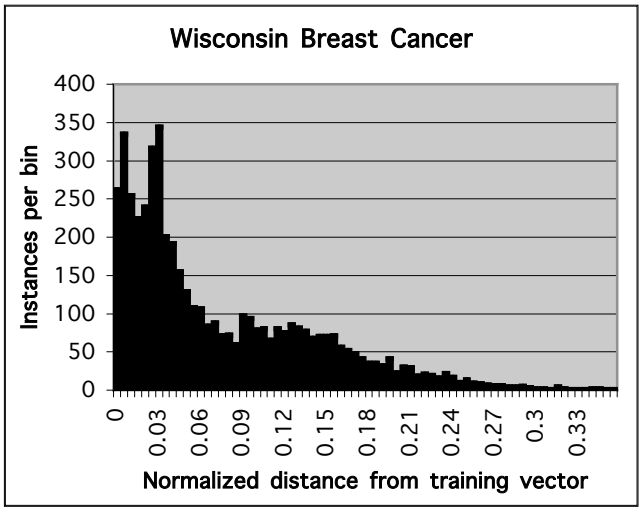

Figure 2. Two data sets were somewhat different from the preceding. Shown above is the distance of candidate memory cells from training antigen for Wisconsin Breast Cancer data.

In addition to the Diabetes data set, we also derived analogous histograms for the Wisconsin Breast Cancer data set, the balance scale data set, the credit.crx problem, iris, ionosphere and SPECT data sets, all from the University of California at Irvine Machine Learning Repository [20], and for an artificial three-class problem whose feature vectors are uniformly distributed over a subset of the real plane, with convoluted boundaries between the non-linearly separable classes (see [1] for a description of the latter data set). We hypothesized that the uniformly distributed nature of the training instances of the latter, in contrast to the "bursty" nature of feature vectors from naturally occurring classification problems, might affect how close the memory cells would be to their respective training antigens. However, the overall shape of the histograms was similar for all the problems except the balance scale problem, the breast cancer problem, and the SPECT problem. The histogram for the Wisconsin Breast Cancer data is shown in Figure 2.

Both the balance scale data and the Wisconsin Breast Cancer data produced histograms with no deep "valley" between the initial spike at zero and the first peak. They also both show a noticeable secondary rounded peak in the distribution. The histogram for SPECT is quite different from all other histograms and is shown in Figure 3. While it confirms that AIRS behaves differently for different data sets, it is also the case that AIRS's accuracy for SPECT is low. The distribution, which shows most of the memory cell candidates a distance of between 0.25 and 0.5 in a normalized space where the maximum distance between two entities is 1.0, may reflect a difficulty in distinguishing between training vectors on the basis of Euclidean distance in feature space.

We then experimented with three functions to replace the part of AIRS which generates candidate memory cells:

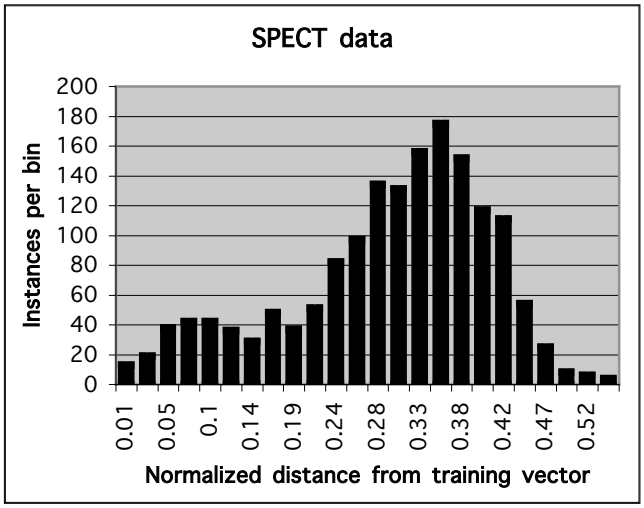

Figure 3. One distribution was markedly different from all the others. Above is the distribution of distances of candidate memory cells from the training antigens for the SPECT data set.

