Cognitive Approach Based User Node Activity Monitoring for Intrusion Detection in Wireless Networks

G Sunilkumar [1], Thriveni J [1], K R Venugopal [1], L M Patnaik [2]

**[1] Department of Computer Science and Engineering
University Visvesvaraya College of Engineering, Bangalore University
Bangalore 560 001, India**

**[2] Vice Chancellor, Defence Institute of Advanced Technology
Pune, India**

Abstract

Cognitive networks are the solution for the problems existing on the current networks. Users maintain integrity of the networks and user node activity monitoring is required for provision of security. Cognitive Networks discussed in this paper not only monitor user node activity but also take preventive measures if user node transactions are malicious. The intelligence in cognitive engine is realized using self-organizing maps (CSOMs). Gaussian and Mexican Hat neighbor learning functions have been evaluated to realize CSOMs. Experimental study proves the efficiency of Gaussian Learning function is better for cognition engine. The cognition engine realized is evaluated for malicious node detection in dynamic networks. The proposed concept results in better Intrusion detection rate as compared to existing approaches.

Keywords: *Intrusion Detection, Cognitive networks, Soft computing, Self-organizing maps, Computational intelligence.*

1. Introduction

In recent years, Cognition has become buzzword that can be achieved for different networks, then such networks are called as cognitive networks. Cognitive networks are also called as smart networks. They can be used to improve Quality of Service, Security and access control, resource management. Cognitive networks impart computational intelligence by Soft computing techniques like neural networks, Fuzzy logic, artificial immune systems etc., The cognitive networks provide better end to end performance compared to other networks.

Cognitive networks should use network metrics and patterns as input for decision making process and it should provide output in the form of a set of actions that can be implemented in modifiable network elements. Ideally, the cognitive networks should be forward looking, rather than reactive and it should make attempt to adjust to problems before it occur. For implementing a cognitive network, it requires a system which is more complex in terms of architecture and operation. To build Intrusion Detection System (IDS), it needs to consider many issues, such as data collection, data preprocessing, intrusion recognition, reporting and response [1]. Among these, intrusion recognition is the most critical task. Monitored data is compared with detection models, so that it describes the patterns of intrusive behavior and from these successful and unsuccessful intrusion attempts can be detected.

In cognitive networks, the learning happens through the feedback loop called Observe, Orient, Decide and Act (OODA) loop [2][3]. In observation phase, collection of statistics and behaviours of the node from the environment will be done. In orientation phase, the percentage of deviation is calculated between monitored behaviour and actual behaviour of the node. The necessary decisions is taken from the computational intelligence in decide and act phase. Cognition uses the Machine learning and it improves its performance from the gained experience [4]. Poole et al.,[5] have designed Computational Intelligence system. Which learns from experience and takes appropriate decisions within the given perceptual limitations and finite Computation.

Intrusion detection systems are categorised into two. Namely, misuse detection and anomaly detection. In misuse detection, it identifies intrusions by matching observed transactions with pre-defined behaviour. But it fails easily for unknown intrusions. Anomaly detection is based on artificial intelligence and it overcomes the drawback of misuse detection [6]. Intrusion detection systems will dynamically monitor the events and it decides whether these events are indicative of an attack or it constitutes a genuine use of the system [7].

Self-Organizing Map (SOM) is an unsupervised approach and it requires neighbor learning function for self-learning of behaviors of the node. There are two neighbor learning functions available for unsupervised approach, they are Mexican hat and Gaussian functions. User node activity monitoring is required for provision of security to the network and the proposed concept in this paper not only monitors user node activity but it takes necessary preventive measures, if the transactions are malicious. For this, cognitive engine requires intelligence and it can be achieved by using self-organizing maps. The implementation evaluates both functions to choose the best for the CSOM to detect Intrusion.

The key challenges of Cognitive Networks are:
Network Complexity: The introduction of wireless links in the network increases the complexity because of number of nodes and alternate routes and as well as the number of communication mediums and protocols. Radio signals are subject to frequent fading and signal interference, but the wireless nodes communicates over radio channels and the nodes are joining the network in ad-hoc manner and mesh type connections can be established. Mobility allows wireless terminals to dynamically change their location, point of attachment to the network. It also affects path availability, it makes difficult to reach stability in limited timeframe. Hence the network management and optimization must add as functionalities for self-healing in cognitive network.

