IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 24, 2020, accepted March 11, 2020, date of publication March 17, 2020, date of current version April 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2981373

A Many Objective-Based Feature Selection Model for Anomaly Detection in Cloud Environment

ZHIXIA ZHANG, JIE WEN, JIANGJIANG ZHANG, XINGJUAN CAI, AND LIPING XIE
Complex System and Computational Intelligence Laboratory, Taiyuan University of Science and Technology, Taiyuan 030024, China

Corresponding authors: Xingjuan Cai (caixingjuan@tyust.edu.cn) and Liping Xie (lipingxie@163.com)

This work was supported in part by the Key Research and Development Program of Shanxi Province (High Technology) under Grant 201903D121119, and in part by the Postgraduate Education Innovation Project of Shanxi Province under Grant 2019SY495.

ABSTRACT With the development of cloud computing technology (CCT), the processing of network traffic data becomes particularly important. However, the existing intrusion detection systems (IDS) are not efficient enough in analyzing network traffic data for anomaly detection. Therefore, this paper proposes a new data processing model for network anomaly detection. The model can simultaneously optimize the number of features (NF), accuracy, recall, false alarm rate (FAR) and precision. In order to better solve the model, an integrating dominance algorithm (MaOEA-ABC) with adaptive selection probability is proposed. In model, firstly, MaOEA-ABC is used to obtain the optimal feature subset by optimizing the above five objectives. Then, K-Nearest Neighbor (KNN) is used for network anomaly classification according to the optimal feature subset. Finally, MaOEA-ABC is compared with the existing standard MaOEAs algorithm (NSGA-III, EFR-RR, MaOEA-RD and PICEAg). The experimental results show that the approach can reduce the number of features on the basis of ensuring accuracy and FAR, thereby reducing the cost of detection.

INDEX TERMS Cloud computing, intrusion detection system, feature selection, many-objective optimization, network anomaly detection.

I. INTRODUCTION

With the rapid development of science and technology, computer information technology has been widely used in various industries. In the last decade, cloud computing technology (CCT) has gradually developed and played an important role in production and life [1]. CCT is developed by integrating traditional computer technologies such as grid computing [2], parallel computing [3], utility computing [4], network storage [5] and virtualization with network technologies [6]. Compared with the traditional network, it is equivalent to a large resource pool. People can obtain resources from this resource pool of the network according to their needs. However, with the widespread application of CCT, data has become extremely huge and security issues have become a huge challenge for cloud platforms [7], [8]. To respond to these challenges, researchers have widely used intrusion detection systems (IDS) as defense strategies for cloud security [9].

IDS [10] collects key information in the computer network system, such as the audit data of the operating system, system logs and network data packets. IDS can analyze these network data in detail and provide strategies for deploying security tools to ensure the integrity, confidentiality, and reliability of computer systems [11]. At first, Dorothy proposed an intrusion detection model [12], which is the first IDS with expert system mechanism. Subsequently, Gene and Mark proposed an intrusion detection (ID) architecture using multiple independent autonomous agents that are flexible, scalable, fast recovery and programmatic self-learning. It provided a significant reference for the subsequent development of a distributed IDS. However, the traditional IDS [13], [14] in the cloud environment is facing problems such as massive data, high concurrent access and software compatibility. The massive network data contains many redundant features, which consumes a lot of computing resources and reduces the accuracy of detection [15]. Cyber attackers attack virtual machines through cloud system vulnerabilities and deploy large-scale distributed denial of service (DDoS) [16]–[18], which making cloud users unable

The associate editor coordinating the review of this manuscript and approving it for publication was Sabu M Thampi.

to access cloud systems normally. In response to this problem, Chung *et al.* [19] proposed a distributed vulnerability detection, measurement, and countermeasure selection mechanism. It based on attack graph analysis models and reconfigurable virtual network countermeasures. Aiming at the massive and high-dimensional intrusion behavior in cloud computing system [20], Deng *et al.* [21] proposed a distributed ID method based on hybrid gene expression programming. However, cloud ID technology still has high false alarm rate (FAR) and false negative rate (FNR) [22].

In cloud environment, IDS include misuse detection, anomaly detection, hypervisor introspection, virtual machine introspection and their combination. Chetouani [23] used a black-box recognition method of non-linear autoregressive with exogenous input model for fault detection. Saganowski *et al.* [24] proposed an anomaly detection method based on matching tracking for ID. The approach used the average projection of the reconstructed network signal to determine whether the detected trace is normal or abnormal. The main goal of anomaly detection is to effectively detect attack behavior. In order to reduce the impact of attack behavior, it is equally important to find the attack at the beginning. To this end, Jabez and Muthukumar [25] used neighborhood outlier factor to detect abnormal data. In addition, aiming to improve the performance of IDS, the author also proposed a training model based on a distributed storage environment. The result shows that this method is effective. However, due to the huge number of intrusions and the increasing diversity of intrusions, existing IDS use machine learning and deep learning techniques [26] for anomaly detection. Marir *et al.* [27] adopted a distributed method that combining deep feature extraction and multi-layer integrated support vector machines (SVM) to detect abnormal behavior in large-scale networks. Compared with other ways, the performance of this method is greatly improved. In order to improve the performance of the classifier, Dey *et al.* [28] used a series of feature selection methods to process the dataset. And they used the random forest classifier to detect the exception of the OpenFlow controller. Then, they combined a deep neural network with a recursive unit-long short-term memory network and adopted a recursive feature elimination selection method to improve the performance of the classifier [29]. Furthermore, Zhang *et al.* [30] improved LSTM neural network structure and the spatiotemporal characteristics of streams to perform ID. Experiments show that the model has better accuracy.

The above research shows that anomaly detection technology is an important component of network information security infrastructure. It can analyze network data in detail and provide strategies for deploying security tools to ensure the integrity, confidentiality and reliability of computer systems. However, massive network traffic data contains many irrelevant or redundant features, which not only consume a large amount of computing resources, but also reduce the accuracy of detection in cloud environment [31]. For this reason, an efficient feature selection (FS) algorithm is of great significance for improving anomaly detection techniques [32].

