Anomaly Intrusion Detection Based on A hybrid GSA–SVM Classifier

Salima Omar Ben Qdara

University of Benghazi

Faculty of education

Abstract

Support vector machine (SVM) is commonly used in IDSs because of its robustness and efficiency in the network classification. However, Parameter optimization influences the classification accuracy of SVM significantly. In order to improve, a hybrid classifier is designed based on a combination of the GSA and SVM algorithms to optimize the accuracy of the SVM classifier. In the GSASVM classifier, the GSA guides the selection of potential subsets that lead to the best detection accuracy. The GSA-SVM algorithm evaluated using KDD CUP 99 data set and compared to the outperformance of the original SVM algorithms. The results show that the performance of GSA-SVM algorithm has higher detection rate with lower false positive rate.

Keywords: Network Intrusion Detection, Gravitational Search Algorithm (GSA), support vector machines (SVM) .

الملخص

يستخدم نظام كشف التسلل (IDS) للكشف عن عدة أنواع من التصرفات المريبة التي يمكن أن تنتهك نظام الحاسب الأمني وتفقده موثوقيته ، هذا يتضمن هجمات الشبكة ضد الخدمات الضعيفة ، البيانات التي تدفع الهجمات على التطبيقات ، الهجمات القائمة على استضافة الامتيازات ، الدخول غير المصرح به والوصول إلى الملفات الحساسة ، والبرامج الضارة (الفيروسات ، حصان طروادة ، والديدان) خوارزمية (SVM) Support vector machine تستخدم في تصميم أجهزة كشف التسلل داخل الشبكة ، لأنها تتميز بقدرتها وفاعليتها في تصنيف البيانات على الشبكة ، ولكن هناك صعوبة في تمثيل عوامل خوارزمية (SVM) Support vector machine مما يؤثر على دقة تصنيف بيانات نظام كشف التسلل (IDS) من أجل تحسين دقة تصنيف البيانات في نظام كشف التسلل (IDS) والتغلب على مشكلة صعوبة تمثيل عوامل خوارزمية (SVM) Support vector machine ، ثم تصميم خوارزمية مصنف هجين يجمع خوارزمية GSA و SVM لتحسين دقة المصنف SVM ، وقد أثبتت النتائج أن الخوارزمية GSA – SVM لها القدرة على تحسين أداء جهاز كاشف التسلل (IDS) وتقليل نسبة الخطأ في عملية تحليل البيانات في الشبكة .

I. Introduction

The importance to safeguard computer network against confidentiality, integrity and availability breaches is an important issue and intrusion detection plays vital role in ensuring a secured network. Security policies or firewalls have difficulty in preventing such attacks because of the hidden vulnerabilities contained in software applications. Therefore, intrusion detection system (IDS) is required as an additional wall for protecting systems despite the prevention techniques. Support vector machine (SVM) is a powerful technique for solving problems related to learning, classification and prediction. It has been extensively applied to provide potential solutions for the IDS problem (Horng et al., 2011). However, one of the primary problems is how to select the kernel function and its parameter values for SVM. This problem is a crucial step in handling a learning task with an SVM since it has a heavy impact on the classification accuracy (Ranaee et al., 2010).

II. Related Work

Peddabachigari et al. (2007), used two hybrid approaches for modelling IDS. Decision trees (DT) and support vector machines (SVM) are combined as a hierarchical hybrid intelligent system model (DT– SVM) and an ensemble approach combining the base classifiers. In this model, the training set is passed through the DT classifier to generate leaf–node information. Then, the SVM classifier is trained using the training set

together with leaf-node information (as an additional attribute) to produce the final output.

Shih et al. (2008), a particle swarm optimization-based approach, capable of searching for the optimal parameter values for SVM to obtain a subset of beneficial features. The PSO + SVM approach is applied to eliminate unnecessary or insignificant features, and effectively determine the parameter values, in turn improving the overall classification results.

Wang et al. (2010b) proposed a hybrid approach to the design of an IDS. The proposed approach combines the support vector classifier and ABC algorithm (ABC-SVM). The ABC algorithm is used to elect the C, and parameter parameters and beneficial features for the SVM. The results showed that the ABC-SVM approach achieved high accuracy rates.