Heterogeneity: The internet uses different combinations of transmission technologies, applications and transmission protocols and no layer in the Transmission Control Protocol or Internet Protocol accounts for heterogeneity. To improve the performance; the connection is divided into segments, each optimized for a particular domain. Enhancing performance requires awareness of underlining transmission technologies between sender and receiver nodes over the entire path. Across multiple domains, the optimization process should be distributed and should achieve the goal which is defined at the connecting ends.

Quality of Service (QoS): To support QoS requirements of different applications and users, limited provisioning of delay bounds and bandwidth unavailability require the implementation of reservation control mechanisms. From the technical and the business perspective.

***Motivation*:** The existing intrusion prevention techniques like firewalls, access control or encryption, have failed to protect the networks completely. Hence intrusion detection systems (IDS) have become major component of security infrastructure to detect the threats before it damages the network. The intrusion detection faces difficulty for huge network traffic volumes and abnormal behavior and it requires continuous adaptation for constantly changing environment.

Contribution: Malicious nodes are detected in cognitive networks using self-organizing maps. To achieve high Intrusion detection accuracy and fast processing times, artificial intelligence and machine learning is used. Fuzzy based SOMs and cognition based SOMs are compared. Gaussian and Mexican hat neighbour learning functions are evaluated to select the best learning function for cognitive network.

Organization: This paper is organized into the following sections: Section II gives the related work, Section III Explains the background and Section IV gives the problem definition, Section V explains the system Model, Section VI describes the algorithms, the performance evaluation of the proposed system model is given in Section VII, and Conclusions are given in Section VIII.

2. Related Work

With the evolution of communication technologies and paradigm of shift from static to mobile access and centralized to distributed network has made the way for more research in security. There are many security tools available for network. Among these, intrusion detection systems are important tool. Many researchers have proposed various intrusion detection systems for wireless networks. In this paper we have discussed some of intrusion detection systems.

Doumit and Agrawal [8] have proposed light-weight intrusion detection system. It uses both anomaly based intrusion detection and host-based data collection mechanism. In this approach, all the normal behaviour of a node will be stored into knowledgebase and if suspicious activities are found in node then it compares with knowledgebase. It also uses the hidden markov model for acquired sensor reading based on the self-organized criticality of the deployment region. The addition of new sensor nodes is easy and it is a hybrid model. The drawback of this model is, it mainly focuses on the individual sensor nodes rather than the complete network.

Strikos [9] have proposed cluster based intrusion detection model. It divides the entire network into clusters. Cluster-first protocol forms the clusters and cluster-heads. Once the cluster heads are formed, it monitors all other nodes in the cluster. The clusters are further divided into teams and it monitors the cluster head also in round-robin fashion. This model uses centralized routing. All cluster heads will determine the cluster set and then IDS are deployed on those cluster nodes. The drawback of this model is that, it requests retransmission of all previous messages from the suspicious node to detect malicious node.

Harshal et al.,[10] have proposed a cluster based hierarchical model. It uses an ant colony based approach for intrusion detection. It is divided into two phases. The first phase is initial configuration of network and second phase is for identification of attacks and routing. In this model the IDS is deployed on cluster heads using hierarchical approach. Here the network is divided into clusters and later cluster heads are selected. Then ant pheromone is deployed into cluster heads for routing and to detect malicious activities based on pheromone values. The main advantages of this model are, it detects both internal and external attack. The resource consumption for anomalies detection is low. The drawback is formation of cluster without malicious node is difficult.