Function 1 (Mod 1): This was the initial function that we had created before looking at the histogram distributions. Since it had been more successful than expected, we retained it: for each antigen in the training set, generate a single cell mutated from the antigen whose distance from the antigen in feature space conforms to a hyperbolic (1/x) probability distribution with a constant multiplier which caused the distribution to tail off very rapidly. Consequently, most candidate memory cells were very close to their originating antigens.

Function 2 (Mod 2): The second function was also hyperbolic, with` a multiplier which allowed more of the distributional mass to be in a tail, rather like figure 2, but without the secondary hump.

Function 3 (Mod 3): The probability distribution was derived from the actual probability distribution of AIRS's memory cells during the solution of a representative problem. In particular, we created a function by smoothing and interpolating the probability distribution for the memory candidate cells produced by AIRS during a ten-way cross-validation training run on the Ionosphere problem. This distribution was almost identical in shape to Figure 1, except that a thin tail extended out to 0.45, affording an opportunity for some cells generated by the function to be somewhat further away from their respective training antigens.

RESULTS

Figures 4, 5, 6, and 7 show results for four of the data sets which were tested. Mod 0 refers to the revision of AIRS described in [19] (the version of AIRS which is being distributed as of this writing). Mods 1, 2, and 3 refer to the modifications of that algorithm that replace candidate memory cell generation with the three functions described in the preceding section. While AIRS can be run pretty much "as delivered", it does have a number of user-specifiable parameters which can be modified to suit a given data set. The term "Unoptimized" in the figures refers to running the algorithm using the default values for its parameters. The term "Optimized" refers to systematically changing those parameters and observing whether average performance improves. The test runs in all cases are averages over three runs of 10-way validation using the parameters suggested by the optimization suite of tests. A number of trends are clear from examination of the performances of these unoptimized and "optimized" modifications of AIRS.

AIRS run "as delivered" with default parameters tends to deliver accuracy only a few percentage points below its optimum performance.

We note that the scales on Figures 4 and 5 are somewhat exaggerated. For the Wisconsin Breast Cancer data, the functions are marginally less effective than the original algorithm in both unoptimized and optimized cases, with the exception of the optimized version of the modification which used a smoothed function derived from the original AIRS' own candidate memory cell generation distribution. For the iris data, the unoptimized functions were slightly more accurate than the unoptimized unmodified AIRS, but the optimized original AIRS was slightly more accurate than the optimized functions. Both of these trends were observed in the remaining four data sets not illustrated in the figures which follow.

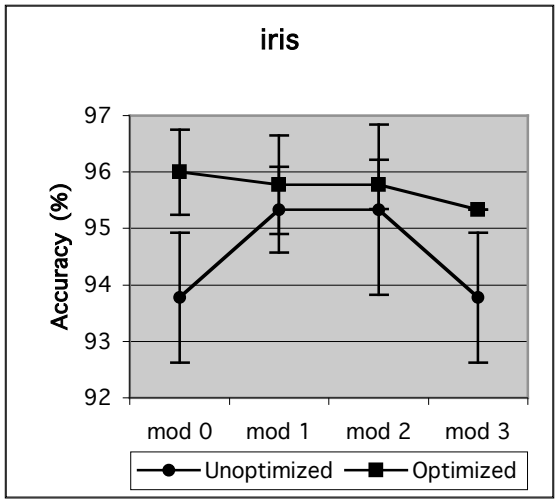

Figure 4.

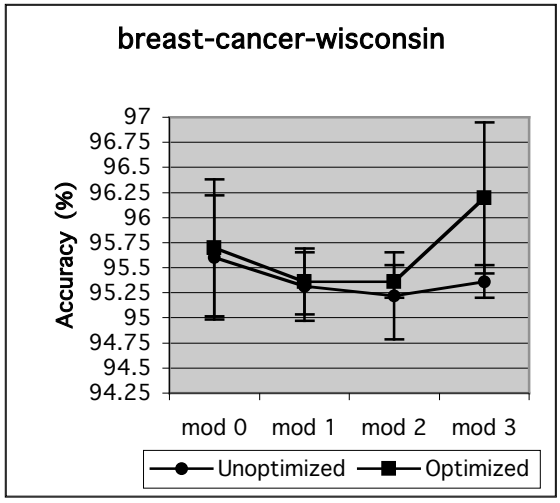

Figure 5.