Kuang et al. (2014) proposed a new intrusion detection system composed of kernel principal component analysis (KPCA) and GA with SVM. The N-KPCAGA-SVM system consists of two stages. In the first stage, KPCA is used to reduce the dataset and extract the features of the normalized data. The second stage deals with the detection classifier. The GA is used to optimize the accuracy of the SVM classifier by detecting the subset of the best values of kernel parameters for the SVM classifier. The results showed that the classification accuracy of the proposed system achieved faster convergence speed and better detection accuracy compared with a single SVM classifier.

Dastanpour et al. (2014) presented an approach for an IDS composed of the ANN algorithm and GSA optimization. The proposed system consists of two stages. In the first stage, the ANN algorithm is executed on the training dataset and the recognition results of the ANN are sent to next stage. In the second stage, the recognition results of the ANN are classified by the hybrid GSA-ANN algorithm. The KDD 99 dataset was used to evaluate the proposed system, with the results showing that the GSA-ANN hybrid approach achieved high accuracy compared with a single ANN algorithm.

Manekar and Waghmare (2014) proposed an IDS based on the machine learning technique. The proposed system consists of two machine learning algorithms: SVM and PSO. In the first step in the proposed system, the PSO algorithm is used to optimize the value of the C and parameters and important features for the SVM. In the second step, the parameters and features are used to train the SVM. The results showed that the proposed system k improved the detection accuracy compared to a single SVM classifier.

In this paper, a hybrid classifier is designed based on a combination of the GSA and SVM algorithms. The main purpose of designing the GSA-SVM classifier is to optimize the accuracy of the SVM classifier by detecting the subset of the best values of the kernel parameters for the SVM classifier.The performance of the proposed approach has been tested on

KDD CUP 99 data set, and the results have been compared with original SVM algorithm. The rest of this paper is organized as follows. Review of the support vector machines (SVM) algorithm is given in section 3.In section 4 present a brief overview of the gravitational search algorithm. Section 5 describes proposed approach. Presents Experiment results and analysis Section 6. Finally, Section 7 makes conclusions.

III. Support vector machines (SVM)

The SVM introduced by Vapnik (1998) is a technique for solving problems related to learning, classification and prediction.. The basic idea of SVM is mapping the training samples from the input space into a higher dimensional feature space via a mapping function Ø. Given a training set $S = \{(x_i, y_i) | x_i \varepsilon H, y_i \varepsilon \{\pm 1\}, i = 1, 2, \ldots, l\}$, where xi are the input vectors and y_i the labels of the xi, the target function is

$$\begin{cases} \min \ \Phi(\mathbf{w}) = \frac{1}{2}(\mathbf{w}\cdot\mathbf{w}) + C\sum_{i=1}^{l}\xi_i \\ s.t. \ \ y_i((\mathbf{w}\cdot\phi(\mathbf{x}_i)) + b) \geq 1 - \xi_i, \ \ \xi_i \geq 0 \ \ i = 1,2,\ldots,l, \end{cases} \quad (1)$$

Where C is a penalty parameter, ξ_i are non-negative slack variables. So the problem of constructing the optimal hyper plane is transformed into the following quadratic programming problem:

$$\begin{cases} \max \quad L(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ s.t. \quad \sum_{i=1}^{l} \alpha_i y_i = 0, \quad 0 \le \alpha_i \le C, \quad i = 1, 2, \ldots, l. \end{cases} \tag{2}$$

The decision function can be shown as:

$$f(\mathbf{x}) = \text{sign}\left[\sum_{i=1}^{l} y_i \alpha_i K(\mathbf{x}_i \cdot \mathbf{x}) + b\right]. \tag{3}$$

IV. Standard Gravitational Search Algorithm (GSA)

The gravitational search algorithm (GSA) is a heuristic optimization algorithm based on the gravitational interaction between masses (Rashedi *et al.*, 2009). It has the capability to discover the whole search space. The GSA avoids entrapment in a local optima by following the best results obtained for every individual object. However, in the final steps of the search, the process slows down to ensure the exploitation aspect of the solution finding. The algorithm looks to find the best possible solution. Newtonian gravity laws have been applied to the construction of the GSA. The GSA views all entities as objects with masses and these masses are attracted to each other by the force of gravity. Objects are drawn to other objects with heavier masses. Thus, due to gravitational force, heavier masses become heavier (Rashedi *et al.*, 2009).

To describe the GSA, consider a system with N masses (agents) in which the position of the ith mass is defined as follows:

$$X_i = (x_i^1, ..., x_i^d, ..., x_i^n) \quad (4)$$

The mass of each agent is calculated after computing a current population's fitness as follows:

$$M_i(t) = \frac{fit_i(t) - worst(t)}{\sum_{j=1}^{N} (fit_j(t) - worst(t))} \quad (5)$$

Where $M_i(t)$ and $fit_i(t)$ represent the mass and the fitness value of the agent i at t, respectively.