In recent years, swarm intelligence optimization algorithms [33]–[35] have been widely used to deal with various optimization problems, such as medicine [36], [37], wireless sensor networks [38]–[40], image processing [41] and recommendation systems [42], [43]. Therefore, many researchers have also use intelligent algorithms for feature selection, such as genetic algorithm [44], particle swarm optimization [45], cuckoo search algorithm [46], [47]. For examples, Eesa *et al.* [48] adopted cuttlefish optimization algorithm to select features, which takes the number of selected feature (NF) and FAR as the optimization objectives. And the selected features are judged by the decision tree classifier. Jiang [49] used an improved multi-objective evolution algorithm to select the optimal feature subset. The approach compressed the feature space by optimizing the detection rate and FAR. Maza and Touahria [50] used the error rate and the NF as the objective for algorithm optimization and proposes a FS algorithm based on multi-objective evolution distributed.

After analysis, different scholars use diverse objectives for FS, but basically they are NF, accuracy, precision, recall and FAR. These objectives characterize the performance of the FS methods from different perspectives. Therefore, the overall consideration of the above five performance of anomaly detection is very meaningful for constructing an efficient anomaly detection strategy.

In intelligent algorithms, optimization problems with more than three objective functions are called many-objective optimization problems (MaOPs) [51], [52]. Therefore, the problem of network data anomaly detection is a MaOPs. However, few researchers do these studies. For MaOPs, with the problem dimension increases, the number of non-dominated solutions increases exponentially. So, the number of non-dominated solutions exceeds the size of the population and the performance of algorithm decreased [38], [53]–[55]. In view of the above problems, the contribution of this article as follows.

(1) We propose a many objective-based feature selection model for anomaly detection, which can simultaneously optimize accuracy, recall, precision, false alarm rate (FAR), and number of features (NF). And the model will be solved by many-objective evolutionary algorithm (MaOEA).

(2) In order to solve the MaOPs, proposed an integrating dominance algorithm based on adaptive selection probability for MaOPs. This algorithm is implemented by external populations, which communicate with external populations at any time during the evolution process. Then, dynamic probability is used to adjust the weight of each dominance relationship and BFE method is used to update individuals to improve the convergence and diversity of the algorithm.

(3) Aiming at the above model, the proposed algorithm is combined with different classifiers. The model is

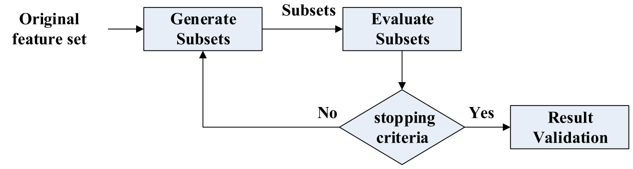


FIGURE 1. The basic process of FS.

tested by using 12 kinds of combination methods to obtain a more suitable method and improve the detection accuracy.

The rest of the paper is organized as follows. The dominance method of MaOEA and basic anomaly detection feature selection techniques are given in Section 2. In Section 3, we detail describe a hybrid model of network anomaly detection with many-objective based on feature dimensionality reduction. The dynamic probability based coupling dominance algorithm is presented in Section 4. Simulation experiment is implemented in Section 5. Finally, the work is summarized in Section 6.

II. THE RELATED MECHANISM

In this section, we introduce basic FS methods for anomaly detection and individual dominance relationships for MaOPs.

A. FEATURE SELECTION METHOD

Currently, massive features of network traffic data become the challenges of IDS. These features are usually redundant and irrelevant. Each network connection record with 41 features are audited by IDS, which will generate $2^{41} - 1$ feature subsets. Such a large number of feature subsets will affect the generalization ability and accuracy of IDS. It can cause computational difficulties and high processing overhead. Therefore, IDS need to preprocess this data and removes irrelevant and redundant features. Then, select some representative features from the original feature set, so that reduce the dimension of the feature space and improve the efficiency and performance of IDS. As shown in Fig.1, FS generally goes through three processes of feature subset generation, evaluation, and verification. Specifically, the FS method is divided into filter and a wrapper [56], as shown in Fig. 2.

In Figure 2 (left), the filter uses information-based metrics to score each feature and select features with scores above the threshold. Information metrics measure the information contained in a feature to assess its importance. In general, information metrics include mutual information (MI) [57], [58], information entropy (IE) [59], and information gain (IG) [60]. In Figure 2 (right), the wrapper is inseparable from the classifier that uses the performance of the classifier as an objective function to evaluate the current feature subset. The wrapper trains a new model for each feature subset, which the computational cost is relatively large. But it can improve the accuracy of classification and find the best-performing feature set compared to the filter [61]. In the paper, k-Nearest Neighbor (KNN) is adopted to attain the class performances.

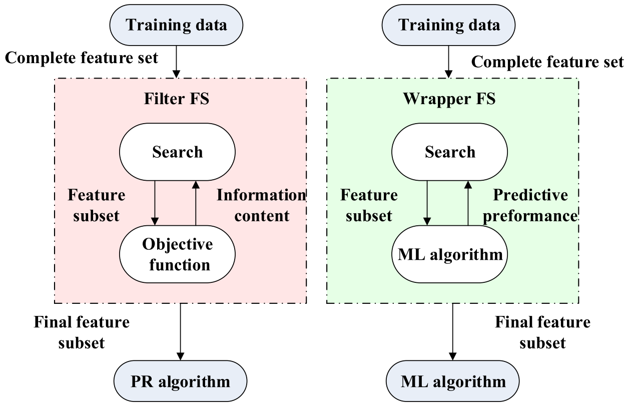
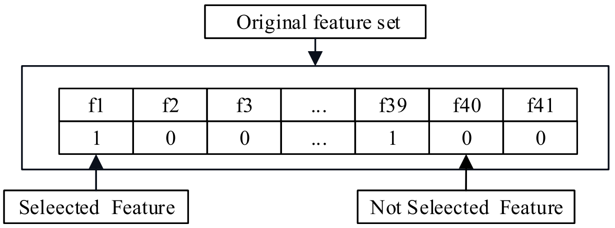


FIGURE 2. Two methods of feature selection.



f1	f2	f3	...	f39	f40	f41
1	0	0	...	1	0	0

FIGURE 3. The encoding of feature set.

B. INDIVIDUAL DOMINANCE RELATION

In MaOEA, individual dominance relation in environment selection has an important impact on the performance of the algorithm. Here are three classic dominance relationships: *Pareto-dominance* [44], *RP-dominance* [62] and *θ-dominance* [63].

