ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.18, No.6, June 2007, pp.1369−1378
DOI: 10.1360/jos181369

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于改进多目标遗传算法的入侵检测集成方法*

俞 研[1,2+], 黄 皓[1,2]

[1](南京大学 计算机科学与技术系,江苏 南京 210093)

[2](计算机软件新技术国家重点实验室(南京大学),江苏 南京 210093)

An Ensemble Approach to Intrusion Detection Based on Improved Multi-objective Genetic Algorithm

YU Yan[1,2+], HUANG Hao[1,2]

[1](Department of Computer Science and Technology, Nanjing University, Nanjing 210093, China)

[2](State Key Laboratory for Novel Software Technology(Nanjing University), Nanjing 210093, China)

+ Corresponding author: Phn: +86-25-83202540, E-mail: yuyann@hotmail.com

Yu Y, Huang H. An ensemble approach to intrusion detection based on improved multi-objective genetic algorithm. *Journal of Software*, 2007,18(6):1369−1378. http://www.jos.org.cn/1000-9825/18/1369.htm

Abstract: There exist some issues in current intrusion detection algorithms such as unbalanced detection performance on different types of attacks, and redundant or useless features that will lead to the complexity of detection model and degradation of detection accuracy. This paper presents an ensemble approach to intrusion detection based on improved multi-objective genetic algorithm. The algorithm generates the optimal feature subsets, which achieve the best trade-off between detection rate and false positive rate through an improved MOGA. And the most accurate and diverse base classifiers are selected to constitute the ensemble intrusion detection model by selective ensemble approach. The experimental results show that the algorithm can solve the feature selection problem of intrusion detection effectively. It can also achieve balanced detection performance on different types of attacks while maintaining high detection accuracy.

Key words: intrusion detection; feature selection; optimization; multi-objective genetic algorithm; selective ensemble

摘 要: 针对现有入侵检测算法中存在着对不同类型攻击检测的不均衡性以及冗余或无用特征导致的检测模型复杂与检测精度下降的问题,提出了一种基于改进多目标遗传算法的入侵检测集成方法.利用改进的多目标遗传算法生成检测率与误报率均衡优化的最优特征子集的集合,并采用选择性集成方法挑选精确的、具有多样性的基分类器构造集成入侵检测模型.实验结果表明,该算法能够有效地解决入侵检测中存在的特征选择问题,并在保证较高检测精度的基础上,对不同类型的攻击检测具有良好的均衡性.

关键词: 入侵检测;特征选择;优化;多目标遗传算法;选择性集成

* Supported by the National Natural Science Foundation of China under Grant No.60303023 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2003AA142010 (国家高技术研究发展计划(863)); the High-Tech Research Plan of Jiangsu Province of China under Grant No.BG2004030 (江苏省高技术计划)

Received 2006-06-18; Accepted 2006-12-06

中图法分类号: TP18 文献标识码: A

1 Introduction

Along with the popularization of Internet, more and more attacks to the computer systems and networks emerge. Accordingly, Intrusion Detection Systems (IDS) have become important tools for ensuring network security. Intrusion detection is based on the assumption that intrusion activities are noticeably different from normal system activities and thus detectable. In order to detect intrusion activities, many soft computing techniques, such as Neural Networks and SVMs, etc.[1,2] have been widely used by the IDS community due to their generalization capabilities that help in detecting known and unknown intrusions. But there are still some drawbacks, such as low detection accuracy, unbalanced performance on different attack types and long response time.

Recently, ensemble classifier has been used to improve the detection accuracy in intrusion detection area[3,4]. It has been demonstrated that an ensemble of classifiers is more accurate than any of its members if the classifiers in the ensemble are both accurate and diverse[5]. Varying the feature subsets used by each member can help to promote the diversity[6−8]. Feature selection through multi-objective genetic algorithms (MOGAs) is a very powerful tool for finding a set of good classifiers[9−11]. MOGAs can be used as a means to search for subsets of features, which contain discriminatory information to perform the classification of intrusions. Consequently, these feature subsets are Pareto-optimal solutions that can achieve the best trade-off between several objectives, e.g. detection rate (DR) and false positive rate (FPR) in intrusion detection. However, it is impractical to find the true Pareto-optimal solutions of combinatorial optimization problems. In this case, one promising approach is to improve the local search ability of MOGAs to try to drive populations to true Pareto-optimal solutions as close as possible for obtaining a variety of near Pareto-optimal solutions. After optimal feature subsets are generated, ensemble classifier can be constituted. It is worth mentioning that ensembling many of the available components may be better than ensembling all of them[12]. That means selective ensemble will also improve the classification accuracy.