To compute the acceleration of an agent, the total forces from a set of heavier masses that act on it should be considered based on the law of gravity (Eq. (6)), followed by the calculation of an agent acceleration using a law of motion (Eq. (7)). After that, the next velocity of an agent is calculated as a fraction of its current velocity added to its acceleration (Eq. (8)). Then, its next position can be calculated using Eq. (9).

$$F_i^d(t) = \sum_{j \in kbest, j \neq i} rand_j G(t) \frac{M_j(t) M_i(t)}{R_{i,j}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \quad (6)$$

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} = \sum_{j \in kbest, j \neq i} rand_j G(t) \frac{M_j(t)}{R_{i,j}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \quad (7)$$

$$v_i^d(t+1) = Rand_i \times v_i^d(t) + a_i^d(t) \quad (8)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1)$$

(9)

V. Proposed Approach

In this paper, a hybrid classifier is designed based on a combination of the GSA and SVM algorithms. The main purpose of designing the GSA–SVM classifier is to optimize the accuracy of the SVM classifier by detecting the subset of the best values of the kernel parameters for the SVM classifier. In the GSA–SVM classifier, the GSA is introduced as an optimization technique to optimize the SVM parameters. The GSA starts with n–randomly selected agents and searches for the optimal agent iteratively. Each agent is an mdimensional vector and represents a candidate solution. The SVM classifier is built for each candidate solution to evaluate its performance through evaluation of the fitness function. The fitness function value is based on the classification accuracy of the SVM classifier. The GSA guides the selection of potential subsets that lead to the best prediction accuracy. The parameter settings used in the experiment are shown in Table 1. The GSA–SVM algorithm is illustrated in Figure 1, and the detailed steps of the algorithm are explained in the Algorithm .

Table 1: Key parameters values used in GSA−SVM

Parameter	Value/Qty	Description
N	5	Number of agents
max_it	500	Maximum number of iterations
Threshold	2.9	Based on the experiment
rand	0–1	Two uniformly distributed random numbers between 0 and 1
G0	1	Gravitational constant
ε	1	Small value to avoid division by zero
v_i^d	Variable	The velocity of *i*th agent in the *d*th dimension
a_i^d	Variable	The acceleration of the agent *i* in direction *d*th
x_i^d	Variable	The position of *i*th agent in the *d*th dimension
$R_{i,j}$	Variable	Euclidean distance between two agents *i* and *j*
F_i^d	Variable	The total force that acts on agent *i* in a dimension
ξ_i	Variable	Non-negative slack variable.
C	Variable	Penalty parameter, that control of the decision function and the number of training samples.