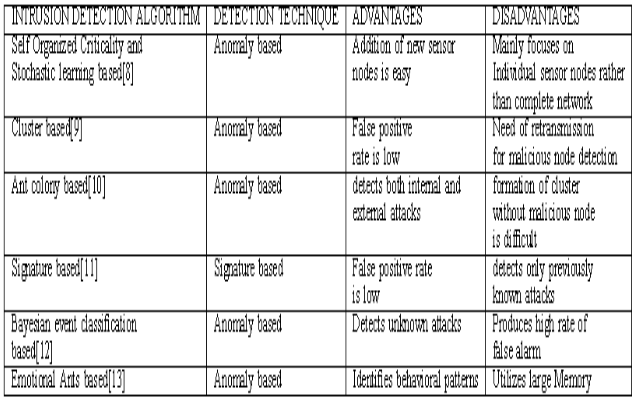
Nadkarni and Shenoy [11] have provided signature-based intrusion detection system. This requires constant updates of their database because it depends on predefined set of attack signatures. This model detects the malicious threats by comparing the monitored packets of network and the database of signatures. The advantage of this model is, it produces a low rate of false positive alarm. The drawback is it detects only previously known attacks.

Christopher et al., [12] explains the anomaly based intrusion detection method. In this method it creates a base line profile of the normal system. Hence it will distinguish the incoming system activity is normal or anomalous. If the activities are found to be anomalous then the detection system generates anomaly alarm. The benefit of this model is, it detects the previously unknown attacks. The demerit is, it produces high rate of false alarm.

Banerjee et al.,[13] have provided intrusion detection system based on emotional ants for sensor networks. This approach is also called as IDEAS and it is a parallel search algorithm. In this algorithm, multiple ant agents are used to deploy pheromone values on nodes for malicious detection. The emotional ants identifies the direct and indirect paths among neighbours and it communicates the characteristic of path through pheromone balancing to other ants. If any imbalance is found in pheromone values then it alerts the network administrator. The merit of this model is, it identifies behavioural patterns. The drawbacks are, it utilizes large memory to store pheromone values and the congestion increases in network due to ant packets.

Comparison of Various Intrusion Detection Algorithms is given in Table I.

Table I: Comparison of Various Intrusion Detection Algorithms



INTRUSION DETECTION ALGORITHM	DETECTION TECHNIQUE	ADVANTAGES	DISADVANTAGES
Self Organized Criticality and Stochastic learning based[8]	Anomaly based	Addition of new sensor nodes is easy	Mainly focuses on Individual sensor nodes rather than complete network
Cluster based[9]	Anomaly based	False positive rate is low	Need of retransmission for malicious node detection
Ant colony based[10]	Anomaly based	detects both internal and external attacks	formation of cluster without malicious node is difficult
Signature based[11]	Signature based	False positive rate is low	detects only previously known attacks
Bayesian event classification based[12]	Anomaly based	Detects unknown attacks	Produces high rate of false alarm
Emotional Ants based[13]	Anomaly based	Identifies behavioral patterns	Utilizes large Memory

3. Background

Jazzar and Aman [14] have designed fuzzy based selforganizing maps (SOM-FCM) for intrusion detection. It explains that the relationships among the neurons will be built by the global matrix. This matrix is built by assigning the weights to neurons which is according to total effect factor. Then the fuzzy cognitive maps will calculate the causal relation between the suspicious or odd neurons, which is noted by the SOM. To determine the attack and the severity of odd neurons. If the neurons have low causal relations, then that neurons will be dropped to reduce the false alerts.

Fuzzy based self-organizing maps have achieved the detection rate as 89.71% and the false alarm rate as 10.29%. The main drawback of this concept is it uses the data sets for training. The creation of datasets is more complicated and the reduced false alert by this concept is not sufficient for real time implementation.

4. Problem Definition

Given a network with n nodes deployed randomly;

The objectives are:

(i) Realization of the cognition engine using self-organizing maps.
(ii) To evaluate the Mexican and Gaussian neighbor learning functions.
(iii) To improve the performance of network using cognition.
(iv) To increase the Intrusion Detection rate.
(v) To reduce the False Detection rate.

Assumptions are:

(i) Random deployments of nodes are done to create test bed.
(ii) Often movements of nodes are considered.
(iii) All nodes are having same transmission range.
(iv) Unsupervised method is used for monitoring the transactions of nodes.

5. System Model

Dynamic networks or ad-hoc networks are characterized by the variation of nodes and their behaviors. In order to achieve cognition in such networks, we proposed to use kohonen self-organizing maps techniques to observe and study the node behavior.

