Anomaly Detection Inspired by Immune Network Theory: A Proposal

HuiKeng Lau, Jon Timmis, Iain Bate

Abstract— **Previous research in supervised and unsupervised anomaly detection normally employ a static model of normal behaviour (*normal-model*) throughout the lifetime of the system. However, there are real world applications such as swarm robotics and wireless sensor networks where what is perceived as normal behaviour changes accordingly to the changes in the environment. To cater for such systems, dynamically updating the *normal-model* is required. In this paper, we examine the requirements from a range of distributed autonomous systems and then propose a novel unsupervised anomaly detection architecture capable of online adaptation inspired by the vertebrate immune system.**

I. Introduction

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analysing them for instances which violate (or possibly violate) related security policies or practices [1]. Based on intrusion detection methods, intrusion detection system (IDS) can be categorised into misuse detection and anomaly detection. Misuse detection detects attacks based on known attack signatures. Although this method could achieve very high accuracy, it is unlikely to detect novel or unknown attacks. For this reason, anomaly detection which has the potential to detect unknown attacks and variations of known attacks was proposed.

Ever since its introduction by Denning [2] in 1987, anomaly detection has been an active field with many new approaches proposed by researchers. Anomaly detection approaches build models of normal data and attempt to detect deviations from the *normal-model* in data. The generation of the model can be categorised into supervised and unsupervised approaches. In the supervised approach, the model is generated based on a dataset with purely normal instances or with labelled anomalous instances [3]. However, there are drawbacks with such an approach [refer to section III-A]. In the unsupervised approach, raw data is used to generate the model with clustering techniques being typically employed for this task [4]. At the end of clustering, the clusters with instances less than predefined threshold are considered anomalous. After training (either supervised or unsupervised), the derived model is use and remain static throughout the lifetime of the system. If the monitored system operates within a static environment, having a static *normal-model* would be sufficient. However, in the real world environment, there are systems where the changes in the environment affect the model and the model needs to be updated dynamically to reflect those changes and to avoid false alarms. There are some early works on maintaining a dynamic *normal-model* but the initial model generation is still supervised requiring data that might not be readily available.

In this paper, we propose an unsupervised anomaly detection architecture with adaptive model maintenance that is inspired by the immune network theory proposed by Jerne [5]. This is motivated by the analogy between immune system and distributed autonomous systems and also the learning and tolerance ability of the immune network. The remainder of this paper is structured as follows. Section II briefly introduces two examples of distributed autonomous systems namely wireless sensor networks (WSN) and swarm robotics and outline the requirements for anomaly detection in such systems. Then, in section III, a brief description on the architectures of current anomaly detection systems (ADS) is presented. In section IV, we present a brief outline of the immunology relevant to our work, with a focus on the immune network theory. Section V is the ADS architecture we are proposing and section VI conclude this paper and point out our future work.

II. Distributed Autonomous Systems

In recent years, technological advances allow the development of small, low-cost and low-power devices that are capable of sensing phenomena in the physical world [6]. Such devices can be connected together to form a network for applications beyond traditional boundaries, offer numerous advantages such as large scale and wide geographical coverage, high quality of sensed data, and deployment for continuous monitoring in difficult and inaccessible terrains. WSN and swarm robotics are examples of two application domains greatly influenced by these advances and the implementation of these systems becomes possible due to the cost reduction in such devices. Both systems fall under the category of distributed autonomous system. Segel [7] defined a distributed autonomous system as a system that is composed of many entities (agents) and that activities of the system are accomplished by the combined action of many of these entities without centralised control. Each entity interacts with its environment (which consists of other entities) but acts with some degree of autonomy [8].

Distributed autonomous systems promise a wide range of new application areas including applications that requires minimal human intervention and deployment in difficult and

H.K Lau and I. Bate are with the Department of Computer Science, University of York, Heslington, York, YO10 5DD, UK (email: {hklau, iain.bate}@cs.york.ac.uk).

J. Timmis is with the Department of Computer Science and Department of Electronics, University of York, Heslington, York, YO10 5DD, UK (email: jtimmis@cs.york.ac.uk).

978-1-4244-2959-2/09/$25.00 ©

inaccessible terrain. For example, WSN have been applied in variety of applications including military applications, environmental and habitat monitoring, health care, smart homes and agriculture while swarm robotics could be applied in application with collective task that cannot be solved by a single entity.