In this paper, we present an ensemble approach to intrusion detection based on improved multi-objective genetic algorithm, E-IMOGA. Firstly, E-IMOGA algorithm generates near Pareto-optimal solutions, i.e. feature subsets, through an improved MOGA which makes them converge to the true Pareto-optimal front better. After that, the most accurate and diverse members, which are trained by corresponding feature subsets, are selected to constitute ensemble intrusion detection model. The experimental results show that the model not only improves the detection accuracy, but also achieves the balanced detection performance on different types of attacks.

2 Related Work

Ensemble of classifiers for intrusion detection demands for the accuracy and diversity of base classifiers. Many researchers have focused on both of the issues. Mukkamala, *et al*.[3] used the ensemble of SVMs, MARs and ANNs with different training functions to achieve better classification accuracies. The majority voting approach was used to build ensemble classifier whose base classifiers were learned from the above algorithms respectively. The diversity of different learning algorithms was utilized. Chebrolu, *et al*.[4] investigated the performance of two feature selection algorithms involving Bayesian Networks (BN) and Classification and Regression Trees (CART) respectively. Then, the ensemble of BN and CART was built. Both of the approaches considered the diversity of different learning algorithms for intrusion detection. So, the detection performance was promoted accordingly.

Feature selection is another method to promote the accuracy and diversity. Optiz[6] presented a genetic algorithm (GA) approach for searching for an appropriate set of feature subsets for ensembles. Using neural

networks as the classifier, results showed better than the ensemble approaches of Bagging and Boosting. Tsymbal, *et al*.[7] presented an algorithm for building ensembles of simple Bayesian classifiers by using different feature subsets generated with the random subspace method. In this case, the ensemble consists of multiple classifiers constructed by randomly selecting feature subsets, that is, classifiers constructed in randomly chosen subspaces. Guerra-Salcedo, *et al*.[8] used a genetic search approach to find subsets of features that could be suitable for ensemble creation. Comparing to the idea of randomly selecting subspaces to construct ensembles of table-based classifiers, the method showed better performance and usage of the storage space.

Since feature selection for intrusion detection can be treated as the issue of multi-objective optimization, MOGAs are used to search optimal feature subsets. Oliveira, *et al*.[9] used a MOGA to search for subsets of features, which contained discriminatory information to perform the classification of handwritten digits. The strategy took into account an efficient MOGA to generate a set of alternative solutions and the use of a cross-validation method to indicate the best accuracy/complexity trade-off. The classification accuracy was supplied by neural networks and sensitivity analysis. Radtke, *et al*.[10] presented a two-level approach to create ensemble of classifiers based on intelligent feature extraction and MOGA for recognizing isolated handwritten symbols. The first stage optimized a set of representations, which was used to create classifiers. Then the second stage optimized the base classifiers.

The rest of the paper is organized as follows. Section 3 briefly introduces MOGAs for feature selection, and gives its improved variation by combining with sequential feature selection strategy. Section 4 describes our E-IMOGA algorithm. In Section 5, we present the experimental results and analysis of using E-IMOGA algorithm in intrusion detection. The paper concludes with Section 6.

3 Feature Selection and Improved Multi-Objective Genetic Algorithm

3.1 Feature selection

Since the amount of network data that an IDS needs to examine is very large, analysis is difficult because extraneous features can make it harder to detect suspicious behavior patterns[13]. Irrelevant and redundant features may lead to complex intrusion detection model as well as poor detection accuracy. At the same time, it is well known that ensemble approaches need accurate and diverse members in order to achieve higher accuracy. Therefore, building intrusion detection model based on all features is infeasible, and feature selection becomes indispensable.

Feature selection problem involves the selection of a subset of d features from a total of D original features, based on a given optimization criterion. Formally, without loss of generality, minimizing the criteria function, the problem of feature selection is to find a subset $X \subseteq S$ such that

$$J(X) = \min_{Y \subseteq S} J(Y) \tag{1}$$

where $X \subseteq S$ is the optimal feature subset, S is the original feature set, and $J(\cdot)$ is the feature selection criterion function.