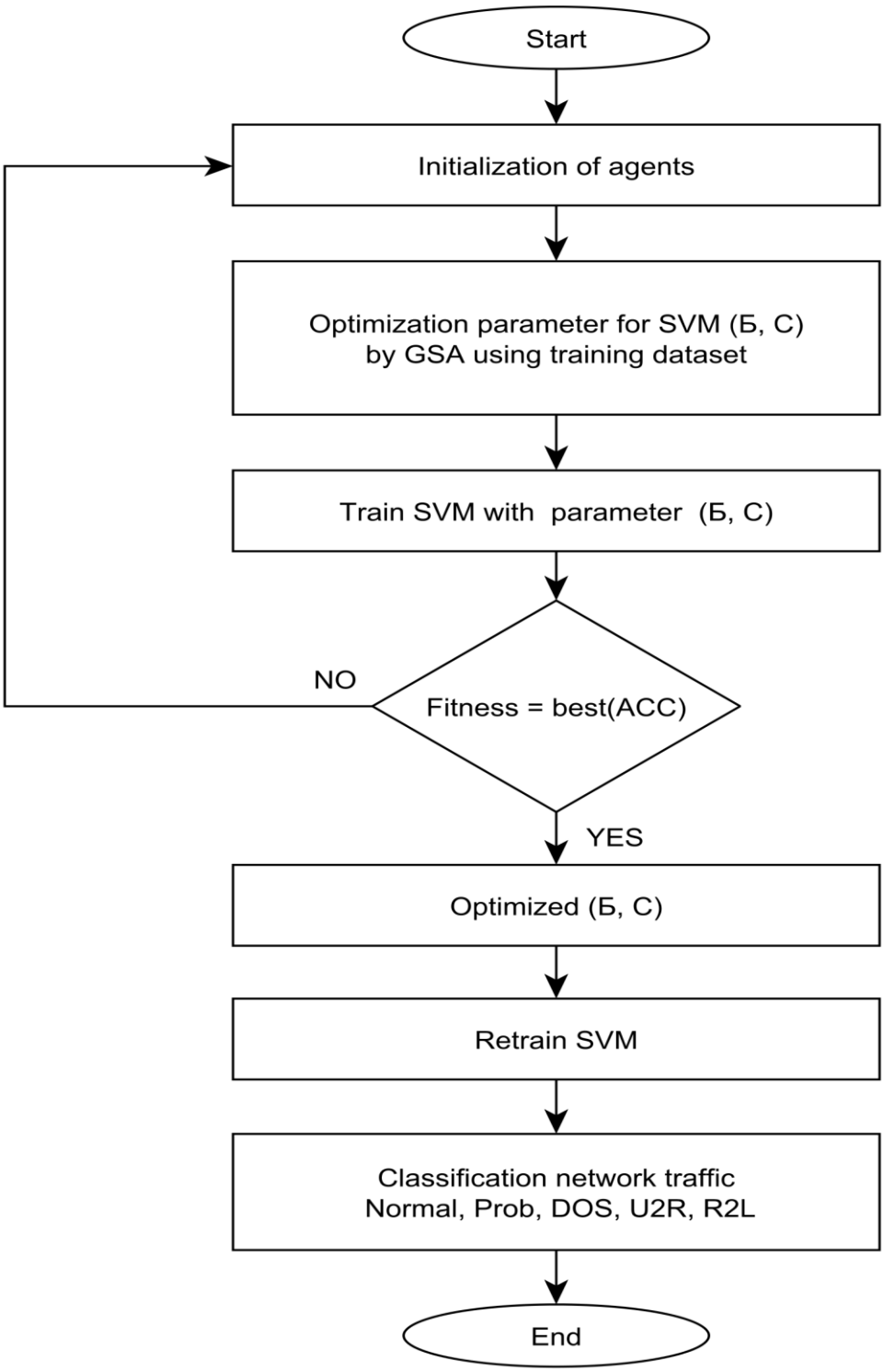
Start
Initialization of agents
Optimization parameter for SVM (Б, C)
by GSA using training dataset
Train SVM with parameter (Б, C)
Fitness = best(ACC)
NO
YES
Optimized (Б, C)
Retrain SVM
Classification network traffic
Normal, Prob, DOS, U2R, R2L
End

Figure 1: Flowchart of GSA–SVM Hybrid Algorithm

Algorithm 4 GSA-SVM

Initialize the position and velocity of agents randomly,Set the parameters of GSA-SVM (N, G0, ε, tmax)
Repeat
For each mass $i = 1, 2, ..., N do$
Train SVM
Evaluate fitness function of each agent
Calculate mass for all of the agents
Calculate force for all of the agents
Calculate acceleration for all of the agents
Update the velocity position of agents
Update the position of the agents
End For
Until: cluster centroid not change or max-iter
Retrain SVM and classification results

VI. Experiment Data

This research used the dataset KDD Cup 1999, which was collected by MIT Lincoln Lab. IT is the largest publicly available and sophisticated benchmarks for researchers to evaluate intrusion detection algorithms or machine learning algorithms. The dataset contains 4,940,000 traffic connections consisting of normal network traffics and 24 types of attacks from 4 categories of attacks which are Probe, DoS, U2R and R2L. Each connection contains 41 features available in every connection record in the dataset. This study, as most of the literature research, used 10% version of the dataset consisting of 494,020 traffic connections with similar ratio of attacks as in the full dataset. (Mukkamala et al., 2003; Tsai et al., 2009).

VII. Experiment Results and Analysis

The experiments are performed on KDD99 data set. The training data set separated into attack data sets and normal datasets. The training data set are feeding into the hybrid GSA-SVM classifier, the GSA algorithm is used to seek the optimal parameters C, σ in the SVM. Through the training process, the parameter values, and training dataset are used for building SVM classifier. Then feed the test dataset into the GSA-SVM classifier.