To clarify the motivation of our work, we will briefly introduce two examples of distributed autonomous system namely WSN and swarm robotics with focus on WSN and the type of WSN applications where our ADS could be very useful.

A. Wireless Sensor Networks

A wireless sensor network is normally composed of a large number of low-power, low-cost and mostly static (can be mobile) sensor nodes that are usually densely deployed to monitor certain phenomenon of interest [6], [9]. A sensor node is normally made up of a sensing unit, a processing unit, a transceiver and a power unit [6]. The sensors in a sensor network collect measurement data from their respective environments, process it and then transmit it through a communication network to a base station(s) or sink node(s). A base station can be a gateway to another network, a more powerful data processing or storage centre, or an access point to a human interface [9].

Sensor networks are different from the traditional computer networks and ad-hoc networks. In addition to being more constrained in terms of resources, WSN are more prone to failure; often due to deliberate attack or malfunctions in their deployed environment, (frequent) change of topology with the addition, removal or failure of sensor nodes, (often) with no global identification and more specialised communication patterns [6], [9]. Sensor networks are considered to be complex systems with the ability to automatically and dynamically self-organising (due to the communication protocol) its connectivity [10]. Sensor nodes may fail and new sensor nodes may be added. The self-organising ability allows the network topology to be dynamically reconfigured to reflect those changes.

The challenges in designing anomaly detection system for sensor networks lie with the self-organising nature of sensor networks, resource constraints (typically 2 AA batteries, 4K-64K RAM), variation of anomalies (anomalies in data collected by sensors, anomalies in physical displacement of sensor nodes and anomalies in communication patterns due to attacks), and (for most applications) changing of normal patterns due to changing environment and resources. For example, consider a sensor network deployed to detect forest fire. In this case, the thermal sensor on the sensor node will collect temperature data at regular interval. For different seasons, the expected temperature will be different e.g. higher in summer, lower in winter. If the *normal-model* for temperature (in degree Celsius) in the ADS is set to be within the highest and lowest temperature expected: [*lowest_winter*, *highest_summer*] such as *lowest_winter* = -5 and *highest_summer* = 35, the ADS would not detect the temperature of -1 degree Celsius in summer as anomaly (when it is obviously is). Similarly, a temperature of 20 degree Celsius in winter would not be detected as anomaly. Thus, an ADS that would adaptively changes the *normal-model* according to the seasons is preferable. This change of normal behaviour at different time is described by the notion of time bands [11].

Sensor networks are in essence a type of real-time system, thus the response time is also important. The nature of such sensor networks necessitate that ADS developed must fulfil the requirements of

- **Accuracy**
  The ADS must be able to detect the anomalies with considerable accuracy. Measurement of accuracy can be true positive, false positive or probability of detection.
- **Responsiveness**
  The detection and responses to anomalies must be within acceptable time frame.
- **Resource usage**
  With limited resources, the ADS must be lightweight consuming minimal computing and battery
- **Robustness**
  The ADS must be able to adapt to a changing environment such as the decrease in battery power. With low battery power, an increase in the ratio of error packets, frequency of route discovery and updates are expected. In addition, as resources become low then a degraded level of service may become acceptable. Inversely, those increases when battery power is still acceptable should not be tolerated.

B. Swarm Robotics

Similar to WSN, swarm robotics involve the coordination of a large number of simple, (mostly) homogeneous robots to achieve some collective tasks that cannot be achieved by a single robot. It was inspired from the system-level functioning of social insects where a desired collective behavior emerges from the interactions between the insects and interactions with the environment [12]. Anomalies in swarm robotics can be categorised into anomalies within a single robot (internal faults) and anomalies that affect the behaviour of the swarm (system-level faults).

There are many works on internal faults detection as reviewed in [13]. However to the best of our knowledge, no work has been done on detection of system-level faults. At the system-level, a faulty robot might not be able to participate and complete the tasks it is assigned to and thus forces other robots to adapt accordingly to complete the task. If a swarm robotic system is assigned a task where similar behaviour is expected for all robots within a local proximity (such as swarm foraging), a robot that behaves differently that all other robots may be faulty. Therefore, other participating (observing) robots within the same proximity should detect such changes.