1) PARETO- DOMINANCE

For minimizing objective problems, a vector $\vec{f}(\bar{x}) = (f_1(\bar{x}), f_2(\bar{x}), \ldots, f_n(\bar{x}))$ is composed by m objectives component $f_i (i = 1, \ldots, n)$. Given two decisive variables $\bar{x}_u, \bar{x}_v \in U$ randomly:

- If $\bar{x}_u$ dominates to $\bar{x}_v$ only when $\forall i \in \{1, \ldots, n\}$, $f_i(\bar{x}_u) < f_i(\bar{x}_v)$, it is also called $x_u \prec x_v$.
- If $\bar{x}_u$ slightly dominates to $\bar{x}_v$ only when $\forall i \in \{1, \ldots, n\}, f_i(\bar{x}_u) \leq f_i(\bar{x}_v)$ and $\exists j \in \{1, \ldots, n\}, f_j(\bar{x}_u) < f_j(\bar{x}_v)$ that is called $\bar{x}_u$ weakly dominates $\bar{x}_v$.
- If $\bar{x}_u$ and $\bar{x}_v$ are equivalent, only when $\exists i \in \{1, \ldots, n\}, f_i(\bar{x}_u) < f_i(\bar{x}_v)$ and $\exists j \in \{1, \ldots, n\}, f_j(\bar{x}_u) > f_j(\bar{x}_v)$,which is also called $\bar{x}_u$ and $\bar{x}_v$ do not dominate each other.

The NSGA-II algorithm based on *Pareto-dominance* has become a very popular MOEA in the past decade due to its excellent robustness. But it also has certain limitations. For example, non-dominated sorting itself has a high time complexity. In addition, when processing MaOPs, traditional non-dominated sorting reduces the ability of the search process to lead the solution to the Pareto Front (PF).

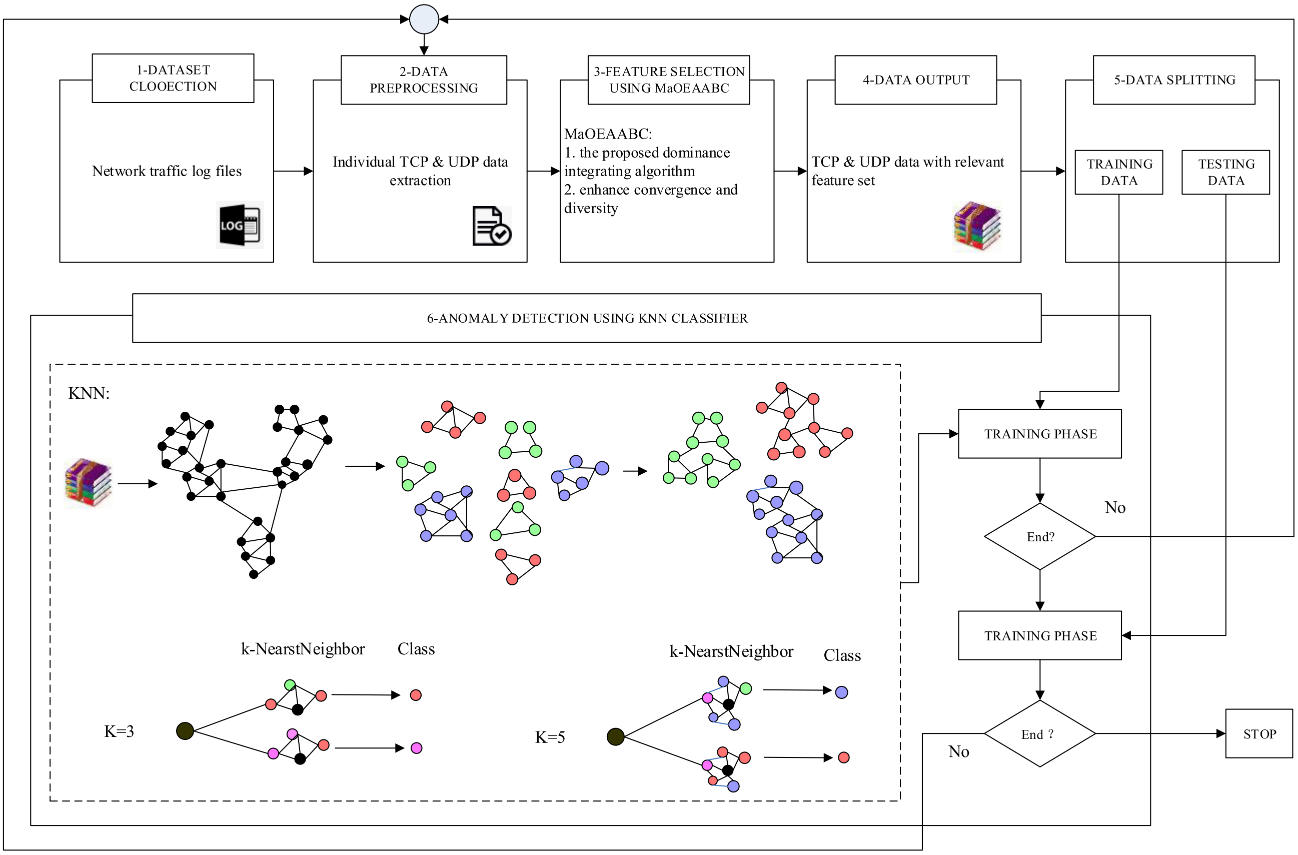

1-DATASET CLOOECTION
Network traffic log files
LOG
2-DATA PREPROCESSING
Individual TCP & UDP data extraction
3-FEATURE SELECTION USING MaOEAABC
MaOEAABC:
1. the proposed dominance integrating algorithm
2. enhance convergence and diversity
4-DATA OUTPUT
TCP & UDP data with relevant feature set
5-DATA SPLITTING
TRAINING DATA
TESTING DATA
6-ANOMALY DETECTION USING KNN CLASSIFIER
KNN:
TRAINING PHASE
End?
No
TRAINING PHASE
End ?
No
STOP
k-NearstNeighbor
Class
K=3
k-NearstNeighbor
Class
K=5

FIGURE 4. The framework of the hybrid model of network anomaly detection with many objectives.