Obviously, the choice for assessing the performance of an intrusion detection model is to estimate its DR and FPR. It is preferable to maximize DR and minimize FPR. For DR, we can modify the objective function to 1-DR for minimizing it. Accordingly, the problem of feature selection becomes to find the subset $S^* \subseteq S$ such that

$$S^* = \arg\min_{X \subseteq S}\{\boldsymbol{J}(X)\} = \arg\min_{X \subseteq S}\{[J_{1-DR}(X), J_{FPR}(X)]^T\} \tag{2}$$

where S is the original feature set. The objective function $\boldsymbol{J}(\cdot)$ is a vector function consisting of two objectives, namely minimizing 1-DR and FPR.

In order to evaluate the results of intrusion detection with different feature subsets, we employ SVM as the learner because of its speed and scalability[3].

Finding a useful feature subset is a form of search. Ideally, feature selection methods search through the subsets of features, and try to find the best one among the competing 2^N candidate subsets according to some evaluation functions. However this procedure is exhaustive as it tries to find only the best one. It may be too costly and practically prohibitive. Meanwhile, when we use ensemble approach to constitute intrusion detection model, we need a set of accurate and diverse feature subsets to learn to generate various components of ensemble.

Usually, there are several search strategies for feature selection[14]. Among the categories of feature selection algorithms, genetic algorithm (GA) is naturally applicable to feature selection since the problem has an exponential search space[15,16]. In most cases, the aim is to optimize a single criterion. However, for intrusion detection, DR and FPR should be considered simultaneously. So feature selection naturally poses as a multi-objective search problem. Therefore, MOGAs are well suited for feature selection in ensemble approach to intrusion detection.

3.2 Multi-Objective genetic algorithms

Since the problem of feature selection for intrusion detection requires the simultaneous optimization of several conflicting objectives, e.g. DR and FPR, the solution is usually computed by combining them into a single objective according to some utility function. However, the utility function is not a prior to the optimization process. So, the problem of feature selection for intrusion detection should be treated as a multi-objective optimization problem.

Assuming, without loss of generality, a minimization problem, a multi-objective optimization problem can be defined formally as follows:

Given a n-dimensional decision vector $\mathbf{x}=\{x_1,\ldots,x_n\}$ in the solution space X, find a vector $\mathbf{x}^*$ that minimizes a give set of k objective functions $f(\mathbf{x}^*)=\{f_1(\mathbf{x}^*),\ldots,f_k(\mathbf{x}^*)\}$. The solution space X is generally restricted by a series of constraints, such as $g_j(\mathbf{x}^*)=b_j$ for $j=1,\ldots,m$.

The goal of multi-objective optimization is to find the solutions giving the best trade-off between multiple objectives, known as Pareto optimum. Several concepts are mathematically defined as follows:

Definition 1 (**Dominance**/**Inferiority**). A vector $\mathbf{u}=(u_1,\ldots,u_n)$ is said to dominate to $\mathbf{v}=(v_1,\ldots,v_n)$ iff $\mathbf{u}$ is partially less than $\mathbf{v}$, i.e., $\forall i=1,\ldots,n: u_i\leq v_i \wedge \exists i=1,\ldots,n: u_i<v_i$.

Definition 2 (**Pareto optimal**). A solution $\mathbf{x}_u\in X$ is said to Pareto optimal iff there is no $\boldsymbol{x}_v\in X$ for which $\mathbf{v}=f(\mathbf{x}_v)=(v_1,v_2,\ldots,v_n)$ dominates $\mathbf{u}=f(\mathbf{x}_u)=(u_1,u_2,\ldots,u_n)$.

The set of all non-dominated solutions in X is referred to the Pareto optimal set. For a given Pareto optimal set, the corresponding objective function values in the objective space are non-dominated, and called the Pareto front.

There are several well-known MOGAs[11,17,18]. NSGA-II[17] is one of the most popular algorithms. NSGA-II was proposed as an improvement of NSGA[19]. The idea behind NSGA-II is that a Pareto ranking based selection method is used to emphasize good points and a niche method is used to maintain stable subpopulations of good points. At the same time, in order to avoid high computational complexity of non-dominated sorting and need for the sharing parameter existing in NSGA, NSGA-II also uses a fast non-dominated sorting approach, an elitist-preserving strategy, and a parameter-less niching operator.

