MILA – Multilevel Immune Learning Algorithm

Dipankar Dasgupta, Senhua Yu, and Nivedita Sumi Majumdar

Computer Science Division, University of Memphis, Memphis, TN 38152, USA
{dasgupta, senhuayu, nmajumdr}@memphis.edu

Abstract. The biological immune system is an intricate network of specialized tissues, organs, cells, and chemical molecules. T-cell-dependent humoral immune response is one of the complex immunological events, involving interaction of B cells with antigens (Ag) and their proliferation, differentiation and subsequent secretion of antibodies (Ab). Inspired by these immunological principles, we proposed a Multilevel Immune Learning Algorithm (MILA) for novel pattern recognition. It incorporates multiple detection schema, clonal expansion and dynamic detector generation mechanisms in a single framework. Different test problems are studied and experimented with MILA for performance evaluation. Preliminary results show that MILA is flexible and efficient in detecting anomalies and novelties in data patterns.

1 Introduction

The biological immune system is of great interest to computer scientists and engineers because it provides a unique and fascinating computational paradigm for solving complex problems. There exist different computational models inspired by the immune system. A brief survey of some of these models may be found elsewhere [1]. Forrest et al. [2–4] developed a *negative-selection algorithm* (NSA) for change detection based on the principles of self-nonself discrimination. This algorithm works on similar principles, generating detectors randomly, and eliminating the ones that detect self, so that the remaining detectors can detect any non-self. If any detector is ever matched, a change (non-self) is known to have occurred. Obviously, the first phase is analogous to the censoring process of T cells maturation in the immune system. However, the monitoring phase is logically (not biologically) derivable.

The biological immune system employs a multilevel defense against invaders through nonspecific (innate) and specific (adaptive) immunity. The problems for anomaly detection also need multiple detection mechanisms to obtain a very high detection rate with a very low false alarm rate. The major limitation of binary NSA is that it generates a higher false alarm rate when applied to anomaly detection for some data sets. To illustrate this limitation, some patterns, for example, 110, 100, 011, 001, are considered as normal samples. Based on these normal samples, 101, 111, 000, 010 become abnormal. A partial matching rule is usually used to generate a set of detectors. As described in [5], with matching threshold ($r = 2$), two strings (one represents

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2723, pp. 183–194, 2003.

candidate detector, another is a pattern) match if and only if they are identical in at least 2 contiguous positions. Because the detector must fail to match any string in normal samples, for the above example, the detectors cannot be generated at all, and consequently anomalies cannot be detected; except for r = 3 (length of the string), which results in exact match and requires all non-self strings as detectors.

In order to alleviate these difficulties, we proposed an approach, called *Multilevel Immune Learning Algorithm (MILA)*. There are several features which distinguish this algorithm from the NSA; in particular, multilevel detection and immune memory. In this paper, we describe this approach and show the advantages of using new features of *MILA* in the application of anomaly detection.

The layout of this paper is as follows. Section 2 outlines the proposed algorithm. Section 3 briefly describes the application of MILA to anomaly detection. Section 4 reports some experimental results with different testing problems. Section 5 discusses new features of *MILA* indicated in the application of anomaly detection. Section 6 provides concluding remarks.

2 Multilevel Immune Learning Algorithm (MILA)

This approach is inspired by the interaction and processes of T cell-dependent humoral immune response. In biological immune systems, some B cells recognize antigens (foreign protein) via immunoglobulin receptors on their surface but are unable to proliferate and differentiate unless prompted by the action of lymphokines secreted by T helper cells. Moreover, in order for T helper cells to become stimulated to release lymphokines, they must also recognize specific antigens. However, while T helper cells recognize antigens via their receptors, they can only do so in the context of MHC molecules. Antigenic peptides must be extracted by several types of cells called antigen-presenting cells (APCs) through a process called “Ag presentation.” Under certain conditions, however, B-cell activation is suppressed by T suppressor cells, but specific mechanisms for such suppression are yet unknown. The activated B cells and T cells migrate to the primary follicle of the cortex in lymph nodes, where a complex interaction of the basic cell kinetic process of proliferation (cloning), mutation, selection, differentiation, and death of B-cells occurs through germinal center reaction [6] and finally secretes antibodies. These antibodies function as effectors to the humoral response by binding to antigens and facilitating their elimination. The proposed artificial immune system is an abstract of complex multistage immunological events in humoral immune response. The algorithm consists of initialization phase, recognition phase, evolutionary phase and response phase. As shown in Fig.2, the main features of each phase can be summarized as follows:

- In *initialization* phase, the detection system is “trained” by giving the knowledge of “self”. The outcome of the initialization is to generate sets of detectors, analogous to the populations of T helper cells (T_h), T suppressor cells (T_s) and B cells, which participate in T cell dependent humoral immune response.