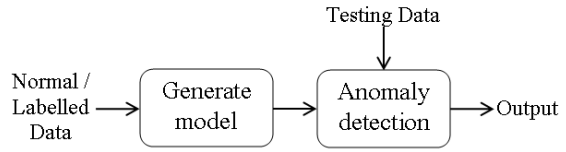
Anomaly detection in such systems exhibit similar characteristics as WSN. Requirements such as responsiveness might

not be as stringent as WSN but other requirements remain the same.

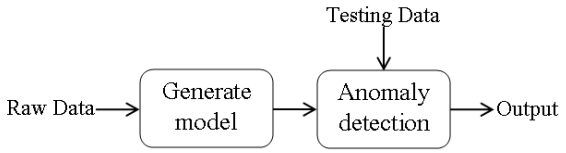
III. Anomaly detection systems

A. Supervised vs unsupervised model generation

Depending on what type of data is presented to the ADS for *normal-model* generation, ADS can be categorised into supervised and unsupervised learning. In supervised anomaly detection, a training dataset which is purely normal or with correctly labelled attacks is used to train and generate the model of normal behaviour [3] (Figure 1(a)). Many supervised ADS techniques were proposed over the years; neural network was used in [14], Hidden Markov Model in [15], [16], [17], and junction tree algorithm in [1]. These techniques have showed very impressive results on benchmark datasets. However, for real world applications, it is very difficult and very seldom to have purely normal data or perfectly labelled data. If the data instances of intrusions are not identified or correctly identified, the training algorithm may not be able to detect future instances of these attacks [3]. Furthermore, it is tedious and impractical to manually classify and label the enormous amount of audit data available [3], [18]. To address this issue, unsupervised anomaly detection was proposed to generate the model using raw and unlabelled data.



(a) Supervised model generation



(b) Unsupervised model generation

Fig. 1. Model generation in anomaly detection system.

An unsupervised anomaly detection algorithm takes a set of unlabelled data as inputs and attempts to find intrusions buried within the data (Figure 1(b)). It is similar to the classical outlier detection problem [3]. Hence, clustering is typically used for unsupervised model generation. In [3], the authors presented cluster-based estimation, K-nearest neighbour (KNN) and one class Support Vector Machine (SVM) while [4] presented a slightly different immune-inspired approach. For unsupervised ADS, two general assumptions are made [18], [4].

- The number of normal instances outnumber intrusions
- Intrusions are quantitatively different from normal instances

Essentially what all these algorithms do is to cluster all similar data together. At the end of the clustering, clusters with less than certain percentage (10% in [4], 5% in [19]) of data would be consider as anomalous (thus the assumptions). The rest of the clusters would serve as the model of normal behaviour. Experimenting with benchmark dataset such as the 1999 KDD Cup dataset [20], these algorithms reported very good results.

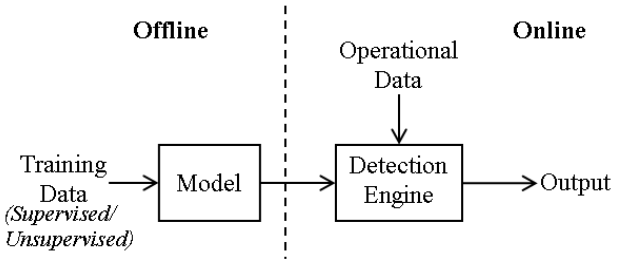
Both supervised and unsupervised anomaly detection with a static *normal model* will perform well in applications where the environmental changes would have little or no effect on the model of normal behaviour. However, for applications of some distributed autonomous systems such as WSN or swarm robotics where the model of normal behaviour is affected by spatial and temporal conditions, the model needs to be updated dynamically to avoid false alarms.

B. Static vs dynamic model

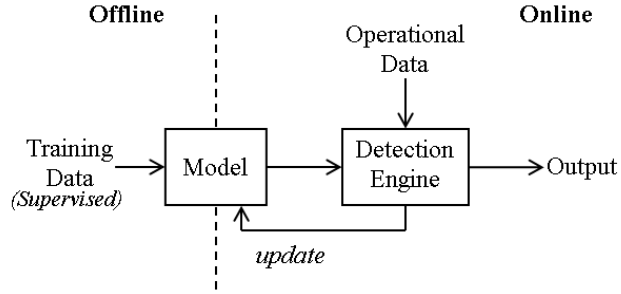
Most of the current ADS only employ a static model. After the training phase, the model remains static throughout the lifetime of the system as depicted in Figure 4(a). The majority of the ADS proposed for WSN in the literature are in this category. As mentioned in Section III-A, deriving a static *normal-model* during initialisation would be sufficient for most applications. However, for applications where the model of normal behaviour actually changes due to spatial or temporal conditions, a dynamic model is required.