In NSGA-II, before the selection is performed, the population is ranked on the basis of an individual's non-domination. The non-dominated individuals present in the population are first identified from the current population. Then, all these individuals are assumed to constitute the first non-dominated front in the population and assigned a dummy fitness value. The same fitness value is assigned to give an equal reproductive potential to all these non-dominated individuals. In order to preserve diversity in the population, a crowded comparison approach is used to guide the selection process towards a uniformly spread-out Pareto-optimal front. Formally,

$$i \prec_n j \text{ if } ((i_{rank} < j_{rank}) \text{ or } ((i_{rank} = j_{rank}) \text{ and } (i_{distance} > j_{distance})) \quad (3)$$

where $\prec_n$ is crowed comparison operator, i_{rank} and j_{rank} refer to non-domination ranks, and $i_{distance}$ and $j_{distance}$ refer

to crowding distances. The more details about NSGA-II can be found in Ref.[17].

3.3 Our improved MOGA algorithm

As was described above, MOGAs fit for feature selection in ensemble approach. However, we find that solutions provided by a MOGA are likely to be inferior or only comparable to classical heuristic search algorithms in feature selection. Although MOGAs are able to escape from local optima by means of the crossover and mutation operator, they are weak in fine-tuning near local optimum points and disabled to find a perfect solution because of "premature convergence". This makes the obtained solutions be not as close as possible and uniformly spread-out towards the true Pareto-optimal front. To improve the search capability of MOGAs, we present an improved MOGA to enhance the local search capability of MOGA by sequential search strategy.

There are many sequential search strategies for feature selection[20]. SFFS and its backward counterpart, SBFS, are popular methods[21]. They take use of backtracking and are capable of "correcting wrong inclusion/removal decisions" until the quality of the current set of selected features cannot be improved by including or removing another feature. Because SFFS is considered as the best amongst the sequential search algorithms for feature selection, and can find fairly good solutions in moderate time, we employ the similar search strategy to trace the adjacent subsets of the Pareto-optimal solutions that are generated by MOGA.

In view of the first non-dominated front obtained by MOGA includes solutions that achieve the best trade-off between DR and FPR, they are regarded as the start points of sequential search algorithm. When sequential search is applied, the best neighbor of the current solution is selected with respect to a single weighted objective function. If the neighbor is superior to the current solution, the current solution is immediately replaced with the neighbor.

Since a sequential search strategy requires single objective function, a weighted objective function can be used

$$f(\boldsymbol{x}) = w_1^x f_1(\boldsymbol{x}) + w_2^x f_2(\boldsymbol{x}) + ... + w_n^x f_n(\boldsymbol{x}) \tag{4}$$

where w_i^x is the dummy weight for the i-th objective $f_i(\mathbf{x})$ for the solution $\mathbf{x}$, and can be defined as

$$w_i^x = \frac{\dfrac{f_i^{\max} - f_i(\boldsymbol{x})}{f_i^{\max} - f_i^{\min}}}{\displaystyle\sum_{j=1}^{n} \frac{f_j^{\max} - f_j(\boldsymbol{x})}{f_j^{\max} - f_j^{\min}}}, i = 1,2,...,n \tag{5}$$

where $f_i^{\max}$ and $f_i^{\min}$ are the maximum and minimum values of the i-th objective $f_i(\mathbf{x})$ in the current population, respectively, and $\sum_{i=1}^{n} w_i^x = 1$.

Once the dummy weights are calculated, the sequential search from each solution $\mathbf{x}$ independently with the purpose of optimizing $f(\mathbf{x})$ begins. Since the dummy weight w_i^x dictates roughly the priority of different objective functions at solution $\mathbf{x}$, optimizing $f(\mathbf{x})$ will produce a Pareto-optimal or a near Pareto-optimal solution.

4 E-IMOGA Algorithm

4.1 Selective ensemble approach

Now, we can constitute our ensemble intrusion detection model using the optimal feature subsets which are generated by our improved MOGA. However, according to Ref.[12], ensembling many of neural networks may be better than ensembling them all. That is the basis that we devise our selective ensemble approach on.

So, once the set of classifiers have been trained, our task is to pick the members of ensemble which are the most diverse and accurate. Let's look at GASEN algorithm which was proposed to build the selective ensemble[12]. GASEN assigns a random weight to each of the available learners at first. Then it employs genetic algorithm to

evolve those weights so that they can characterize to some extent the fitness of the learners in joining the ensemble. Finally it selects the learners whose weights are bigger than a pre-set threshold λ to make up the ensemble.