- In *recognition* phase, B cells, together with T cells (T_h, T_s) and antigen presenting cells (APCs), form a multilevel recognition. APC is an extreme high-level detector, which acts as a default detector (based on environment) identifying visible damage signals from the system. For example, while monitoring a computer system, screen turning black, too many lining-up printing jobs and so on may provide visible signals captured by APC. Thus, APC is not defined based on particular normal behavior in input data. It is to be noted that T cells and B cells recognize antigens at different levels. The recognition of T_h is defined as a bit-level (lowest level) recognition, such as using consecutive windows of data pattern. Importantly, B cells in the immune system only recognize particular sites called *epitope* on the surface of the antigen, as shown in Fig.1. Clearly, the recognition (matching) sites are not contiguous when we stretch out the 3-dimension folding of the antigen protein. Thus, the B cell is considered as feature-level recognition at different non-contiguous (occasionally contiguous) positions of antigen strings. Accordingly, MILA can provide multilevel detection in hierarchical fashion, starting with APC detection, B-cell detection and T-cell detection. However, T_s acts as suppression and is problem dependent. As shown in fig. 2, the logical operator can be set to $\wedge$ (AND) or $\vee$ (OR) to make the system more fault-tolerant or more sensitive as desired.
- In *evolutionary* phase, the activated B cells clone to produce memory cells and plasma cells. Cloning is subject to very high mutation rates called somatic hypermutation with a selective pressure. In addition to passing *negative selection*, for each progeny of the activated B cell (parent B cell), only the clones with higher affinity are selected. This process is known as *positive selection*. The outcome of evolutionary phase is to generate high-quality detectors with specificity to the exposed antigens for future use.
- *Response* phase involves primary response to initial exposure and secondary response to the second encounter.

Accordingly, the above mechanism steps, as shown in the Fig.2, give a general description of MILA, however, based on applications and timeliness of execution, some detection phase may not be considered.