Table 2 summarizes the performance of the proposed GSA-SVM classifier and SVM in relation to the detection accuracy, false positive rate and detection rate. The results showed that the GSA-SVM classifier outperformed the SVM classifier in terms of detection rate and detection accuracy in all five traffic classes. The GSA-SVM classifier achieved a high detection rate and detection accuracy with an average rate of 96.85 % and 97.05 %, respectively. However, the SVM classifier achieved 90.10 % and 77.16 % for the detection rate and detection accuracy, respectively. According to the results, the GSA-SVM classifier achieved a lower false positive rate compared to the SVM classifier in all five classes (with an average rate of 0.03 %).

Table 1: Performance results for GSA-SVM and SVM

Class	SVM Classifier			GSA-SVM Classifier		
	ACC (%)	FPR (%)	DR (%)	ACC (%)	FPR (%)	DR (%)
Normal	81.64	0.19	49.97	98.52	0.03	99.99
Prob	93.40	0.11	96.67	95.22	0.09	100
DoS	95.29	0.00	84.85	98.28	0	94.74
U2R	40.55	0.41	26.30	46.26	0.41	46.3
R2L	73.31	0.19	22.25	96.17	0	92.67
AVG	**90.10**	**0.1**	**77.16**	**97.04**	**0.03**	**96.85**

Legend:
In bracket is %; ACC=Detection accuracy, FP=False positive rate; DR=Detection rate
AVG: this average excluded the up normal value

Figure 2 and Figure 3 illustrate the comparison in terms of overall detection accuracy and false positive rate for the GSASVM classifier and the SVM classifier. The results on the detection accuracy (Figure 2 showed that the the GSA-SVM classifier achieved the highest accuracy. The detection accuracy for the the GSA-SVM classifier improved by 6.95% as compared to the SVM classifier. The results on the false positive rate (Figure 3) showed that the GSA-SVM classifer achieved the lowest false positive rate. The false positive rate for the GSA-SVM classifier reduced by 0.07 % compared to the SVM classifier. The results, as presented in the figures, showed that the GSA-SVM classifier outperformed the SVM classifier in terms of detection accuracy and false positive rate because it included the GSA as an optimization technique to optimize the SVM parameters. The

results showed that the detection effectiveness was improved by optimizing the accuracy of the SVM classifier to enhance the classification process.

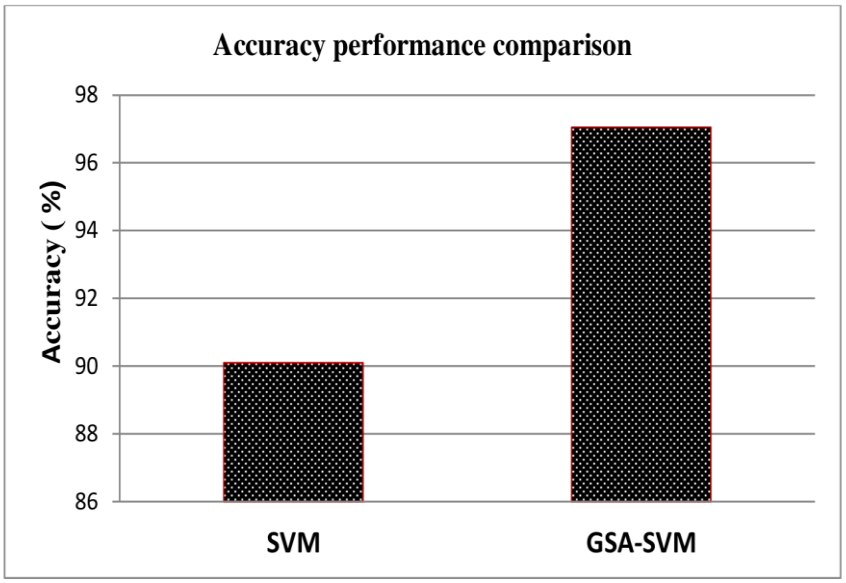

Figure 2: Detection accuracy performance of GSA-based AIDS with different classifiers