Each individual in the evolving population is a weight vector $\mathbf{w}=(w_1,w_2,\ldots,w_t)$, where w_i is the weight for the i-th component classifier according to Pareto-optimal solution, i.e. feature subset. In order to evaluate the goodness of the classifiers, a separate validation dataset is used. Let $\hat{E}_w$ denote the estimated generalization error of the ensemble corresponding to the individual $\mathbf{w}$ on the validation set. It is obvious that $\hat{E}_w$ can express the goodness of $\mathbf{w}$, i.e. the smaller $\hat{E}_w$ is, the better $\mathbf{w}$ is. So, $f(\boldsymbol{w})=1/\hat{E}_w$ can be used as the fitness function naturally.

In our selective ensemble approach, a weight with 0 or 1 is assigned to each individual. That means if w_i is 1, then the i-th classifier is selected as a member of ensemble, otherwise is not. So, the weight $\mathbf{w}$ is a vector of {0,1}. And manually setting the threshold for selecting component learners according to their evolved weights is not needed. E-IMOGA algorithm is represented in the next subsection in detail.

Algorithm Representation

The main procedure of E-IMOGA algorithm is described as follows.

Algorithm 1 (**E-IMOGA**).

Input: training set, validation set, learner L, and original feature set U;

Output: Ensemble intrusion detection model $N^*(\mathbf{x})$.

Procedure:

1. Generate randomly an initial population P_0, t=0;
2. Create a children population Q_0 of size n;
3. Combine parent and children population $R_t=P_t\cup Q_t$;
4. Generate all non-dominated fronts $F=(F_1,F_2,\ldots)$ of R_t;
5. Sort the non-dominated fronts using $\prec_n$;
6. Choose the best solutions needed to fill the population;
7. Use selection, crossover and mutation to create a new population Q_{t+1}, $t=t+1$;
8. If the maximum number of generations is not reached, go to (3);
9. Begin the sequential search procedure, $Z=\varnothing$;
10. Put features of 1-bits in $\mathbf{r}\in F_1$ into X, $F_1=F_1\backslash\{\mathbf{r}\}$, where $\mathbf{r}$ is a solution in F_1;
11. Let $X^*=X$;
12. Find the most significant feature $\mathbf{x}^+\in U\backslash X$, and include it into X, i.e. $x^+=\arg\min_{x\in U\backslash X} f(X\cup\{x\})$, $X=X\cup\{x^+\}$, where
$$f(\boldsymbol{x})=w_{dr}^{\boldsymbol{x}}f_{dr}(\boldsymbol{x})+w_{fpr}^{\boldsymbol{x}}f_{fpr}(\boldsymbol{x}),$$
$$w_{dr}^{\boldsymbol{x}}=\frac{\dfrac{f_{dr}^{\max}-f_{dr}(\boldsymbol{x})}{f_{dr}^{\max}-f_{dr}^{\min}}}{\left(\dfrac{f_{dr}^{\max}-f_{dr}(\boldsymbol{x})}{f_{dr}^{\max}-f_{dr}^{\min}}+\dfrac{f_{fpr}^{\max}-f_{fpr}(\boldsymbol{x})}{f_{fpr}^{\max}-f_{fpr}^{\min}}\right)},$$
$$w_{fpr}^{\boldsymbol{x}}=\frac{\dfrac{f_{fpr}^{\max}-f_{fpr}(\boldsymbol{x})}{f_{fpr}^{\max}-f_{fpr}^{\min}}}{\left(\dfrac{f_{dr}^{\max}-f_{dr}(\boldsymbol{x})}{f_{dr}^{\max}-f_{dr}^{\min}}+\dfrac{f_{fpr}^{\max}-f_{fpr}(\boldsymbol{x})}{f_{fpr}^{\max}-f_{fpr}^{\min}}\right)};$$
13. Find the least significant feature $x^-\in X$, i.e. $x^-=\arg\min_{x\in X} f(X\backslash\{x\})$;
14. If $f(X\backslash\{x^-\})\le f(X)$, then remove it, $X=X\backslash\{x^-\}$, go to (13);