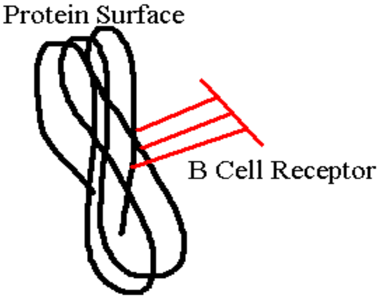


Fig. 1. B Cell Receptor Matches an antigenic protein in its surface

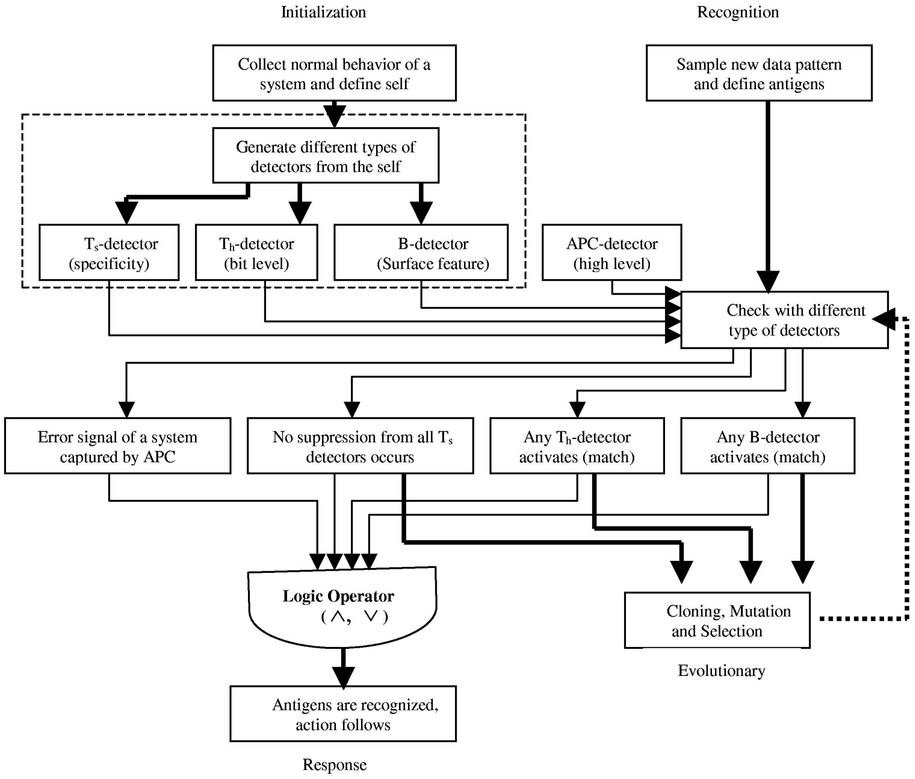

Initialization
Recognition
Collect normal behavior of a system and define self
Sample new data pattern and define antigens
Generate different types of detectors from the self
T$_s$-detector (specificity)
T$_h$-detector (bit level)
B-detector (Surface feature)
APC-detector (high level)
Check with different type of detectors
Error signal of a system captured by APC
No suppression from all T$_s$ detectors occurs
Any T$_h$-detector activates (match)
Any B-detector activates (match)
Logic Operator (∧, ∨)
Cloning, Mutation and Selection
Evolutionary
Antigens are recognized, action follows
Response

Fig. 2. Overview of Multilevel Immune Learning Algorithm (MILA)

3 Application of MILA to Anomaly Detection Problems

Detecting anomaly in a system or in a process behavior is very important in many real-world applications. For example, high-speed milling processes require continuous monitoring to assure high quality production; jet engines also require continuous monitoring to assure safe operation. It is essential to detect the occurrence of unnatural events as quickly as possible before any significant performance degradation results [5]. There are many techniques for anomaly detection, and depending on application domains, these are referred to as novelty detection, faulty detection, surprise pattern detection, etc. Among these approaches, the detection algorithm with better discrimination ability will have a higher detection rate. In particular, it can accurately discriminate the normal data and the observed data during monitoring. The decision-making systems for detection usually depend on learning the behavior of the monitored environment from a set of *normal* (*positive*) data. By normal, we mean usage data that have been collected during the normal operation of the system or a process. In order to evaluate the performance, *MILA* is applied to the anomaly detection problem. For this problem, the following assumptions are made to simplify the implementation:

- In *Initialization* phase and *Recognition* phase, T_s detectors employ more stringent threshold than T_h detectors and B detectors.
- T_s detector is regarded as a special self-detecting agent. In *Initialization* phase, T_s detector will be selected if it still matches the self-antigen under more stringent threshold, whereas in *Recognition* phase the response will be terminated when T_s detector matches a special antigen resembling self-data pattern. Similar to T_h and B cells, the activated T_s detector undergoes cloning and positive selection after being activated by a special Ag.
- APC-detectors, as shown in Fig.2, are not used in this application.
- The lower the antigenic affinity, the higher the mutation rate. From a computational perspective, the purpose of this assumption is to increase the probability of producing effective detectors.
- For each parent cloning, only ONE clone whose affinity is the highest among all clones is kept. The selected clone will be discarded if it is similar to the existing detectors. This assumption solves the problem using minimal resources without compromising the detection rate.
- Currently, the response phase is dummy as we are only dealing with anomaly detection tasks.

This application employs a distance measure (Euclidean distance) to calculate the affinity between the detector and the self/nonself data pattern along with a partial matching rule. Overall, the implementation of MILA for anomaly detection can be summarized as follows:

1. Collect *Self* data sufficient to exhibit the normal behavior of a system and choose a technique to normalize the raw data.
2. Generate different types of detectors, e.g., B, T_h, T_s detectors. T_h and B detectors should not match any of self-peptide strings according to the partial matching rule. The sliding window scheme [5] is used for T_h partial matching. The random position pick-up scheme is used for B partial matching. For example, suppose that a self string is $<s_1, s_2, \ldots, s_L>$ and the window size is chosen as 3, then the self peptide strings can be $<s_1, s_3, s_L>$, $< s_2, s_4, s_9 >$, $< s_5, s_7, s_8 >$ and so on by randomly picking up the attribute at some positions. If the candidate B detector represented as $<m_1, m_2, m_3 >$ fails to match *Any* self-feature indexed as $<1, 3, L_1>$ in self-data patterns, the candidate B detector is selected and represented as $<(1, m_1), (3, m_2), (L_1, m_3)>$. Two important parameters, *T_h threshold* and *B threshold*, are employed to measure the matching. If the value for the distance between the T_h (or B) detector and the self string is greater than T_h (or B) threshold, then it is considered as *matching*. T_s detector, however, is selected if it can match the special self strings by employing more stringent suppressor threshold called *T_s threshold.*