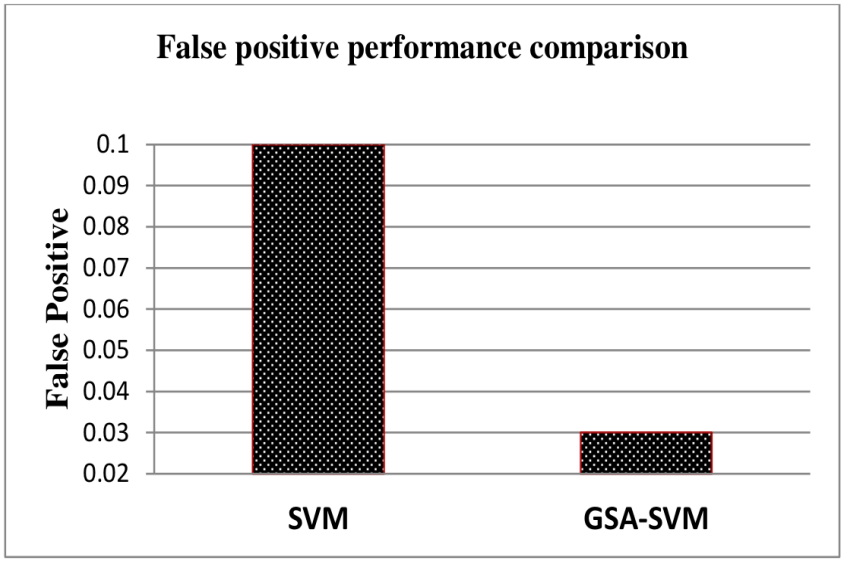

Figure 3: False positive performance of GSA-based AIDS with different classifiers

VIII. . Conclusion

In this paper, propose a new hybrid GSA-SVM classifier was designed to enhance the classification process of the detection classifier. In the GSA-SVM classifier, the GSA is used to optimize the accuracy of the SVM classifier by detecting the subset of the best values of kernel parameters for the SVM classifier. The GSA avoids being trapped in the local optima and, by following the best results obtained by every individual object, obtains accurate results. Moreover, it has the capability to optimize and improve the performance of a classification classifier. In the experiments, the detection accuracy improved by 6.95 % while the false positive rate reduced by 0.07 % when using the GSA-SVM classifier. In addition, the

results showed an improvement in the U2R class and R2L class. This occurs because the difficulty of correctly detecting the imbalanced dataset is reduced by optimizing the accuracy of the SVM classifier. Thus, the detection effectiveness is improved when the GSASVM classifier implements the GSA to optimize the kernel function parameters for the SVM classifier.

References

Dastanpour, A., Ibrahim, S., Mashinchi, R. and Selamat, A. (2014). Using Gravitational Search Algorithm to Support Artificial Neural Network in Intrusion Detection System. *SmartCR*. 4(6), 426–434.

Hongying Zheng .(2011) .An Efficient Hybrid Clustering-PSO Algorithm for Anomaly Intrusion Detection, JOURNAL OF SOFTWARE, VOL. 6(12).306-313.

Kuang, F., Xu, W. and Zhang, S. (2014). A novel hybrid KPCA and SVM with GA model for intrusion detection. Applied Soft Computing. 18, 178–184.

Manekar, V. and Waghmare, K. (2014). Intrusion Detection System using Support Vector Machine (SVM) and Particle Swarm Optimization (PSO). International Journal of Advanced Computer Research. 4(3), 25–30.

Mukkamala, S., Sung A. and Abraham, A. (2003). Intrusion Detection Using Ensemble of Soft Computing Paradigms. Proceedings of 3rd. International Conference on Intelligent Systems Design and Applications. 239-248.

Rashedi,E., Nezamabadi,H.and Saryazdi, S.(2009). GSA: A gravitational search algorithm", Information Sciences, VOL. 179. 2232-2248.

S. Peddabachigari, A. Abraham, C. Grosan, J. (2007). Thomas: Modeling intrusion detection system using hybrid intelligent systems. In Journal of Network and Computer Applications, 30. 114-132 .

Shih.W , Kuo.C , Shih.C and Zne. L. (2008).Particle swarm optimization for parameter determination and feature selection of support vector machines, Expert Systems with Applications. 1817–1824.

Tsai, C., Hsu, Y., Lin, C. and Lin, W. (2009). Intrusion Detection by Machine Learning: A Review. Expert Systems with Applications. 36(10).11994–12000.

V. Ranaee, A. Ebrahimizadeh, R. Ghaderi. (2010). Application of the PSO–SVM model for recognition of control chart patterns, ISA Transactions, VOL 49 (4) (2010) .577–586.

Vapnik, V. (1998) .Statistical learning theory. Wiley, New York.

Wang, J., Li, T. and Ren, R. (2010b). A real Time IDS Based on Artificial Bee Colony-Support Vector Machine Algorithm. In *The Third InternationalWorkshop* 133 *on Advanced Computational Intelligence (IWACI).* IEEE, 91–96.