15. If $X^{*}\neq X$, go to (12);
16. Reconstruct the individual **p** according to feature subset X, and add it into Z, i.e. $Z=Z\cup\{\mathbf{p}\}$;
17. If $F_1\neq\varnothing$, go to (10);
18. Begin selective ensemble procedure, generate a population of weight vectors with values 0 or 1 randomly;
19. Evolve the population, the fitness of a weight vector **w** is measured as $f(\boldsymbol{w})=\frac{1}{m}\sum_{(\boldsymbol{x}_i,y_i)\in S:N(\boldsymbol{x}_i)\neq y_i}1$, where $N(\boldsymbol{x}_i)=\arg\max_{y\in Y}\sum_{w_t=1:N^{(t)}(\boldsymbol{x}_i)=y}1$, S is the validate set, and m is the size of S, $f(\mathbf{w})$ denotes the number of misclassified samples, t is the cardinality of weight vector **w**;
20. **w***=the evolved best weight vector,

$$N^{*}(\boldsymbol{x})=\arg\max_{y\in Y}\sum_{w_t^{*}=1:N^{(t)}(\boldsymbol{x}_i)=y}1 .$$

5 Experimental Results and Analysis

5.1 Dataset

Experiments have been carried out on a subset of the dataset created by DARPA for the 1998 Intrusion Detection Evaluation Program[22]. It consists of 7 weeks of labeled network-based attacks inserted in the normal background data. Each connection is represented with a 41 dimensional feature vector. Connections are also labeled as belonging to one out of five classes as follows: (1) Normal traffic; (2) DoS (denial of service); (3) Probe, surveillance and probing; (4) R2L, unauthorized access from a remote machine; (5) U2R, unauthorized access to local superuser privileges by a local unprivileged user.

In order to evaluate the performance of E-IMOGA algorithm, the dataset is generated randomly from a dataset in KDD[23], which contains 492 000 records. The training and test dataset comprises of 5 914 and 6 781 records, respectively. Then, a separate validation set which contains 10 158 records is generated. All IDS models are trained and tested with the same set of data.

5.2 Performance measures

To evaluate E-IMOGA for intrusion detection, we are interested in two major indicators of performance: DR and FPR. DR is defined as the number of intrusion instances detected by the system divided by the total number of intrusion instances present in the dataset. FPR is defined as the total number of normal instances.

5.3 Experiment and analysis on improved MOGA

Firstly, we use our improved MOGA and standard MOGA to perform feature selection respectively, and compare their results. In the experiment, we use NGSA-II as a standard MOGA, and base on it to devise our improved MOGA algorithm. We run NGSA-II and our improved MOGA ten times on the same training and test dataset, respectively. Then the average results are calculated. Figure 1 reports the comparison of Pareto-optimal fronts between NSGA-II and our improved MOGA.

As represented in Fig.1, we can improve the trade-off between DR and FPR through local sequential search, especially in the case of U2R. For DoS, the performance is almost unchanged because both of the algorithms can obtain very high accuracy closing to 100 percent. For U2R, the accuracy is promoted nearly 25 percent. For Probe and R2L, although the promotions of DR are not very great, they get much lower FPRs, respectively. That means our improved algorithm can find the local optimal points that are omitted by MOGA because of its weakness of

local search capability. So, the results of our algorithm can drive the optimal solutions to converge and spread out uniformly to the true Pareto-optimal front closer. Consequently, more accurate and diverse base classifiers, which will be used to constitute the ensemble intrusion detection model, can be obtained. On the other hand, the ensemble constituted by such classifiers can achieve higher accuracy.