3. When monitoring the system, the logical operator shown in Fig.1 is chosen as "AND ($\wedge$)" in this application. The unseen pattern is tested by T_h, T_s, B detector, respectively. If any T_h and B detector is ever activated (matched with current pattern) and all of the T_s detectors are not activated, a change in behavior pattern is known to have occurred and an alarm signal is generated indicating an abnormality. The same matching rule is adopted as used in generating detectors. We calculate the distance between the T_h / T_s detector and the new sample as described in [5]. B detector is actually an information vector with the information of binding sites and values of attributes in these sites. For the B detector <(1, m_1), (3, m_2), (L, m_3)> in the above example, if an Ag is represented as <n_1, n_2, n_3, …, n_L>, then the distance is calculated only between points <m_1, m_2, m_3> and < n_1, n_3, n_L >.
4. Activated T_h, T_s, B detectors are cloned with a high mutation rate and only one clone with the highest affinity is selected. Detectors that are not activated are kept in detector sets.
5. Employ the optimized detectors generated after the detection phase to test the unseen patterns, repeat from step 3.

4 Experiments

4.1 Data Sets

We experimented with different datasets to investigate the performances of *MILA* for detecting anomalous patterns. The paper only reported results of using speech-recording time series dataset (see reference [8]) because of space limitations. We normalized the raw data (total 1025 time steps) at the range 0~1 for training the system. The testing data (total 1025 time steps) are generated that contain anomalies between 500 and 700 and some noise after 700 time steps.

4.2 Performance Measures

Using a sliding (overlapping) window of size L (in our case, L =13), if normal series have the values: x_1, x_2, …, x_m, *self-patterns* are generated as follows:

$$<x_1, \quad x_2, \quad \ldots \quad x_L>$$
$$<x_2, \quad x_3, \quad \ldots \quad x_{L+1}>$$
$$. \quad . \quad . \quad .$$
$$<x_{m-L+1}, \; x_{m-L+2}, \ldots, \; x_m>$$

Similarly, *Ag-patterns* are generated from the samples shown in Fig.4b. In this experiment, we used real-valued strings to represent Ag and Ab molecules, which is different from binary Negative Selection Algorithm [4, 5, 9] and Clone Selection Principle application [10]. Euclidean distance measure is used to model the complex chemistry of Ag/Ab recognition as a matching rule. Two measures of effectiveness for detecting anomaly are calculated as follows:

$$\text{Detection rate } = \frac{TP}{TP+FN}$$

$$\text{False alarm rate} = \frac{FP}{TN+FP}$$

Where *TP* (true positives), anomalous elements identified as anomalous; *TN* (true negatives), normal elements identified as normal; *FP* (false positives), normal elements identified as anomalous; *FN* (false negatives), anomalous elements identified as the normal [11].

The MILA algorithm has a number of tuning parameters. Different detector thresholds that determine whether a new sample is normal or abnormal control the sensitivity of the system. Employing various strategies to change threshold values, different values for detection rate and false alarm rate are obtained that are used for plotting the ROC (Receiver Operating Characteristics) curve, which reflects tradeoff between false alarm rate and detection rate.

4.3 Experimental Results

The following test cases are studied and some results are reported in this paper:

1. For different threshold changing strategies the influence on ROC curves is studied. In this paper, we report the results obtained from three different cases: (1) changing B threshold at fixed T_h threshold (0.05 if B threshold is less than 0.16, otherwise 0.08) and Ts threshold (0.02); (2) changing B threshold at fixed T_h threshold (0.1) and T_s threshold (0.02); (3) changing T_h threshold at fixed B threshold (0.1) and T_s threshold (0.02). The results shown in Fig.3 indicate that the first case obtain a better ROC curve. Therefore, this paper uses this strategy to obtain different values for detection and false alarm rate for *MILA* based anomaly detection.
2. The comparison of performances illustrated in ROC curves between single level detection and multilevel detection (*MILA*) is studied. We experimented and compared the efficiency of anomaly detection in three cases: (1) only using T_h detectors; (2) only using B detectors; (3) combining T_h, T_s, B detectors as indicated in *MILA*. ROC curves in these cases are shown in Fig.4. Moreover, Fig.5 show how detection and false alarm rates change when threshold is modified in these three cases. Since detectors are randomly generated, different values for detection and false alarm rates are observed. Considering this issue, we run the system ten iterations to obtain the average of the values for detection and false alarm rate, as shown in Fig.4 and Fig.5.