There are some early works that *adaptively* regenerates models of normal behaviour such as [21] and [22]. In [21], anomalies were detected based on the statistics of normal traffics such as the minimum, maximum, and standard deviation of the packet arrival time in each sensor node calculated using a sliding window. In [22], the authors proposed the use of Hidden semi-Markov Model (HsMM) to dynamically change the model of Web users' browsing behaviour in order to detect distributed denial of service (DDoS). In their experiment, they used real traffic data collected in [23] for training and emulated DDoS attack using NS2 network simulator. Even though these systems maintain a dynamic model during operation, generation of initial model is still supervised as shown in Figure 4(b). For distributed autonomous systems where a purely normal dataset might not be readily available, it would be beneficial if unsupervised initial model generation could be implemented.

The majority of the unsupervised anomaly detection algorithms are unsuitable for adaptive model generation. Applications such as WSN and swarm robotics have very limited resources (computational, battery and memory) whilst the majority of the unsupervised model generation algorithms require high computational power. We have decided to look at biological distributed autonomous systems for inspiration. We are especially interested by immune network theory by Jerne [5] on how the immune components interact with each other and detect pathogens. This is in part, due to the work by Hart et al [24], Bersini and collaborators [25], and [26] that have showed that immune network exhibits the properties of tolerance and learning that we wish to endow the ADS with.



(a) Static model



(b) Dynamic model

Fig. 2. Maintenance of normal model in anomaly detection.

TABLE I
DESIRED ANOMALY DETECTION PROPERTIES IN EXISTING ADS.

No	Requirement	Existing ADS
i	Accuracy (Probability of detection)	High (Some $\geq$ 95%)
ii	Responsiveness	Not available
iii	Resource usage	Moderate to high
iv	Robustness	Minimal support

Table I is the summary on how existing ADS fulfil the general anomaly detection requirements in distributed autonomous systems. Existing ADS are able to detect anomalies in benchmark datasets with very high accuracy. However, these ADS normally involve complex algorithms that require considerable computational power. In terms of the ability to adapt and dynamically updating the *normal-model*, minimal support and consideration is given. This is fine as these ADS are normally implemented in applications where such adaptivity is not required.

IV. IMMUNOLOGY AS INSPIRATION

For clarification, a brief overview on vertebrate immune system and immune network theory are presented.

A typical view of the immune system is one that its primary function is to fight against invading pathogens such as bacteria, viruses and other harmful microorganisms. It consists of many interacting components (cells, molecules) spread across many organs to protect the body against infection. The immune system can be considered as two systems; an innate system which the host is born with and remains reasonably static, and an adaptive system which is constantly changing over the lifetime of the host. The innate immune system is made up of cells (phagocytes) that ingest and destroy foreign materials, proteins and enzymes while adaptive immunity is mediated by lymphocytes namely the B-cells and T-cells [27]. The cells of an innate immune system fight against pathogens, and are said to be non-specific and therefore react in a general way without prior exposure to them. In the adaptive immune system, lymphocytes have receptors on their surface and any molecule that binds to the lymphocyte's receptor is called an antigen. The receptor on T-cell is called T-cell receptor (TCR) while the receptor on B-cell is called antibody. During adaptive immune response, the binding of antibody to foreign antigens with enough strength (affinity) will eventually eliminate the antigen.

A. Clonal Selection and affinity maturation

As mentioned in Section IV, the elimination of an antigen by antibody depends on the affinity of binding between them. Antibodies with high affinity to antigens will divide and produce clones. These clones will then differentiate into either effector or memory cells. The memory cells have a longer life than normal B-cells and useful for similar infection during subsequent encounter. Burnet [28] called this the clonal selection theory of antibody production where only activated lymphocytes experience clonal expansion. During clonal expansion, some B-cells experience mutation where the antibodies are *edited* producing higher affinity to antigens. This process of increasing affinity is called affinity maturation and is important for faster immune response and detection of variants of the same antigen.