A. Architecture of Cognition Engine

The cognitive architecture is shown in Fig. 1. The node repository maintains the node identities and the architecture recognizes the nodes based on these ids. Numerous data transactions are carried out among the nodes. These transactions need to be monitored to analyze node misbehaviors or malicious nodes. The observed node behaviour is stored in the monitored behavior section of the cognitive frame work. Cognition could be achieved, if the observed behavior is analyzed. To impart intelligence, the nodes use self-organizing maps to predict its peer node behavior. These patterns are stored in the relative transaction set of the architecture. If the observed and the relative behaviour transactions match, then the node behavior is considered normal else-if there is a variation in the behavior and is above the said threshold then the node is said to be malicious.

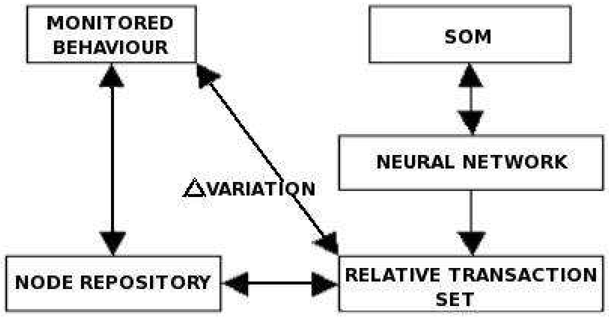


Fig. 1 Architecture of Cognition Engine.

B. Self - Organizing maps (SOM)

It uses vector quantization data compression technique to represent the multidimensional data to one or two dimensional data. It creates a network which stores the information called as training sets, which contains topological relationships among nodes. SOM classifies the data without supervision and it does not require any target vectors like supervised training techniques.

C. Learning Algorithm

The nodes are grouped together to form zones. Each zone represents a particular feature and it has a lattice. Initially, individual nodes weights are initialized to zero. A vector is randomly selected from the set of training data and is given to the lattice of the zone for training. If the input vector arrives, then all individual nodes weight is compared with the input vector. If it matches then that node from particular zone is winning node and is called as best matching unit (BMU).

D. Calculating the best matching unit

To calculate the BMU, iterate through all the nodes and calculate the Euclidean distance between each nodes weight vector and input vector. The node with the closest weight vector is considered as the BMU.

Euclidean distance is given as

$$E - dist = \sqrt{\sum_{j=0}^{j=m} (v_j - w_j)^2} \qquad (1)$$

Where v is the current input vector and w is nodes weight vector.

In kohonen learning algorithm, the area of neighbourhood decreases with time. The exponential decay function is given by

$$\sigma(t) = \sigma_0 exp(-\frac{t}{\gamma}), \quad t = 1, 2, 3, ... \qquad (2)$$

where γ is time constant, t is current time-step and σ_0 is the width of the lattice at time t_0.

E. Adjusting the weights

All the node within the BMUs neighborhood, including the BMU, its weight vector gets adjusted by the following equation:

$$W(T+1) = W(T) + L(T)(V(t) - W(T)) \qquad (3)$$

Where T is the time step, L is learning rate which decreases with time and V is input vector.

The decay of the learning rate is calculated with each iteration and is given as

$$L(t) = L_0 exp(-\frac{t}{\gamma}), \quad t = 1, 2, 3, \ldots \qquad (4)$$

Where γ is time constant, t is current time-step and L_0 is the width of the lattice at time t_0.

F. Notations

The Symbols and Notations used in Algorithms is listed in Table II.

Table II: Notations Used In Algorithms

Notation	Significance
$w_i a_j$	Synaptic Weights
$a_j(n)$	Winning neuron for Mexican Hat
$d(\Delta)^2$	Neighbor learning function
σ^2	Variance Parameter
Δx	Change in X Co-ordinate
Δy	Change in Y Co-ordinate
x(n)	Input Vector
G(n)	Winning Neuron for Gaussian
s(i)	Distance from the Winning Neuron
γ(i)	Updated Strength of Winning Neuron

6. Algorithms

There are two types of unsupervised neighbor learning functions, first one is Mexican hat learning function and second one is Gaussian learning function.