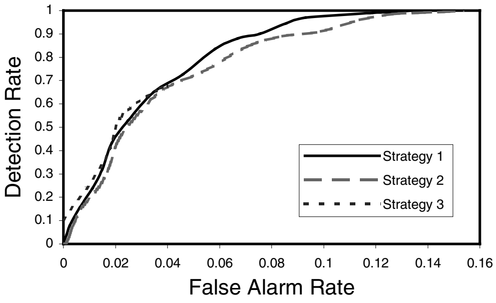


Fig. 3. ROC curves obtained by employing different thresholds changing strategy as described in the section 4.3

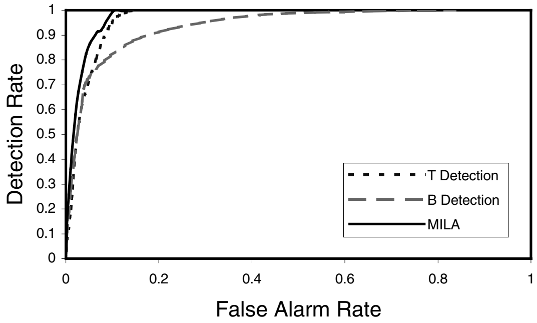
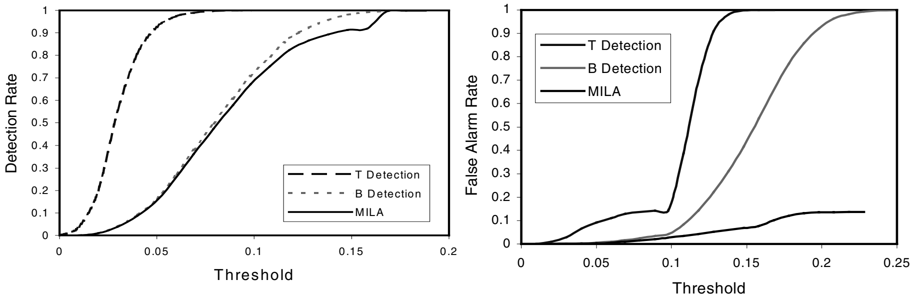


Fig. 4. Comparison of ROC curves between single level detection (e.g., T_h detection or B detection) and multilevel detection (*MILA*)



(a) (b)

Fig. 5. Evolution of detection rate in Fig. 5(a) and false alarm rate in Fig. 5(b) based on single level detection and multilevel detection (*MILA*) with changing threshold values

3. The efficiency of the detector for detecting anomaly is studied. Once detectors, e.g., T_h detectors, T_s detectors and B detects, are generated in *Initialization* phase, we repeatedly tested the same abnormal samples for 5 iterations

with same parameter settings. Since the detector in MILA undergoes cloning, mutation and selection after Recognition phase, the elements in the detector set changes after each iteration in detecting phase, although the same abnormal samples and conditions are employed in Recognition phase. So, for each iteration, different values for detection and false alarm rate are observed, as shown in Fig. 6 through Fig.7.

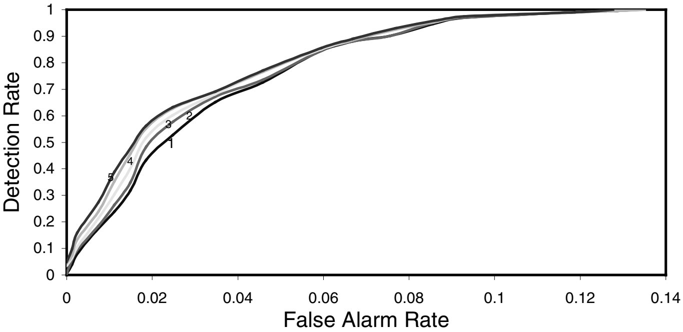


Fig. 6. ROC curves for MILA based anomaly detection in each detecting iteration. 1, 2, 3, …in the ROC curves denote the iterations of detecting same Ag samples. For each iteration, the detector sets are those that are generated in the detect phase of previous iteration.