2) RP-DOMINANCE

Given a population P and a set of reference points R, each individual in the population is assigned to the reference point closest to R. For the two decision variables u and v, if the following two conditions are satisfied: When u and v are equivalent, $RP(u) = RP(v)$, $d_1(u) < d_1(v)$ or $RP(u) = RP(v)$, $d_1(u) < d_1(v)$ and $RPDensity(u) < RPDensity(v)$, which called u RP-dominance v or $u \prec_{RP} v$. Where $d_1(u)$ is the d_1 calculated from the reference point to which u belong. $RP(u)$ is the reference point to which u belongs. $RPDensity(u)$ means the number of solutions to the reference points to which u belongs.

RP-dominance is a new dominance relationship, which can replace the traditional dominance relationship to solve MaOPs and provide a new solution. It can be seen as a combination of Pareto-dominance and decomposition based methods that emphasize convergence and diversity while maintaining Pareto sorting.

3) θ-DOMINANCE

Assume there is a series of reference points A in the population St and each solution is associated with the cluster set by clustering operation. The objective function is described as [64] $F_{\mathrm{i}}(x) = d_{i,1}(x) + \theta d_{i,2}(x)$, $i \in \{1, 2, \cdots, N\}$, θ is a predefined penalty parameter. So the θ-*dominance* is defined as follows.

Given two solutions $x, y \in St$, x is said to θ-*dominance* y, denoted by $x \prec_\theta y$, if $x \in C_i$, $y \in C_i$, and $F_i(x) < F_i(y)$, where $i \in \{1, 2, \cdots, N\}$.

θ-*dominance* is motivated by the strength and weakness of two recently suggested many-objective optimizers (MOEA/D [64] and NSGA-III [65], [66]). It enhances the convergence of the NSGA-III by exploiting the fitness evaluation scheme, but still inherit the strength of the former in diversity maintenance.

III. AN FEATURE SELECTION MODEL OF NETWORK ANOMALY DETECTION WITH MANY-OBJECTIVES

A. THE PROPOSED HYBRID MODEL OF NETWORK ANOMALY DETECTION

In this section, the proposed hybrid model is described in detail, which is used for anomaly detection of network traffic data in CC environment. The overall architectural diagram is shown in Fig. 4, which includes six processing stages: 1) dataset collection, 2) data preprocessing, 3) feature selection using MaOEA-ABC, 4) data output, 5) data segmentation, and 6) network anomaly detection using K- Nearest Neighbor (KNN). They are described in detail below.

Data collection NSL-KDD is the first phase of the hybrid model. There are several types of public datasets related to network security: intrusion detection datasets, software

TABLE 1. NSL-KDD dataset features.

No.	Feature Name	Data Type	No.	Feature Name	Attack Class
1	duration	Continuous	22	num_compromised	Nominal
2	land	Nominal	23	srv_serror_rate	Continuous
3	service	Nominal	24	count	Continuous
4	protocol_type	Nominal	25	srv_rerror_rate	Continuous
5	dst_bytes	Continuous	26	srv_count	Continuous
6	wrong_fragment	Continuous	27	same_srv_rate	Continuous
7	src_bytes	Continuous	28	serror_rate	Continuous
8	urgent	Continuous	29	diff_srv_rate	Continuous
9	flg	Nominal	30	rerror_rate	Continuous
10	hot	Continuous	31	srv_diff_host_rate	Continuous
11	num_failed_logins	Continuous	32	dst_host_count	Continuous
12	num_file_creations	Continuous	33	dst_host_srv_count	Continuous
13	logged_in	Nominal	34	dst_host_same_srv_rate	Continuous
14	su_attempted	Nominal	35	dst_host_diff_srv_rate	Continuous
15	root_shell	Continuous	36	dst_host_diff_srv_rate	Continuous
16	num_access_files	Nominal	37	dst_host_srv_diff_host_rate	Continuous
17	num_root	Continuous	38	dst_host_serror_rate	Continuous
18	num_outbound_cmds	Continuous	39	dst_host_srv_serror_rate	Continuous
19	num_shells	Continuous	40	dst_host_rerror_rate	Continuous
20	is_guest_login	Nominal	41	dst_host_srv_rerror_rate	Continuous
21	is_hot_login	Nominal			

defect data sets and software security vulnerability datasets. This paper is mainly about anomaly detection of network intrusion data. Common intrusion detection datasets include KDD Cup 99, NSL-KDD and UNSW_NB15. In these datasets, the NSL-KDD dataset is often used in the literature to evaluate the performance of FS methods. This data set solves some problems contained in the KDD Cup 99 data set, such as a large number of redundant records. Although NSL-KDD dataset also has some problems, such as high complexity. However, it can still be used as an effective benchmark dataset to compare different intrusion detection methods. Therefore, this article uses the NSL-KDD data set for experiments.

NSL-KDD is a simulation data of US Air Force LAN. According to table 1, NSL-KDD contains 41 features (32 continuous and 9 nominal) which are characterized in four categories (TCP connection (basic features (1∼9), content features (10∼22)) and network traffic statistics (time (23∼31) and host (32∼41))). In the second stage, the model preprocesses the input data of MaOEA-ABC. After the character-type features are converted into numerical features, all the feature values are normalized and provided to MaOEA-ABC. Then, in the feature selection phase, MaOEA-ABC is particularly used to select important feature sets, such as IP addresses and port numbers *et al.*, from a given input data set. MaOEA-ABC is a many-objective optimization algorithm of dominance integrating that can help find the best features from the available data set. Temporary packets with relevant features are provided as output at this stage. Next, data were randomly taken as the training set and test set. In the previous stage, the hybrid model was trained to detect anomalous activity in network traffic data, while in the latter stage, it identified anomalous activity by comparing historical data with current input data. Finally, during the anomaly detection phase, a KNN classifier is used to classify the anomalies in the traffic flow.

B. DATA PREPROCESSING

The data preprocessing of this model is divided into two modules: character-type features are converted to numerical and numerical normalization.

In the type conversion module, the protocol_type, service and flag of NSL-KDD [67] are character features. protocol_type indicates the type of network protocol, including TCP, UDP and ICMP. Service means the network service type of the target host, including 70 types such as aol, auth and bgp. The flag indicates that the connection is normal or incorrect, including OTH, REJ and other 9 states. They are converted to numerical by one-hot encoding. Finally, protocol_type, service and flag are mapped to [1, 3], [1, 70] and [1, 11].