A. Cognition based Mexican Hat Learning Algorithm (CMHLA)

To create a competition among neurons, the lateral connections are used. The neuron which is having the largest activation level among all in the output layer is the winner and the winner neuron produces output signal. The inhibitory effects are produced by the lateral feedback on the basis of distance from the winning neuron and this is possible by using CMHLA function: The synaptic weights among neurons in the output layer is described by:

$$a_j(n) = t[P_j(n) + \sum_i w_i a_j + i(n-1)] \qquad (5)$$

The learning rate for CMHLA is given as:

$$Learning\ \ Rate = 0.6259\sigma \qquad (6)$$

Neighbor learning function is given as:

$$d(\Delta)^2 = \frac{\Delta x + \Delta y}{\sigma^2} \qquad (7)$$

$$Neighbor\ \ learning = (1 - d(\Delta)^2)e^{-\frac{d(\Delta)^2}{2}} \qquad (8)$$

B. Cognition based Gaussian Neighbor Learning Algorithm (CGNLA)

The learning algorithm have two aspects, they are competition and co-operation among the neurons in output lattice. In competitive learning, the competition is implemented, here the input vector x(n) is compared with weight vector which is from weight matrix w of the winning neuron G(n) and the distance for winning neuron is minimum.

The neurons which is located in a topological neighbourhood of the winning neurons G(n), their weights gets updated with a strength γ(i) with respect to their distance s(i) from the winning neuron

$$s(i) = |q(i,:) - q(G(n),:)|, \quad for \quad i = 1, \ldots, m \qquad (9)$$

The learning rate for CGNLA is given as:

$$Learning\ \ Rate = 1.7741\sigma \qquad (10)$$

Neighbor learning function is given as:

$$Neighbor\ \ learning = e^{-\frac{(\Delta x + \Delta y)^2}{2\sigma^2}} \qquad (11)$$

The map formation is heavily dependent on learning parameters; the learning gain and spread of CGNLA function. Usually both parameters are time - varying. Generally, the learning gain is almost equal to unity in ordering phase and denying the convergence phase, it reduces below unity. During the ordering phase, the spread of the neighborhood function initially include all neurons for any winning neuron and it slowly reduces to include only few neurons in neighbourhood of winner and during the convergence phase, only the winning neuron is included by the neighborhood function.

7. Performance Evaluation

A. Simulation Background

To evaluate the performance of the cognitive engine, we have developed a simulator to simulate the network, in this, 100 nodes have been deployed to create the test bed using random topology and it is dynamic in nature. Every node do the valid transactions with the internodes and sometimes the same node may do the invalid transactions at random probabilities. Some time the same node behaves like a malicious node or other malicious node may do the invalid transaction, at that situation the recipient node detects the invalid transaction and rejects.

The packet from the malicious nodes is detected using kohonen self-organizing maps. In this simulator, both learning algorithms is implemented and improved using Cognition and evaluated with respect to false detection rate. The snap shots of the simulator before and after the simulation are shown in Fig. 2. and Fig. 3. The network is simulated for 30 times and the average reading is taken to plot the graph. The Metrics used for performance evaluation are given below;

Training cycle: Number of behaviors used to train the node.
Detection rate: is percentage of Number of Malicious transactions detected over the total number of malicious transactions in the network.
Learningrate: The percentage of learning of behaviors of node.
Execution time: Total time taken to complete the training process with respect to number of training cycle.

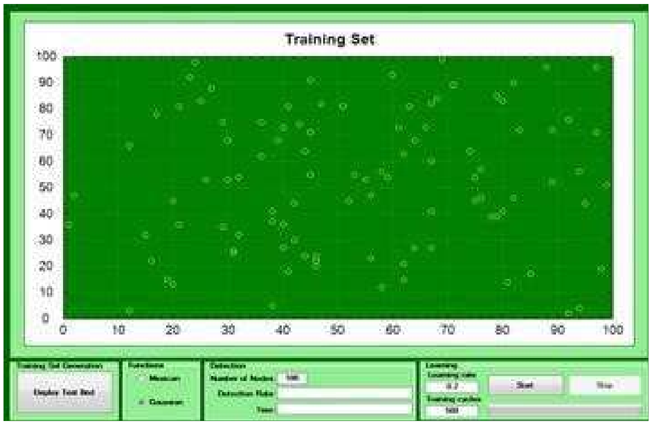


Fig. 2. Simulator before simulation.