B. Immune network theory

The immune network theory by Jerne [5] postulated the immune system as a network of interacting immune components that exists even in the absence of antigen. It was theorised that interactions exist not only between antibodies and antigens but also between antibodies and antibodies. The network formed by these interactions representing an internal image of antigens. The network can respond either positively or negatively. A positive response results in cell proliferation, activation and antibody secretion. A negative response would lead to network suppression. From these interactions immunological behavior such as tolerance and memory emerge [29]. Many versions of the network models are presented in the literature e.g. in [30], [31] but generally it can be summarised [32] as

$$RPV = NetSim - NetSup + eNew - eDeath$$

where:
RPV = Rate of population variation;
$NetSim$ = Network stimulation;
$NetSup$ =Network suppression;
$eNew$ =Influx of new elements;
$eDeath$ = Death of unstimulated elements.

There are many published work in the literature based on immune network theory applied in different application areas. Examples include aiNet in [33] for data compression

and Resource Limited Artificial Immune System (RLAIS) in [34] that was further enhanced in [35]. Our purposed work has similar motivation and context to [35] but we adopt the immune network for anomaly detection in systems with changing *normal-model*.

V. Proposing unsupervised adaptive anomaly detection

In this paper, we propose an instance-based anomaly detection architecture capable of online adaptation based on immune network theory with unsupervised model generation.

Our study is different from [4] that is also inspired by immune network theory in that they only used immune network algorithm to represent the distribution of the original input dataset and then used Hierarchical Agglomerative Clustering (HAC) to perform clustering analysis in generating model of normal behaviour. In addition, their ADS employs a static model of normal behaviour. Instead our approach is to use (adapted) immune network algorithm for the whole adaptive anomaly detection process.

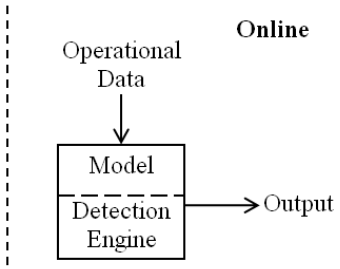
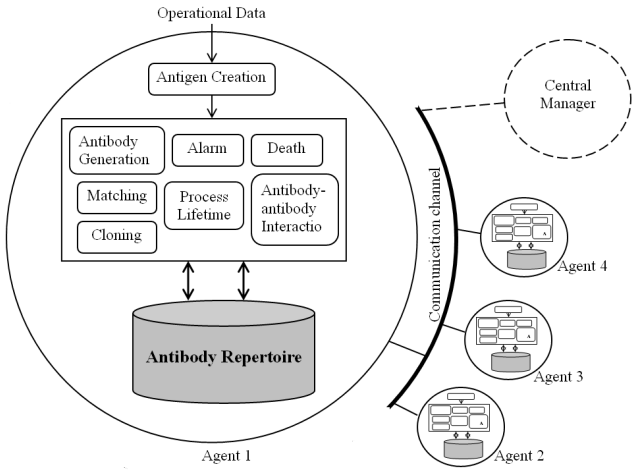


Fig. 3. Proposed immune inspired architecture.

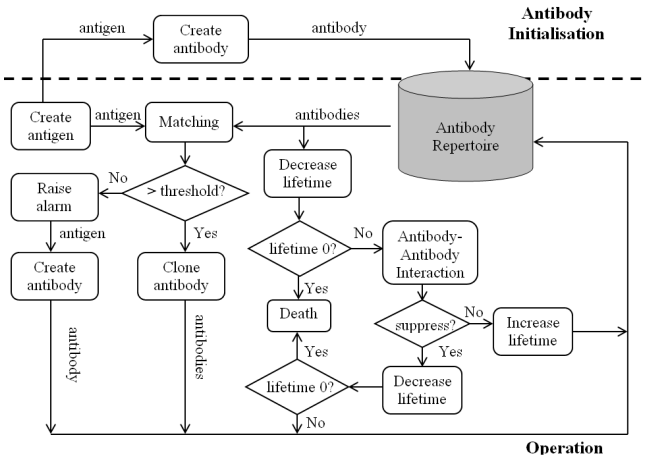
Figure 3 shows the general view of the unsupervised anomaly detection system with dynamic model maintenance we are proposing. Unlike other ADS, the model generation is online and the learning is continuous. This requirement is important in order for the proposed ADS to run on resource limited systems such as sensor nodes, and dynamically updating the *normal-model*.