The normalization process can be briefly described as follows. Firstly, the average value and average absolute error of each attribute are calculated by using Eq. (1) and Eq. (2). Then, Eq. (3) then normalizes the metric for each data record. Finally, Eq. (4) normalizes each value to [0, 1].

$$\overline{x_k} = \frac{1}{n}\sum_{i=1}^{n} x_{ik} \tag{1}$$

$$S_k = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ik} - \overline{x_k})^2} \tag{2}$$

$$Z_{ik} = \frac{x_{ik} - \overline{x_k}}{S_k} \tag{3}$$

$$x^* = \left\{ \begin{array}{ll} \dfrac{x - \min}{\max - \min}, & \max \neq \min \\ 0, & \max = \min \end{array} \right\} \tag{4}$$

TABLE 2. Size of the training and test datasets.

Type of dataset	dataset Total number of instances	Number of normal instances	Number of normal instances
Training	8000	5600	2400
Test	4000	2500	1500

where x_k and S_k represent the mean and average absolute error of the $k - th$ feature, respectively. x_{ik} and Z_{ik} are the $k - th$ feature of the $i - th$ data and after normalization. max and min represent the maximum value and the minimum value of the sample data. x^* is the normalized data.

C. FEATURE SET ENCODING

The feature set is encoded in binary form in figure 3. Each piece of network data consists of 41 feature attributes, where "0" indicates that the feature is not selected and "1" indicates that the feature is selected [45]. Then, an initial population is generated by this coding method. Population is optimized by using MaOEA-ABC to obtain an optimal feature subset.

D. DATA SEPARATION

The NSL-KDD data set is divided into labeled training data and unlabeled test data. The test data and training data have different probability distributions. The test data contains some types of attacks that do not appear in the training data, which makes intrusion detection more realistic. As shown in table 2, we randomly select 8000 data from the training dataset, and 4000 data from test dataset.

E. THE OBJECTIVE FUNCTIONS OF FEATURE SELECTION

This paper proposes a coupling dominance optimization algorithm that simultaneously optimizes NF, accuracy, recall, FAR and precision. Then, it can search for the optimal feature subset by population iteration. For detailed algorithm description, refer to IV. Each record of NSL-KDD is classified as normal or attacks (DOS, Probe, U2R and R2L) connections. Table 3 shows the confusion matrix of FS. The detection instances are divided into four cases: True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). Under the KNN classifier, five optimization objectives as follows.

$$NF = \text{the number of feature subsets selected} \tag{5}$$

$$accuarcy_{IDS} = \frac{TP + TN}{TP + TN + FN + FP} \tag{6}$$

$$recall_{IDS} = \frac{TP}{TP + FN} \tag{7}$$

$$FAR_{IDS} = \frac{FP}{FP + TN} \tag{8}$$

$$precision_{IDS} = \frac{TP}{TP + FP} \tag{9}$$

TABLE 3. Confusion matrix of the FS.

Actual value	Predicted value	
	Normal	**Attacks**
Normal	TP	FN
Attacks	FP	TN

In the above equations, a smaller NF value indicates that irrelevant and redundant features are removed to a greater extent, and the detection speed and accuracy are improved. The higher the precision, the better the quality of detection for ID. In practical application, data predicted as attack need to allocate certain resources. Therefore, the smaller the FAR, the less resources will be wasted.

IV. THE PROPOSED INTEGRATING DOMINANCE ALGORITHM

In this section, MaOEA-ABC, an integrating approach based on Pareto-dominance, RP-dominance and θ-dominance, is proposed to solve above hybrid model. The general framework of the MaOEA-ABC is given in Algorithm1. Then, several important operations involved in the algorithm framework are described in detail, including environmental selection and update operation.

Since the environmental selection operator determines the search direction of population in the evolution process, which largely determines the convergence and diversity of MaOEAs. The purpose of environment selection is to select new populations from the parents and offspring. This article analyzes the characteristics of three individual domination methods in Section II: Pareto-dominance, RP-dominance and θ-dominance. Then, integrate them to achieve better efficiency. Meanwhile, during the running of the algorithm, the higher selection probability of the dominance approach, the better solutions are obtained and it will be executed in the next generation with greater probability. In order to be fair, the initial probability of the three domination methods is set to 1/3. Then, the selection probabilities of different domination methods will be adaptively adjusted according to the pros and cons of their solutions.

In additional, maintaining the diversity of the distribution of solution sets can better avoid the problem of premature local optimal value in the many-objective optimization algorithm. In order to maintain good convergence and diversity of solution set, a diversity preserving strategy of Balanceable Fitness Estimation (BFE) method is adopted. The BFE method takes into account the two influencing factors of convergence distance and diversity distance, and is an effective evaluation index to balance each convergence ability and diversity.

Assuming that a population $P = \{p_1, p_2, \ldots, p_N\}$ contains N individuals. The BFE calculation formula is as follows:

$$BFE(p_i, P) = \alpha^* Cd(p_i, P) + \beta^* Cv(p_i, P) \tag{10}$$

where $Cd(p_i, P)$ represents the normalized diversity of p_i, $Cv(p_i, P)$ represents the normalized convergence distance of p_i; α and β are two weight factors represent the diversity and convergence distance respectively. By changing the parameters of α and β, the influence degree of individual diversity and convergence distance can be adjusted to achieve dynamic balance.

As mentioned before, the MaOEA-ABC combines three basic individual dominance relationships to solve MaOPs. It can save some of the best solutions during the search process by constructing an external archive. First, determine which one dominance method is chosen to evolutionary population through selection probability. Then, the archive update operator is used to update the external archive. Every 10 generations, the sequential probability adaptive adjustment is performed. Among, a and b are the upward and downward adjustments, respectively. Detailed algorithm framework such as Algorithm 1.

V. SIMULATION STUDY

In this section, we first test the performance of the improved algorithm MaOEA-ABC on the benchmark problems to prove the effectiveness of the proposed algorithm. Secondly, in the background of the IDS, the effect of the MaOEA-ABC based on coordinated dominance is validated to process the proposed network anomaly detection model. In addition, SVM and NB are used for comparative experiments to verify that KNN classifier can improve the performance of the model.