From Fig. 2. and Fig. 3. It can be observed that, the buffers of the node are empty before simulation and after simulation it is filled, it indicates that the nodes have been trained. Positions of the nodes are changed before and after simulation for the same test bed due to dynamic nature of node.

The variations in the detection rate with respect to learning rate can be observed in Fig 4. This figure shows that, for CGNLA, as the learning rate increases, the detection rate also increases and as the learning rate reaches to 1, the detection rate reaches the maximum of 94% and becomes saturates for further increase in learning rate. Similarly for CMHLA, the maximum detection rate is achieved as 67% at learning rate 0.5 and further it reaches almost saturation state. The increase in the detection rate is due to cognition. The existing SOM-FCM can also be seen in figure and it has achieved detection rate around 90%.

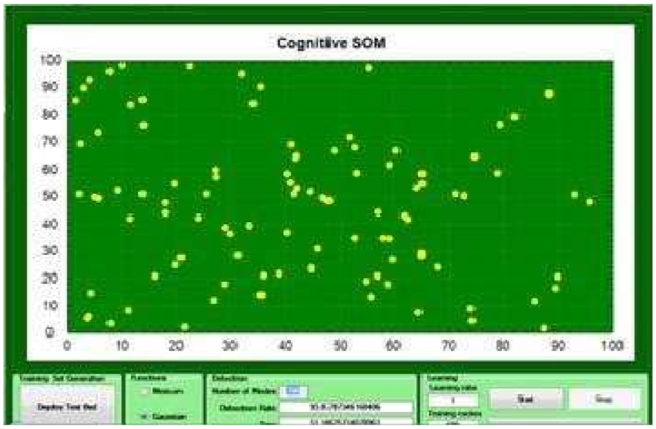


Fig. 3. Simulator after simulation

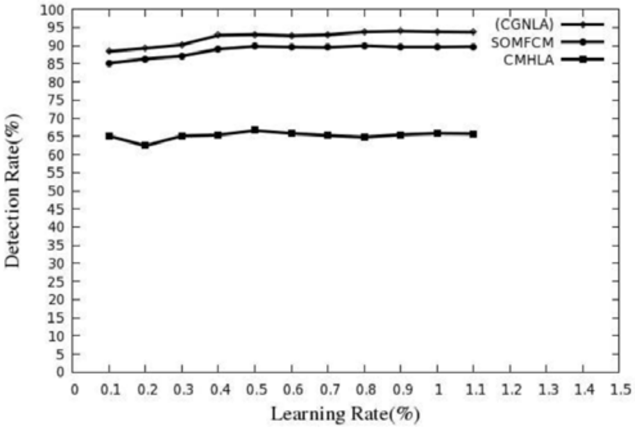


Fig. 4. Learning Rate Vs Detection Rate

From Fig 5, variations in detection rate with respect to training cycles can be observed. As the number of training cycles increases, the malicious node detection rate also increases and false alarm rate decreases for CGNLA and

CMHLA. The detection rate of CGNLA is more as compared to SOM-FCM. The reduction in the false alarm rate is due to achieving the cognition in the network.

Linearly increasing in execution time with respect to training cycles can be seen in Fig 6. For 500 training cycles, the maximum execution time for CGNLA is around 69 seconds and for CMHLA is around 78 seconds. It shows that CGNLA is more efficient than CMHLA. The variations in the graphs are due to dynamic networks and random topology.