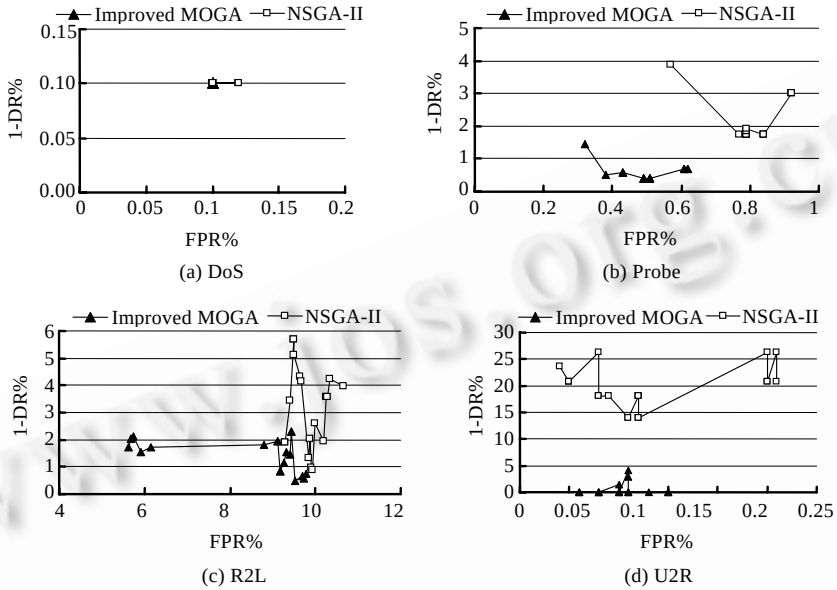

Fig.1 The Pareto-optimal fronts produced by NSGA-II and improved MOGA

5.4 Experiment and analysis on E-IMOGA

As was described above, after optimal feature subsets are selected, which will be used to build base classifiers, some accurate and diverse classifiers should be chosen to constitute ensemble. E-IMOGA algorithm begins with a population of weight vectors, which is generated randomly, to evolve. Each individual indicates which base classifier is selected as a member of ensemble. Finally, the best individual in population is selected to indicate that which classifiers will be used to constitute the ensemble. Here, we use a separate validation dataset. The performance comparison between ensembling all of the classifiers and our selective ensemble approach is listed in Table 1.

Table 1 Performance of both approaches for 4 attack classes

Attack type	Ensemble with all (%)		E-IMOGA (%)	
	DR	FPR	DR	FPR
DoS	99.95	0.04	99.98	0.03
Probe	98.79	0.44	98.96	0.38
R2L	98.76	10.02	98.51	8.91
U2R	99.93	0.14	99.95	0.11

We can notice from Table 1 that almost all of the results of E-IMOGA are superior to those of ensemble with all except DR of R2L attack. It manifests that removal of insignificant or useless classifiers in ensemble leads to the improvement of accuracy. At the same time, comparing with those of Fig.1, we also conclude that ensemble approach results in better accuracy than single classifiers.

Next, we also compare the intrusion detection performance among Wenke Lee's approach, BP neural networks, SVM, Bayesian network, ensemble of Bayesian and CART, and E-IMOGA on KDD Cup dataset[3,4,13]. The comparative results are summarized in Table 2.

From Table 2, it is noted that E-IMOGA can achieve much better than, at least comparable to other approaches. Besides the high detection accuracy, the other important merit of E-IMOGA is its balance performance on all four attack types. On the other hand, the other five approaches are difficult to detect all four attack types simultaneously, e.g. R2L and U2R of Wenke Lee's, U2R of other approaches. With high accuracy and the reduced feature space which lead to a simplified intrusion detection model and balance performance, our E-IMOGA algorithm can be more efficient and effective to detect the intrusion behavior.

Table 2 Comparison of several approaches

Attack type	Wenke Lee's (%)	BP network (%)	SVM (%)	Bayesian (%)	Ensemble (%)	E-IMOGA (%)
DoS	79.9	92.71	99.45	99.69	99.93	99.98
Probe	97.0	97.47	98.57	99.43	99.85	98.96
R2L	60.0	95.73	97.33	99.11	99.47	98.51
U2R	75.0	48.0	64.0	64.0	72.0	99.95

6 Conclusions and Future Works

This paper presents an ensemble approach to intrusion detection based on the improved multi-objective genetic algorithm, E-IMOGA. The algorithm differs from the existing algorithms as it uses an improved MOGA to generate a set of different feature subset, which is then used to build accurate and diverse base classifiers. While the ensemble is constituted, a selective ensemble approach is used to improve the accuracy of ensemble intrusion detection model. The experimental results manifest that E-IMOGA algorithm can achieve high accuracy in both DR and FPR, as well as balanced performance on all four types of attacks.

The future works focus on applying the domain knowledge of security to improve the detection accuracy, especially R2L attack, and visualizing the procedure and result of intrusion detection to promote the understandability of intrusion detection.

Acknowledgement Our special thanks to Dr. Kalyanmoy Deb at Indian Institute of Technology Kanpur for the provision of the source code of NSGA-II algorithm.