A. PERFORMANCE TEST OF MAOEA-ABC

In this study, two famous benchmark functions, DTLZ (DTLZ1-DTLZ7) and WFG (WFG1-WFG9), are adopted to test the effectiveness of MaOEA-ABC. Both the DTLZ and WFG benchmark problems contain a variety of characteristics, which show varying degrees of problem complexity and can be used to test the performance of MaOEAs. Among them, DTLZ can test the properties of the algorithm, including multi-mode, hybrid and convex. WFG can test the connectivity, preference and discontinuity of the algorithm. The objective number of these functions from 3 to 15, including 3,5,8,10,15. For the DTLZ1 [68], its objective functions $f_i \in [0, 0.5]$, rather than other test functions (DTLZ2-DTLZ7) $f_i \in [0, 1]$. For the WFG [69], the true PF of WFG2 is convex and disconnected, but WFG3 and WFG4-WFG9 are linear and concave, respectively.

Inverse generation distance (IGD) [70], is an evaluation index proposed based on Wilcoxon test and Friedman test. It has been widely used in the distribution and convergence effects of comprehensive evaluation algorithms on benchmark problems. To measure the performance of MaOEA-ABC, IGD is used to consider both convergence and diversity of non-dominated individuals [71] as indicator. IGD is given as follows.

$$IGD = \frac{\sqrt{\sum\limits_{k=1}^{n} d_k^2}}{PF*} \tag{11}$$

Algorithm 1 MaOEA-ABC Algorithm Framework

```
Input: population size ($N$), the number of objective($M$)
        maximum number of iterations ($T$), Initial selection
        probability($pro_1, pro_2, pro_3$), Step length ($a$, $b$)
while $t < T$ do
    Initialize the Population $P = \{p_1, p_2, \cdots, p_N\}$,
    initialize the $Archive$ (External population);
    Calculate the mean of BFE ($Mean$) with $P$ according
    to Eq. (10);
    Off = Tournament Selection ($P$);
    If  mod(t,10) == 0
        A = EnvironmentalSelection_1($P$, Off);
        B = EnvironmentalSelection_2($P$, Off);
        C = EnvironmentalSelection_3($P$, Off);
        Calculate the number of more than Mean in A,
         B and C, called to $Na$, $Nb$ and $Nc$;
       If  $(Na > Nb)$ && $(Na > Nc)$
        $pro_1 = pro_1 + a$;  $pro_2 = pro_2 - b$;
       $pro_3 = pro_3 - b$;
       Else If  $(Nb > Na)$ && $(Nb > Nc)$
        $pro_1 = pro_1 - b$;  $pro_2 = pro_2 + a$;
       $pro_3 = pro_3 - b$;
       Else
        $pro_1 = pro_1 - b$;  $pro_2 = pro_2 - b$;
       $pro_3 = pro_3 + a$;
       End If
  Else
     Q = rand;
     If  Q <= $pro_1$
       Population = EnvironmentalSelection_1($P$, Off);
     Else If  Q > $pro_1$ && Q < ($pro_1$+ $pro_2$)
       Population = EnvironmentalSelection_2($P$, Off);
     Else If  Q > ($pro_1$+ $pro_2$) && Q < $pro_3$
  Population = EnvironmentalSelection_3($P$, Off);
  $Archive$ = Update Archive($Archive$, $P$);
     End If
  End If
End While
Output: $Archive$
```

where, n is the number of solutions in the true PF^* and d_i represents the Euclidean distance from the solution k of PF^* to the closest solution of the approximated PF. And the smaller of IGD means the better performance.

To test the performance effects of the proposed algorithms and the effectiveness of constructing the dominant policy pool, MaOEA-ABC was compared with a variety of algorithms with high performance effects, including NSGA-III [65], MaOEA-RD [72] EFR-RR [73] and PICEAg [74], and all the key parameters were set according to their original literature for convincing results.

For more details about these algorithms, please refer to the related literature. In table 4, the population size setting is given. Table 5 describes the parameter settings of 4 different

TABLE 4. The population size setting.

Number of Objectives (M)	Divisions(H)	Population Size (N)
3	12	91
5	6	210
8	H1=3, H2=2	156
10	H1=3, H2=2	275
15	H1=2, H2=1	135

TABLE 5. Parameter settings of 4 different algorithms.

Algorithm	Parameter
EFR-RR	*k*=2
MaOEA-RD	*pc*=1, pm=1/D
PICEAg	*NGoal=100*M*
NSGA-III	*pc*=1, pm=1/D

algorithms. And *pc* denotes the cross probability; D is the number of variables; *k* is Number of nearest weight vectors. Table 6 and Table 7 respectively shows the comparison results of MaOEA-ABC and the remaining four existing MaOEAs on DTLZ and WFG. Meanwhile, all algorithm run independently 20 times. Among them, the "+", "−", and "=" respectively mean that the number of compared algorithm is better, worse and similar than the optimized result of MaOEA-ABC.

From the results of Table 4, MaOEA-ABC exceeds NSGA-III and MaOEA-RD on 20 problems, while NSGA-III and MaOEA-RD achieve better solutions than MaOEA-ABC on 7 problems. For the rest of 8 problems, both of them obtain similar performance. Compared to EFR-RR and PICEAg, our approach is better on 14 problems, but EFR-RR achieves better results on 10 problems. For the rest of 11 problems, both of them obtain similar performance. But PICEAg obtains better results on 13 problems. For DTLZ5, MaOEA-ABC is better than the other five algorithms. MaOEA-RD achieves poor convergence and diversity on DTLZ2. For DTLZ3 and DTLZ6, MaOEA-ABC is second only to PIEAg. It is difficult for other algorithms to obtain solutions with good convergence and distribution. For DTLZ1, DTLZ4 and DTLZ7, MaOEA-ABC is better than other algorithms, NSGA-III,

TABLE 6. IGD of different algorithms on the DTLZ.