Figure 4(a) is the high level architecture of the ADS in a distributed autonomous system. The ADS is implemented on each entity (agent) and entities in the network can communicate for collaborative detection of anomalies. In WSN, the entity is a sensor node while in swarm robotics, it is a robot. The collaboration can be managed either by a central manager or simply through local communications among the entities. This architecture has some similarity with the one proposed in [35].

The proposed immune network inspired ADS in each entity is shown in Figure 4(b). When an entity receives data from the environment, related antigen will be generated. The antigen represents the data seen by the entity while the antibody is used to detect such antigen. For example, in a swarm robotic system that is assigned a foraging task, the antigen and antibody may be represented as a feature vector of <*robot_id*, *average_food_collected*, *distance_travelled*,



(a) Global collaborative architecture (based on [36])



(b) Local immune network inspired architecture

Fig. 4. Proposed immune network inspired anomaly detection.

battery_life, *standard_deviation_food_collected*>. During the antibody initialisation phase, antibodies are generated and put in the initially empty antibody repertoire. The duration for antibody initialisation is depending on the type of application where the ADS is implemented. This is unsupervised learning as the antigens are unlabelled data.

After initialisation, each antigen is match with all antibodies in the antibody repertoire based on certain affinity function. For simplicity, assume that the matching is based on a similarity measure [refer to [38] for a detailed analysis on representation and affinity metrics]. Only the antibody with the highest affinity greater than predefined threshold (th) with the antigen will be cloned and mutated (Eq. 1). Mutations allow for gradual shifting of normal behaviour to be tolerated and updated accordingly (as depicted in lower left in Figure 5). The number of clones (cn) and probability of mutation (mp) is up to the ADS designer. However, due to resource constraints, the number of clones should be keep as minimal as possible to maintain a reasonable size of antibody population [37]. During matching if there is no antibody

with affinity greater than predefined threshold for an antigen, alarm will be raised as it may be anomalous antigen and a corresponding antibody is generated for this antigen. In this stage, the entity (ADS) might choose to share the antibody with other local entities or with central manager that would distribute the antibody. The generation of antibodies in this case is justifiable since the antigen might instead be a genuine normal pattern not seen before. With the generated antibody, consecutive normal antigens with same pattern will not be flag as anomaly. Such action allows the new (discrete) normal pattern to be adaptively learned (top middle in Figure 5). If it is an anomaly, we should detect it. Based on the assumption that anomalies happen less frequently than normal instances, the antibody generated by anomalous antigen will eventually dies off due to inactivation as depicted in the upper left in Figure 5.

$$\text{Max}(\text{Affinity}(\text{Antigen}, \text{Antibody})) \geq th \qquad (1)$$

During operation at every time interval t (disregard of whether there is antigen or not), the lifetime of each antibody in the antibody repertoire is reduced by certain constant value k_1 (Eq. 2). If the lifetime of an antibody is zero or less, remove the antibody from the antibody repertoire. Depending on how the interaction is implemented (on distance between antibodies or number of similar antibodies), an antibody will be suppressed or stimulated. Stimulated antibody will be added k_2 lifetime while suppressed antibody will be reduced k_3 lifetime. These parameters could be fixed if necessary. Such antibody-antibody interactions will maintain the antibody network (*normal-model*).

$$\text{Lifetime}_{Ab}(t) = \text{Lifetime}_{Ab}(t-1) - k_1 \qquad (2)$$

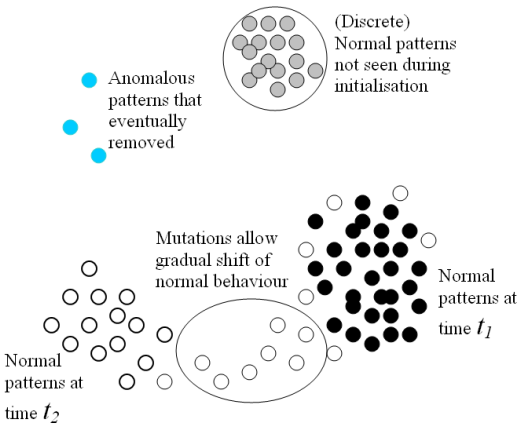


Fig. 5. Shifting of normal patterns.

Table II shows the mapping of proposed ADS to the requirements in section II-A.