References:

[1] Cannady J. Artificial neural networks for misuse detection. In: Proc. of the '98 National Information System Security Conf. (NISSC'98). Arlington: Virginia Press, 1998. 443−456.

[2] Shon T, Seo J, Moon J. SVM approach with a genetic algorithm for network intrusion detection. In: Proc. of the 20th Int'l Symp. on Computer and Information Sciences (ISCIS 2005). Berlin: Springer-Verlag, 2005. 224−233.

[3] Mukkamala S, Sung AH, Abraham A. Intrusion detection using an ensemble of intelligent paradigms. Journal of Network and Computer Applications, 2005,28(2):167−182.

[4] Chebrolu S, Abraham A, Thomas JP. Feature deduction and ensemble design of intrusion detection systems. Computer & Security, 2004,24(4):295−307.

[5] Hansen LK, Salamon P. Neural network ensembles. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1990,12(10): 993−1001.

[6] Opitz DW. Feature selection for ensembles. In: Proc. of the 16th National Conf. on Artificial Intelligence (AAAI). Orlando: AAAI Press, 1999. 379−384.

[7] Tsymbal A, Puuronen S, Patterson DW. Ensemble feature selection with the simple Bayesian classification. Information Fusion, 2003,4(2):87−100.

[8] Guerra-Salcedo C, Whitley D. Genetic approach to feature selection for ensemble creation. In: Proc. of the Genetic and Evolutionary Computation Conf. Orlando: Morgan Kaufmann Publishers, 1999. 236−243.

[9] Oliveira LS, Sabourin R, Bortolozzi RF, Suen CY. Feature selection using multi-objective genetic algorithms for handwritten digit recognition. In: Proc. of the 16th Int'l Conf. on Pattern Recognition (ICPR 2002). Quebec: IEEE Computer Society, 2002. 568−571.

[10] Radtke PVW, Sabourin R, Wong T. Intelligent feature extraction for ensemble of classifiers. In: Proc. of the 8th Int'l Conf. on Document Analysis and Recognition (ICDAR 2005). Seoul: IEEE Computer Society, 2005. 866−870.

[11] Fonseca CM, Fleming PJ. Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In: Proc. of the 5th Int'l Conf. on Genetic Algorithms (ICGA'93). San Mateo: Morgan Kaufmann Publishers, 1993. 416−423.

[12] Zhou ZH, Wu JX, Tang W. Ensemble neural networks: Many could be better than all. Artificial Intelligence, 2002,137(1-2): 239−263.

[13] Lee WK, Stolfo SJ, Mok KW. A data mining framework for building intrusion detection models. In: Proc. of the '99 IEEE Symp. on Security and Privacy. Oakland: IEEE Computer Society, 1999. 120−132.

[14] Jain A, Zongker D. Feature selection: Evaluation, application, and small sample performance. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1997,19(2):153−158.

[15] Yang J, Honavar V. Feature subset selection using a genetic algorithm. IEEE Intelligent Systems, 1998,13(2):44−49.

[16] Cantu-Paz E. Feature subset selection, class separability, and genetic algorithms. In: Proc. of the Genetic and Evolutionary Computation Conf. (GECCO). Berlin: Springer-Verlag, 2004. 959−970.

[17] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation, 2002,6(2):182−197.

[18] Horn J, Nafpliotis N, Goldberg DE. A niched Pareto genetic algorithm for multiobjective optimization. In: Proc. of the 1st IEEE Conf. on Evolutionary Computation (ICEC'94). NJ: IEEE Press, 1994. 82−87.

[19] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation, 1995, 2(3):221−248.

[20] Kudo M, Sklansky J. Comparison of algorithms that select features for pattern classifiers. Pattern Recognition, 2000,33:25−41.

[21] Somol P, Pudil P, Novovicova J, Paclik P. Adaptive floating search methods in feature selection. Pattern Recognition Letters, 1999, 20(11−13):1157−1163.

[22] Lee WK, Stolfo SJ. A framework for constructing features and models for intrusion detection systems. ACM Trans. on Information and System Security, 2000,3(4):227−261.

[23] The UCI KDD Archive. KDD99 cup dataset. 1999. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

YU Yan was born in 1972. He is a Ph.D. candidate at the Nanjing University. His current research areas are network security, machine learning, etc.

HUANG Hao was born in 1957. He is a professor and doctoral supervisor at the Department of Computer Science and Technology, Nanjing University. His current research areas are information security, etc.