Problem	M	NSGA-III	EFR-RR	MaOEA-RD	PICEAg	MaOEA-ABC
DTLZ1	3	1.7903e-1 (2.28e-1) +	5.4176e-1 (3.50e-1) =	1.8022e+0 (1.26e+0) -	3.6232e-1 (3.01e-1) +	6.6328e-1 (4.57e-1)
	5	1.5293e+0 (6.87e-1) =	6.5045e+0 (2.24e+0) -	2.5665e+0 (1.34e+0) -	1.6199e+0 (7.16e-1)-	1.0428e+0 (4.22e-1) +
	8	1.8738e+0 (6.53e-1) -	9.4858e-1 (4EF.68e-1) =	1.0382e+0 (6.75e-1) =	1.3127e+0 (6.72e-1)-	7.2751e-1 (3.21e-1) +
	10	3.8828e+0 (1.77e+0) -	1.6468e+0 (6.44e-1) =	2.2614e+0 (8.89e-1) -	1.6810e+0 (5.06e-1) =	1.5534e+0 (6.73e-1)
	15	1.6324e+0 (7.41e-1) -	6.1647e-1 (3.50e-1) =	1.2187e+0 (8.72e-1) =	8.8516e-1 (4.80e-1) =	8.4322e-1 (4.82e-1)
DTLZ2	3	5.5013e-2 (2.40e-4) +	5.6641e-2 (7.17e-4) +	6.2441e-2 (1.60e-3) +	1.0818e-1 (7.13e-3) -	7.7850e-2 (8.10e-3)
	5	1.8806e-1 (3.77e-3) =	2.0761e-1 (6.68e-3) -	2.0705e-1 (1.54e-2) -	2.4529e-1 (5.61e-3) -	1.9003e-1 (7.51e-3)
	8	4.0094e-1 (7.82e-2) =	3.8929e-1 (4.53e-3) -	5.1839e-1 (4.70e-2) -	4.5420e-1 (4.15e-2) -	3.6613e-1 (5.39e-3)
	10	5.6248e-1 (7.77e-2) -	4.8093e-1 (1.59e-2) -	6.4629e-1 (5.32e-2) -	5.5746e-1 (5.71e-2) -	4.3840e-1 (1.49e-2)
	15	7.7627e-1 (7.70e-2) -	5.8338e-1 (1.40e-2) +	9.3265e-1 (3.22e-2) -	9.2218e-1 (5.16e-2) -	6.5301e-1 (4.54e-2)
DTLZ3	3	1.0960e+1 (4.27e+0) +	1.6871e+1 (6.74e+0) =	3.4095e+1 (1.21e+1) -	8.4483e+0 (3.41e+0) +	2.0665e+1 (7.74e+0)
	5	6.2072e+1 (1.18e+1) =	1.0191e+2 (1.66e+1) -	7.7574e+1 (1.74e+1) -	4.3304e+1 (1.01e+1) +	5.4845e+1 (1.08e+1)
	8	8.2452e+1 (2.55e+1) -	5.6531e+1 (1.53e+1) -	4.1711e+1 (1.53e+1) =	2.4638e+1 (5.66e+0) +	4.2695e+1 (1.40e+1)
	10	1.6335e+2 (4.26e+1) -	8.7167e+1 (1.84e+1) -	7.9506e+1 (1.62e+1) -	5.2724e+1 (9.56e+0) +	6.7991e+1 (1.46e+1)
	15	1.3327e+2 (3.12e+1) -	3.4359e+1 (1.24e+1) =	3.5855e+1 (1.67e+1) =	1.3102e+1 (5.25e+0) +	3.9374e+1 (1.56e+1)
DTLZ4	3	2.3745e-1 (3.39e-1) -	8.1403e-2 (1.09e-1) +	8.5421e-2 (1.07e-1) +	2.3867e-1 (2.42e-1) -	1.4258e-1 (1.70e-1)
	5	1.9239e-1 (5.83e-3) +	1.9133e-1 (1.78e-2) +	2.1151e-1 (6.91e-2) +	2.5658e-1 (4.21e-2) =	2.7218e-1 (9.97e-2)
	8	4.5176e-1 (1.00e-1) =	3.7185e-1 (2.97e-2) +	4.2406e-1 (4.91e-2) +	4.8264e-1 (5.09e-2) =	5.0789e-1 (1.01e-1)
	10	5.9112e-1 (7.54e-2) -	4.7763e-1 (1.30e-2) +	4.7142e-1 (1.88e-2) +	5.2170e-1 (2.24e-2) =	5.3029e-1 (4.67e-2)
	15	7.2143e-1 (6.13e-2) =	6.3271e-1 (2.32e-3) +	6.8950e-1 (2.14e-2) =	7.3134e-1 (2.33e-2) =	7.2056e-1 (6.31e-2)
DTLZ5	3	1.2656e-2 (9.65e-4) =	3.7133e-2 (5.13e-3) -	7.4213e-1 (5.63e-5) -	8.9460e-2 (7.36e-2) -	1.3239e-2 (1.54e-3)
	5	2.0492e-1 (4.30e-2) -	2.4790e-1 (5.84e-2) -	6.6157e-1 (7.41e-2) -	6.9139e-2 (2.06e-2) =	6.5286e-2 (1.98e-2)
	8	2.5707e-1 (8.43e-2) -	3.3846e-1 (5.34e-2) -	5.3544e-1 (2.24e-1) -	3.2947e-1 (7.61e-2) -	1.7196e-1 (3.99e-2)
	10	2.4113e-1 (6.30e-2) =	3.1627e-1 (4.28e-2) -	2.7706e-1 (1.74e-1) =	2.7481e-1 (5.94e-2) -	2.1218e-1 (4.48e-2)
	15	4.6591e-1 (1.43e-1) -	3.0944e-1 (7.09e-2) =	6.0025e-1 (2.05e-1) -	5.8533e-1 (8.45e-2) -	2.9357e-1 (5.13e-2)
DTLZ6	3	1.7837e-2 (1.90e-3) +	9.6182e-2 (1.63e-1) =	1.6951e+0 (7.22e-1) -	1.4713e-1 (1.51e-1) =	3.8412e-1 (3.62e-1)
	5	3.6415e+0 (7.60e-1) -	6.2168e+0 (6.55e-1) -	4.0878e+0 (9.30e-1) -	2.0344e-1 (5.48e-2) +	2.3861e+0 (5.31e-1)
	8	5.4909e+0 (5.48e-1) -	3.6157e+0 (5.86e-1) -	3.9694e+0 (1.16e+0) -	4.9685e-1 (1.03e-1) +	2.2361e+0 (8.52e-1)
	10	6.7591e+0 (5.46e-1) -	5.5857e+0 (5.26e-1) -	4.8229e+0 (7.87e-1) -	5.5934e-1 (1.27e-1) +	3.8390e+0 (5.14e-1)
	15	6.6434e+0 (6.72e-1) -	2.5588e+0 (8.01e-1) =	3.8910e+0 (1.03e+0) -	7.7758e-1 (1.10e-1) +	2.1963e+0 (1.03e+0)
DTLZ7	3	9.4128e-2 (6.87e-3) +	1.1678e-1 (9.23e-3) +	1.2160e+0 (3.56e-1) -	4.9633e-1 (2.90e-1) -	1.6505e-1 (6.93e-2)
	5	7.9099e-1 (1.70e-1) +	1.8301e+0 (1.01e+0)-	1.1431e+0 (2.25e-1) +	9.9453e-1 (5.06e-1) +	7.5371e-1 (6.56e-2) +
	8	5.5994e+0 (1.06e+0) -	3.6071e+0 (1.21e+0)-	2.8283e+0 (1.06e+0) +	6.0456e+0 (3.75e-1) -	2.1468e+0 (4.91e-1) +
	10	1.4426e+1 (1.24e+0) -	8.3016e+0 (1.67e+0) =	7.9760e+0 (3.06e+0) =	1.1583e+1 (7.82e-1) -	9.3988e+0 (2.26e+0)
	15	1.9105e+1 (2.22e+0) -	1.6361e+1 (2.48e+0) =	1.4878e+1 (4.69e+0) =	1.9485e+1 (2.19e+0) -	1.4995e+1 (3.53e+0)
Best/All		6/35	6/35	3/35	9/35	11/35
+/-/=		7/20/8	8/16/11	7/20/8	11/16/8	