VI. Conclusion

This paper proposes an unsupervised anomaly detection architecture with adaptive *normal-model* maintenance based on immune network theory by Jerne. It first initialises antibody repertoire and dynamically maintain the *normal-model* as the system operates. The proposed architecture aims to support the requirements for applications in resource constrained distributed autonomous systems such as WSN that also require dynamic update for the model of normal behaviour as the system operates.

Whilst the envisaged solution does offer a novel approach to addressing the problems of distributed autonomous systems, the potential advantages will only be realised if significant further research is performed. Specifically there are a number of parameters that have to be tuned in order to give the best balance across the properties of interest. This in itself represents an interesting multi-objective optimisation problem for which significant principled experimentation will be needed.

TABLE II
Mapping of desired anomaly detection properties in proposed ADS.

No	Requirement	Proposed ADS
i	Accuracy (True positive, probability of detection)	Depends on antibody repertoire (tunable)
ii	Responsiveness (Within acceptable time frame - application dependent)	Depends on antibody repertoire (tunable) and matching algorithm
iii	Resource usage (minimum number of detector)	Depends on antibody repertoire (tunable) and complexity of matching algorithm (trade-off with responsiveness)
iv	Robustness (able to change behaviour over time accordingly)	Supported by mutations, antibody-antibody interactions and generation of new antibody

References

[1] E. Nikolova and V. Jecheva, "Anomaly based intrusion detection based on the junction tree algorithm," *Journal of Information Assurance and Security*, vol. 2, pp. 184–188, 2007.

[2] D. E. Denning, "An intrusion detection model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222–232, 1987.

[3] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data," in *Applications of Data Mining in Computer Security*. Kluwer, 2002.

[4] L. Fang and L. Le-Ping, "Unsupervised anomaly detection based on an evolutionary artificial immune network," in *Proceedings of Applications of Evolutionary Computing, EvoWorkshops-2005*, ser. LNCS 3449. Springer, 2005, pp. 166–174.

[5] N. K. Jerne, "Towards a network theory of the immune system," *Annual Immunology*, vol. 125C, pp. 373–389, Jan 1974.

[6] L. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, pp. 102–114, Aug 2002.

[7] L. A. Segel and I. R. Cohen, *Design Principles for the Immune System and Other Distributed Autonomous Systems*. Oxford University Press, USA, 2001.

[8] Y. Liu and K. M. Passino, "Swarm intelligence Literature overview," Ohio State University, Tech. Rep., March 2000.

[9] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *First IEEE International Workshop on Sensor Network Protocols and Applications*. IEEE, 2003, pp. 293–315.

[10] J. Kim, P. Bentley, C. Wallenta, M. Ahmed, and S. Hailes, "Danger is ubiquitous: Detecting malicious activities in sensor networks using the dendritic cell algorithm," in *Proceedings of the 5th International Conference on Artificial Immune Systems (ICARIS-06)*, ser. LNCS 4163. Springer, 2006, pp. 390–403.

[11] A. Burns, I. J. Hayes, G. D. Baxter, and C. J. Fidge, "Modelling temporal behaviour in complex socio-technical systems," University of York, Computer Science Dept, Tech. Rep. tech. report YCS 390, 2005.

[12] L. Bayindir and E. Sahin, "A review of studies in swarm robotics," *Turkish Journal of Electrical Engineering and Computer Sciences.*, vol. 15, no. 2, 2007.

[13] A. L. Christensen, "Fault detection in autonomous robots," *Ph.D. dissertation*, 2008.

[14] A. K. Ghosh and A. Schwartzbard, "A study in using neural networks for anomaly and misuse detection," in *Proceedings of the 8th conference on USENIX Security Symposium*. USENIX Association, 1999, pp. 12–12.

[15] N. Ye, "A markov chain model of temporal behavior for anomaly detection," in *Proceedings of the 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*. IEEE Computer Society Press, 2000, pp. 171–174.

[16] V. G. Jecheva and E. P. Nikolova, "Learning problem and bcjr decoding algorithm in anomaly-based intrusion detection systems," *Journal of Software*, vol. 2, no. 6, pp. 42–52, 2007.

[17] Q. Qian and M. Xin, "Research on hidden markov model for system call anomaly detection," in *Pacific Asia Workshop on Intelligence and Security Informatics (PAISI 2007)*, ser. LNCS 4430. Springer, 2007, pp. 152–159.