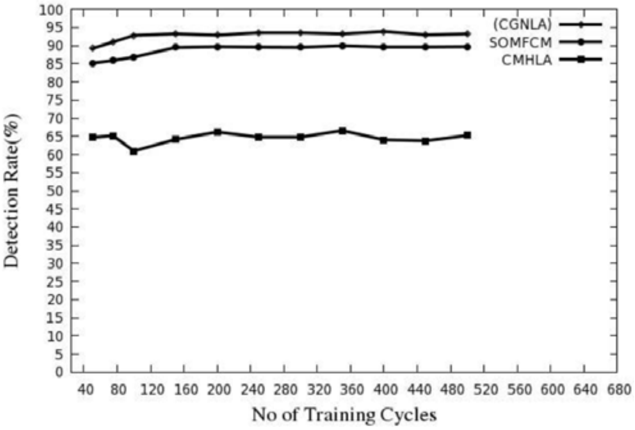


Fig. 5. Training Cycle Vs Detection Rate

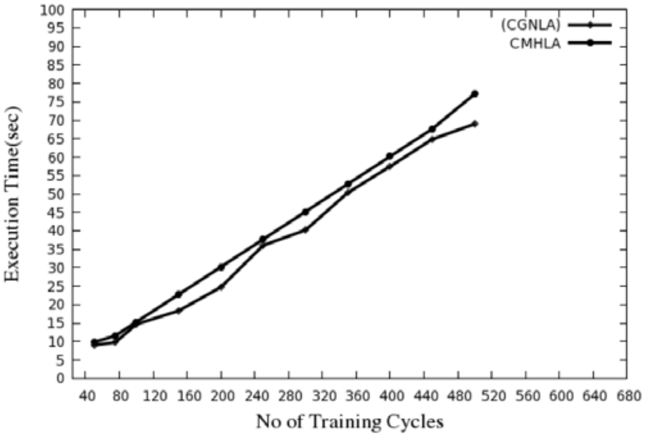


Fig. 6. Training Cycle Vs Execution Time

B. Results

In this experiment, the performance measure of Cognition based - Self organizing maps (CSOM) are carried out with respect to the Intrusion detection rate and the false alarm, it is calculated as follows

$$IDR = (\frac{TMN - TMND}{TMN})100 \qquad (12)$$

where IDR is Intrusion Detection Rate, TMN is Total Malicious Node, TMND is Total Malicious Node Detected.

$$FIDR = (\frac{TNDM}{TNN})100 \qquad (13)$$

where FDR is False Intrusion Detection Rate, TNDM is Total Number of Normal Nodes that are Detected as Malicious, TNN is Total Number of Normal Nodes.

The obtained CSOM Result is compared with the existing SOM based Intrusion detection sensors is given in the following table.

Table III: False Alarm Comparison

Method	Detection Rate	False Detection Rate
SOM	88.30%	11.66%
SOM-FCM	89.71%	10.29%
CSOM	94.12%	5.88%

The performance of the normal Self-organizing map is 88.3%, Fuzzy based self-organizing map is around 90% and the Cognition based self-orgnizing map is around 94%. The table shows that cognition based learning can achieve good performance.

8. Conclusions

Cognitive network provides solutions to the problems existing on the current networks. Cognitive engine is considered as the most critical part of cognitive network design. The approach discussed here uses self-organizing maps to impart intelligence to the cognition engine. The cognition engine evaluated on various test beds exhibit an average malicious node detection rate of about 94%. Both Cognition based Gaussian and Mexican hat algorithms are evaluated. They are computationally heavy and exhibit higher network response time.

Further supervised learning technique can be incorporated to decrease the computational overhead and network response time.

References

[1] Shelly Xiaonan Wu and Wolfgang Banzhaf, “The Use of Computational Intelligence in Intrusion Detection Systems: A

Review," in Applied Soft Computing, vol. 10, pp. 1–35, Elsevier, 2010.

[2] J. Mitola, "Cognitive Radio: An Integrated AgentArchitecture for Software Defined Radio," in PhD thesis, Royal Institute of Technology (KTH), 2000.

[3] John Boyd, "A Discourse on Winning and Losing: Patterns of Conflict," 1986.

[4] M. A. L. Thathachar and P. S. Sastry, "Networks of Learning Automata," Kluwer Academic Publishers, 2004.

[5] D. Poole, A. Mackworth, R. Goebel, "Computational IntelligenceA Logical Approach," Oxford University Press, Oxford, 1998.