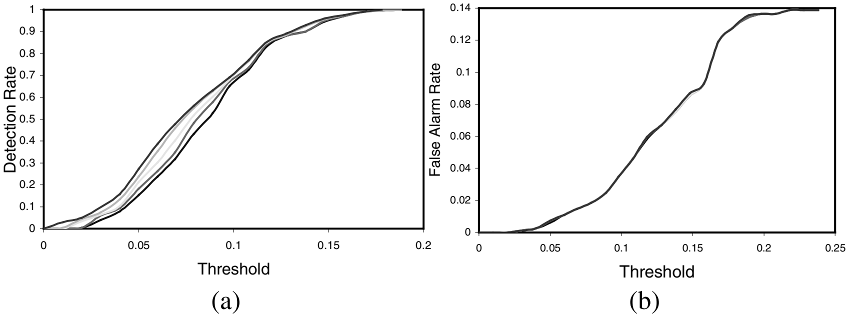


Fig. 7. Evolution of detection rate in Fig. 7(a) and false alarm rate in Fig. 7(b) for *MILA* based anomaly detection in each detecting iteration as described in Fig.6 when threshold is varied.

5 New Features of MILA

The algorithm presented here takes its inspiration from T-cell-dependent humoral immune response. Considering the application to anomaly detection, one of the key features of *MILA* is its multilevel detection; that is, multiple strategies are used to generate detectors, which are combined to detect anomalies in new samples.

Preliminary experiments show that *MILA* is flexible and unique. The generation and recognition of various detectors in this algorithm can be implemented in different ways depending on the application. Moreover, the efficiency of anomaly detection can be improved by tuning threshold values for different detection scheme. Fig.3 shows this advantage of *MILA* and indicates that better performance can be obtained (shown in the ROC curves) by employing different threshold changing strategies. Compared to the Negative Selection Algorithm (*NSA*), which uses single level detection scheme, Fig.4 shows that the performance of multilevel detection of MILA is better. Further results shown in Fig.5 also support the superior performance of MILA. Specifically, when comparing multilevel detection (MILA) with single detection scheme (NSA), the varying trend for detection rate when the threshold is modified is similar as illustrated in Fig.5(a) (at least relative to false alarm rate as shown in Fig.5(b); however, the false alarm rate for multilevel detection (when the threshold is modified) is much lower.

For anomaly detection using *NSA*, the detector set remains constant once generated in the training phase. However, the detector set is dynamic for *MILA* based anomaly detection. *MILA* involves a process of cloning, mutation and selection after successful detection, and some detectors with high affinity for a given anomalous pattern will be selected. This constitutes an on-line learning and detector optimization process. The outcome is to update the detector set and affinity of those detectors that have proven themselves to be valuable by having recognized more frequently occurring anomalies. Fig.6 shows improved performance by using the optimized detector set being generated after the detection phase. This can be explained by the fact that some of the anomalous data employed in our experiment are similar, but generally anomaly is much different from normal series. Thus, when we reduce the distance between a detector and a given abnormal pattern, that is, increase the detector affinity for this pattern, the distances between this detector and other anomalies similar to the given abnormal pattern are also reduced so that those anomalies which formerly failed to be detected by this detector become detectable. However, the distances between the detector and most of the “self”, except for some “self” very similar to “non-self” (anomaly), are still exceeds allowable variation. Therefore, the number of detectors having high affinity increases with the increase in the times of detecting the antigens that are encountered before (at least in a certain range) and thus the detection rate at certain thresholds becomes higher and higher. The experimental results confirm this explanation. Under the same threshold values, Fig.7(a) shows that the detector set produced later has a higher detection rate than the previous detector set, whereas the false alarm rate is almost unchanged as shown in Fig.7(b). In the application to anomaly detection, because of the random generation of the pre-detector, the generated detector set is always different, even if the absolutely same conditions are applied. We cannot guarantee the efficiency of the initial detector set. However, *MILA* based anomaly detection can optimize the detector during on-line detection and thus we can finally obtain more efficient detectors for given samples for monitoring.

As a summary of our proposed principle and initial experiments, the following features of MILA have been observed on anomaly detection:

- Unites several different immune system metaphors rather than implementing the immune system metaphors in a piecemeal manner.