[18] K. Leung and C. Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," in *Proceedings of the 28th Australasian Conference on Computer Science - Volume 38*. Australian Computer Society, Inc, 2005, pp. 333–342.

[19] C. Yang, F. Deng, and H. Yang, "An unsupervised anomaly detection approach using subtractive clustering and hidden markov model," in *Proceedings of the Second International Conference on Communications and Networking in China (CHINACOM '07)*. IEEE, 2007, pp. 313–316.

[20] The third international knowledge discovery and data mining tools competition dataset (kdd99 cup). [Online]. Available: http://kdd.ics.uci.edu/database/kddcup99.html

[21] I. Onat and A. Miri, "An intrusion detection system for wireless sensor networks," in *IEEE International Conference on Wireless and Mobile Computing*. IEEE Computer Society, 2005, pp. 253–259.

[22] Y. Xie and S. Yu, "A dynamic anomaly detection model for web user behavior based on hsmm," in *Proceedings of the 10th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD 2006)*. IEEE Press, 2006, pp. 811–816.

[23] Worldcup98 dataset. [Online]. Available: http://ita.ee.lbl.gov/html/contrib/WorldCup.html

[24] E. Hart, H. Bersini, and F. C. Santos, "How affinity influences tolerance in an idiotypic network," *Journal of Theoretical Biology*, vol. 249, no. 3, pp. 422–136, 2007.

[25] H. Bersini and F. J. Varela, "Hints for adaptive problem solving gleaned from immune network," in *Proceedings of the Workshop Parallel Problem Solving from Nature*, ser. LNCS 496. Springer, 1990, pp. 343–354.

[26] H. Bersini, "Self-assertion versus self-recognition: A tribute to francisco varela," in *Proceedings of the First International Conference on Artificial Immune Systems (ICARIS 2002)*. University of Kent Publishing Unit, 2002, pp. 107–112.

[27] C. A. Janeway, P. Travers, M. Walport, and M. J. Chlomchik, *Immunobiology: The Immune System in Health and Disease*, 6th ed. Garland Publishing, 2005.

[28] F. M. Burnet, *The clonal selection theory of acquired immunity*. Cambridge University Press, Cambridge, 1959.

[29] J. Timmis, P. Andrews, N. Owens, and E. Clark, "An interdisciplinary perspective on artificial immune systems," *Evolutionary Intelligence*, vol. 1, pp. 5–26, March 2008.

[30] J. D. Farmer, N. H. Packard, , and A. S. Perelson, "The immune system, adaptation, and machine learning," *Physica D*, vol. 2, no. 1-3, pp. 187–204, 1986.

[31] F. Varela and A. Coutinho, "Second generation immune networks," *Immunology Today*, vol. 12, no. 5, pp. 159–166, 1991.

[32] L. N. de Castro and J. Timmis, *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer, 2002.

[33] L. N. de Castro and F. J. V. Zuben, "An evolutionary immune network for data clustering," in *Proceedings of the VI Brazilian Symposium on Neural Networks (SBRN 2000)*. IEEE Computer Society, 2000, pp. 84–89.

[34] J. Timmis and M. Neal, "Investigating the evolution and stability of a resource limited artificial immune system," in *Proceedings of the Genetics and Evolutionary Computation Conference - special workshop on artificial immune systems (GECCO 2000)*. AAAI press, 2000, pp. 40–41.

[35] M. Neal, "An artificial immune system for continuous analysis of time-varying data," in *Proceedings of the First International Conference on Artificial Immune Systems (ICARIS 2002)*. University of Kent Publishing Unit, 2002, pp. 76–85.

[36] R. de Lemos, J. Timmis, M. Ayara, and S. Forrest, "Immune-inspired adaptable error detection for automated teller machines," *IEEE Transactions on Systems, Man, and Cybernatics - Part C: Application and Reviews*, vol. 37, no. 5, pp. 873–886, 2007.

[37] J. Timmis, R. de Lemos, M. Ayara, and R. Duncan, "Towards immune inspired fault tolerance in embedded systems," in *Proceedings of 9th International Conference on Neural Information Processing*. IEEE, 2002, pp. 1459–1463.

[38] A. A. Freitas and J. Timmis, "Revisiting the foundations of artificial immune systems: A problem-oriented perspective," in *Proceedings of the Second Internaltional Conference on Artificial Immune Systems (ICARIS 2003)*, ser. LNCS 2787. Springer, 2003, pp. 229–241.