It also tends to be the case that basing the functional distribution of the memory cell candidates on a prototypical distribution from AIRS's solutions to real-world problems is less effective than the other functional strategies; that is, the performance of mod 3 on the Iris data is more reflective of its general competitiveness than its performance on the Wisconsin Breast Cancer problem.

CONCLUSION

If we remove mod 3 from consideration, the differences between the functional modifications and normal AIRS, though observable, are not statistically significant. Hence we conclude that replacing the memory cell candidate generation portion of the AIRS algorithm has not changed the performance significantly. This leads us to hypothesize that the power of this classifier is in its approach to adding and replacing memory cells in the memory cell population. We are in the process of testing this conjecture.

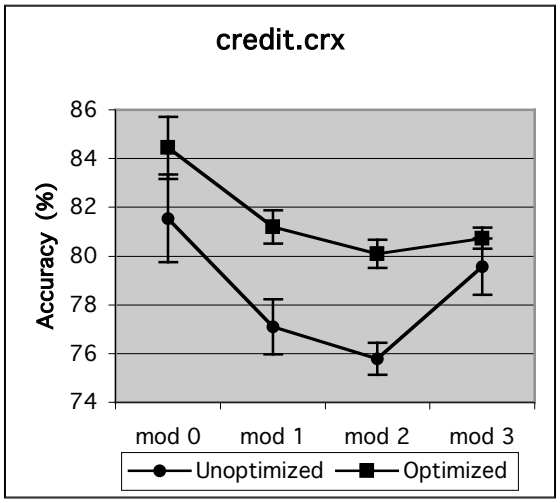

Figure 6.

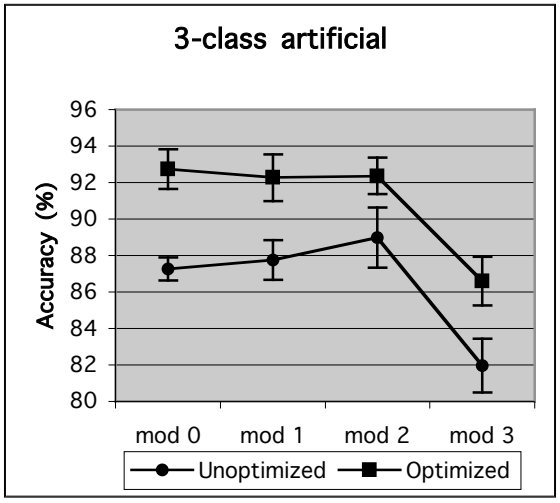

Figure 7. Comparison of the different modifications of AIRS on an artificially generated set of classes in the real plane R^2. The classes have convoluted boundaries and fill the subplane, and their instances are approximately uniformly distributed in the subspace.

We also note that the functional modifications to AIRS described in this paper are fast. In our experience they are not much faster than standard AIRS for most of the problems we have experimented with - one run of standard AIRS using five-way cross-validation takes less than five minutes on a 500MHz laptop for most of the datasets mentioned. However, we have encountered datasets for which certain combinations of user-set parameters caused AIRS to train for long periods before the population of B-cells reached the user-prescribed stimulation threshold, which allowed it to stop training. The function-based variations on AIRS are guaranteed not to have such problems, and do allow a user to get a reasonable estimate of how well AIRS works on the user's classification problem.

REFERENCES

[1] D. Goodman, L. Boggess, and A. Watkins, "Artificial Immune System classification of multiple-class problems," in *Proc. Artificial Neural Networks in Engineering (ANNIE-2002)*, St. Louis, Mo. Nov. 2002.