- Uses multilevel detection to find and patch the security hole in a large computer system as much as possible.
- MILA is more flexible than single detection scheme (e.g. Negative Selection Algorithm). The implementation for detector generation is problem dependent. More thresholds and parameters may be modified for tuning the system performance.
- Detector set in MILA is dynamic whereas detector set in Negative Selection Algorithm remains constant once it is generated in training phase. MILA involves cloning, mutation and selection after detect phase, which is similar but not equal to Clone Selection Theory. The process of cloning in MILA is targeted (not blind) cloning. Only those detectors that are activated in recognition phase can be cloned. The process of cloning, mutation and selection in MILA is actually a process of detector on-line learning and optimization. Only those clones with high affinity can be selected. This strategy ensures that both the speed and accuracy of detection become successively higher after each detecting.
- MILA is initially inspired by humoral immune response but spontaneously unites the main feature of Negative Selection Algorithm and Clone Selection Theory. It imports their merits but has its own features

6 Conclusions

In this paper, we outlined a proposed change detection algorithm inspired by the T-cell-dependent humoral immune response. This algorithm is called Multilevel Immune Learning Algorithm (*MILA*), which involves four phases: Initialization phase, Recognition phase, Evolutionary phase and Response phase. The proposed method is tested with an anomaly detection problem. *MILA* based anomaly detection is characterized by multilevel detection and on-line learning technique. Experimental results show that *MILA* based anomaly detection is flexible and the detection rate can be improved at the range of allowable false alarm rate by applying different threshold changing strategies. In comparison with single level based anomaly detection, the performance of MILA is clearly better. Experimental results show that detectors have been optimized during the on-line testing phase as well. Moreover, by busing different logical operators, it is possible to make the system very sensitive to any changes or robust to noise. Reducing complexity of the algorithm, proposing appropriate suppression mechanism, implementing response phase and experimenting with different data sets are the main directions of our future work.

Acknowledgement. This work is supported by the Defense Advanced Research Projects Agency (no. F30602-00-2-0514). The authors would like to thank the source of the datasets: Keogh, E. & Folias, T. (2002). The UCR Time Series Data Mining Archive [*http://www.cs.ucr.edu/~eamonn/TSDMA/index.html*]. Riverside CA. University of California – Computer Science & Engineering Department.

References

1. Dasgupta, D., Attoh-Okine, N.: Immunity-Based Systems: A Survey. In the proceedings of the *IEEE International Conference on Systems, Man and Cybernetics*, Orlando, October 12–15, 1997
2. Forrest, S., Hofmeyr, S., Somayaji, A.: Computer Immunology. Communications of the ACM **40**(10) (1997) pp 88–96.
3. Forrest, S., Somayaji, A., Ackley, D.: Building Diverse Computer Systems. Proc. of the Sixth Workshop on Hot Topics in Operating Systems (1997).
4. Forrest, S., Perelson, A. S., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. Proc. of the IEEE Symposium on Research in Security and Privacy, IEEE Computer Society Press, Los Alamitos, CA, (1994) 202–212
5. Dasgupta, D., Forrest, S.: An Anomaly Detection Algorithm Inspired by the Immune System. In: Dasgupta D (eds) Artificial Immune Systems and Their Applications, Springer-Verlag, (1999) 262–277
6. Hollowood, K., Goodlad, J.R.: Germinal centre cell kinetics. J. Pathol.**185**(3) (1998) 229–33
7. Perelson, A. S., Oster, G. F.: Theoretical studies of clonal selection: Minimal antibody repertoire size and reliability of self- non-self discrimination. J. Theor.Biol. **81**(4) (1979) 645–670
8. Keogh, E., Folias, T.: The UCR Time Series Data Mining Archive [*http://www.cs.ucr.edu/~eamonn/TSDMA/index.html*]. Riverside CA. University of California – Computer Science & Engineering Department. (2002)
9. D'haeseleer, P., Forrest, S., Helman, P.: An immunological approach to change detection: algorithms, analysis, and implications. Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy, IEEE Computer Society Press, Los Alamitos, CA, (1996) 110–119
10. de Castro, L. N., Von Zuben, F. J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation **6**(3) (2002) 239–251
11. Gonzalez, F., Dasgupta, D.: Neuro-Immune and SOM-Based Approaches: A Comparison. Proceedings of 1st International Conference on Artificial Immune Systems (ICARIS-2002), University of Kent at Canterbury, UK, September 9th–11th, 2002