[6] D. E. Denning, "An Intrusion Detection Model," IEEE Transactions on Software Engineering, vol. 2, no. 13, pp. 222–232, 1987.

[7] H. Debar, M. Dacier, A. Wespi, "Towards a Taxonomy of Intrusion- Detection Systems," Computer Networks, vol. 8, no. 31, pp. 805–822, 1999.

[8] Doumit S. S, Agrawal D. P, "Self-Organized Criticality and Stochastic Learning Based Intrusion Detection System for Wireless Sensor Networks," vol. 1, pp. 609–614, October 2003.

[9] Andreas A. Strikos, "A Full Approach for Intrusion Detection in Wireless Sensor Networks," School of Information and Communication Technology (KTH), Sweden, 2007.

[10] Harshal A. Arolkar, Shraddha P. Sheth, Vaidehi P. Tamhane, "Ant Colony Based Approach for Intrusion Detection on Cluster Heads in WSN," in Proceedings of ICCCS, ACM, February 2011.

[11] Nadkarni S, P. P. Shenoy, "A Causal Mapping Approach to Constructing Bayesian Networks," in Decision Support Systems, pp. 259–281, 2004.

[12] Christopher K, M. Darren, R. William and V. Fredrik, "Bayesian Event Classification for Intrusion Detection," 2003.

[13] Banerjee S, Grosan C, Abraham A, "IDEAS: Intrusion Detection Based on Emotional Ants for Sensors," vol. 1, pp. 344–349, September 2005.

[14] Mohmoud Jazzar, Aman Bin Jantan, "Using Fuzzy Cognitive Maps To Reduce False Alerts In SOM-Based Intrusion Detection Sensors," in Proceedings of 2nd Asia International Conference on Modeling and Simulation, 2008.

Sunilkumar G has completed Bachelor of Engineering in Electronics and Communications from Visvesavaraya Technological University,Belgaum, Master of Engineering in Information Technology from University Visvesvaraya College of Engineering, Bangalore University, Bangalore.he has 5 years of teaching experience. Presently he is working in Dept. of CSE at ACE, Bangalore and pursuing his Ph.D in Cognitive Networks.

Thriveni J has completed Bachelor of Engineering, Masters of Engineering and Doctoral Degree in Computer Science and Engineering. She has 4 years of industrial experience and 16 years of teaching experience. Currently she is an Associate Professor in the Dept. of CSE, University Visvesvaraya College of Engineering, Bangalore. Her research interests include Networks, Data Mining and Biometrics.

Venugopal K R is currently the Principal, University Visvesvaraya College of Engineering, Bangalore University, Bangalore. He obtained his Bachelor of Engineering from University Visvesvaraya College of Engineering. He received his Masters degree in Computer Science and Automation from Indian Institute of Science Bangalore. He was awarded Ph.D. in Economics from Bangalore University and Ph.D. in Computer Science from Indian Institute of Technology, Madras. He has a distinguished academic career and has degrees in Electronics, Economics, Law, Business Finance, Public Relations, Communications, Industrial Relations, Computer Science and Journalism. He has authored 31 books on Computer Science and Economics, which include Petrodollar and the World Economy, C Aptitude, Mastering C, Microprocessor Programming, Mastering C++ and Digital Circuits and Systems etc.. During his three decades of service at UVCE he has over 250 research papers to his credit. His research interests include Computer Networks, Wireless Sensor Networks, Parallel and Distributed Systems, Digital Signal Processing and Data Mining.

L M Patnaik is a Vice Chancellor, Defense Institute of Advanced Technology, Pune, India. He was a Professor since 1986 with the Department of Computer Science and Automation, Indian Institute of Science, Bangalore. During the past 35 years of his service at the Institute he has over 700 research publications in refereed International Journals and refereed International Conference Proceedings. He is a Fellow of all the four leading Science and Engineering Academies in India; Fellow of the IEEE and the Academy of Science for the Developing World. He has received twenty national and international awards; notable among them is the IEEE Technical Achievement Award for his significant contributions to High Performance Computing and Soft Computing. His areas of research interest have been Parallel and Distributed Computing, Mobile Computing, CAD for VLSI circuits, Soft Computing and Computational Neuroscience.