[2] L. N. de Castro and F. von Zuben, "An evolutionary immune network for data clustering," in *Proc. IEEE Brazilian Symposium on Neural Networks,* pp. 84-89, 2000.

[3] J. Timmis,, M. Neal, and J. E. Hunt, "An artificial immune system for data analysis," *Biosystems*, vol. 55, no. 1, pp. 143-150, 2000.

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

[5] A. Watkins and L. Boggess, "A new classifier based on resource limited Artificial Immune Systems," in *Proc. 2002 Congress on Evolutionary Computation (CEC2002),* IEEE Press, Honolulu, Hawaii, May 2002.

[6] A. Watkins and L. Boggess, "A resource limited Artificial Immune classifier.," in *Proc. 2002 Congress on Evolutionary Computation (CEC2002),* special session on Artificial Immune Systems. IEEE Press, Honolulu, Hawaii, May 2002.

[7] G. Marwah and L. C. Boggess, "Artificial Immune Systems for classification : Some issues," in *Proc. First Intl. Conf. Art. Immune Systems,* Canturbury, England, pp.149-153, Sep. 2002.

[8] S. Forrest, S. A. Hofmeyer, and A. Somayaji, "Computer immunology," *Communications of the ACM,* vol. 40, no. 10, pp.88-96, 1997.

[9] S. Forrest, A.S. Perelson, L. Allen, and R. Cherukuri, "Self-Nonself Discrimination in a Computer.". in *Proc. 1994 IEEE Symposium on Research in Security and Privacy*, Los Alamitos, CA: IEEE Computer Society Press, 1994. http://www.cs.unm.edu/~forrest/isa_papers.htm

[10] S. Hofmeyr, S. Forrest, and A. Somayaji, "Intrusion Detection Using Sequences of System Calls." *Journal of Computer Security* vol. 6, pp. 151-180, 1998. http://www.cs.unm.edu/~forrest/isa_papers.htm

[11] D. Dasgupta and S. Forrest, "Novelty detection in time series data using ideas from immunology," in *Proc. International Conference on Intelligent Systems,* 1999. http://www.cs.unm.edu/~forrest/isa_papers.htm

[12] J. Kim and P. Bentley, "Towards an Artificial Immune System for network intrusion detection: An investigation of dynamic clonal selection," in *Proc. 2002 Congress on Evolutionary Computation,* IEEE Press, Honolulu, pp. 1015-1020, May 2002.

[13] Jerome Carter, "The immune system as a model for pattern recognition and classification," *J American Medical Informatics Association (JAMIA),* vol. 7, no. 1, pp. 28-41, 2000.

[14] Jon Timmis, and Mark Neal, "Investigating the evolution and stability of a resource limited artificial immune system," in *Proc. Genetic and Evolutionary Computation Conference (GECCO)*, pp. 40-41, 2000.

[15] J. Timmis, *Artificial immune systems: A novel data analysis technique inspired by the immune network theory*, Ph.D. thesis, Department of Computer Science, University of Wales, Aberystwyth, Sep. 2000.

[16] J. Timmis and M. Neal, "A resource limited artificial immune system for data analysis," *Knowledge Based Systems,* vol. 14, nos. 3-4, pp. 121-130, 2001.

[17] Duch, W. 2000a. "Datasets used for classification: Comparison of results," http://www.phys.uni.torun.pl/kmk/projects/datasets.html.

[18] Duch, W. 2000b. "Logical rules extracted from data," http://www.phys.uni.torun.pl/kmk/projects/rules.html.

[19] A. Watkins and J. Timmis, "Artificial Immune Recognition System (AIRS): Revisions and refinements," in *Proc. 1st International Conf. on Artificial Immune Systems (ICARIS 2002)*, pp. 173-181, Sep. 2002.

[20] C. L. Blake, & C. J. Merz, *UCI repository of machine learning databases*, 1998. Retrieved 22 April 2002, from the University of California, Irvine Computer Science website: http://www.ics.uci.edu/~mlearn/MLRepository.html.