TABLE 7. IGD of different algorithms on the WFG.

Problem	M	NSGA-III	EFR-RR	MaOEA-RD	PICEAg	MaOEA-ABC
WFG1	3	1.0779e+0 (7.86e-2) +	9.1152e-1 (6.72e-2) +	1.2454e+0 (9.73e-2) =	9.6079e-1 (8.23e-2) +	1.1724e+0 (1.29e-1)
	5	1.9037e+0 (3.99e-2) -	1.7608e+0 (7.91e-2) +	2.0134e+0 (8.92e-2) -	1.7203e+0 (6.75e-2) +	1.8310e+0 (9.46e-2)
	8	2.6544e+0 (6.99e-2) -	2.3154e+0 (7.74e-2) +	2.6257e+0 (1.19e-1) -	2.1711e+0 (1.12e-1) +	2.3934e+0 (1.13e-1)
	10	3.1242e+0 (6.41e-2) -	2.9680e+0 (5.14e-2) =	3.0772e+0 (7.67e-2) -	2.7325e+0 (6.05e-2) +	2.9334e+0 (7.99e-2)
	15	3.8377e+0 (1.72e-1) =	3.2365e+0 (1.45e-1) +	4.0980e+0 (1.02e-1) -	3.0179e+0 (1.10e-1) +	3.7708e+0 (1.58e-1)
WFG2	3	1.7060e-1 (4.85e-3) +	1.9343e-1 (6.08e-3) +	1.9099e-1 (8.49e-3) +	3.1327e-1 (1.24e-2) =	3.0430e-1 (2.49e-2)
	5	4.2111e-1 (1.38e-2) +	4.6027e-1 (1.23e-2) +	4.9160e-1 (1.92e-2) =	5.7771e-1 (5.57e-2) -	4.9403e-1 (3.86e-2)
	8	1.0201e+0 (1.26e-1) +	9.6976e-1 (2.48e-2) +	1.3089e+0 (3.09e-1) -	1.4446e+0 (1.58e-1) -	1.1133e+0 (7.39e-2)
	10	1.1212e+0 (5.59e-2) +	1.0660e+0 (4.60e-2) +	1.7084e+0 (6.83e-1) -	1.7358e+0 (2.48e-1) -	1.3062e+0 (5.17e-1)
	15	2.1570e+0 (8.73e-1) =	2.0812e+0 (2.12e-1) -	4.3442e+0 (1.64e+0) -	5.5884e+0 (1.78e+0) -	1.9317e+0 (1.15e-1)
WFG3	3	1.5628e-1 (1.98e-2) -	1.6471e-1 (1.82e-2) -	2.1644e+0 (6.94e-1) -	1.3914e-1 (1.03e-1) -	8.9932e-2 (1.53e-2)
	5	5.8629e-1 (6.10e-2) -	5.7958e-1 (4.92e-2) -	8.0008e-1 (1.05e-1) -	1.5694e-1 (2.01e-2) +	2.7746e-1 (3.56e-2)
	8	1.0717e+0 (2.73e-1) -	9.3760e-1 (8.88e-2) -	1.5405e+0 (3.08e-1) -	4.7378e+0 (1.95e+0) -	5.8361e-1 (1.65e-1)
	10	1.1888e+0 (1.36e-1) -	1.0567e+0 (1.19e-1) -	1.6690e+0 (2.29e-1) -	9.6480e-1 (7.75e-1) =	8.3666e-1 (2.26e-1)
	15	3.4318e+0 (1.28e+0) -	2.8164e+0 (3.65e-1) -	2.5629e+0 (4.70e-1) -	1.5149e+1 (6.28e-1) -	1.6786e+0 (5.74e-1)
WFG4	3	2.3212e-1 (2.47e-3) +	2.4302e-1 (3.37e-3) +	2.7050e-1 (7.70e-3) +	3.6064e-1 (2.60e-2) -	3.0072e-1 (1.34e-2)
	5	9.8690e-1 (6.46e-3) +	1.0478e+0 (1.09e-2) -	1.1204e+0 (1.40e-1) -	1.2419e+0 (3.10e-2) -	1.0183e+0 (1.33e-2)
	8	3.0410e+0 (9.67e-2) =	3.2721e+0 (4.94e-2) -	4.3957e+0 (5.24e-1) -	4.7027e+0 (7.08e-1) -	3.0362e+0 (3.35e-2)
	10	4.3244e+0 (1.11e-1) -	4.2816e+0 (7.01e-2) =	6.8925e+0 (9.44e-1) -	6.3777e+0 (6.06e-1) -	4.2444e+0 (1.24e-1)
	15	9.4428e+0 (3.53e-1) -	9.5362e+0 (3.00e-1) -	1.6635e+1 (1.38e+0) -	1.9871e+1 (2.02e+0) -	9.0502e+0 (3.66e-1)
WFG5	3	2.3781e-1 (1.71e-3) +	2.5180e-1 (4.69e-3) +	2.7014e-1 (6.06e-3) +	4.0007e-1 (4.08e-2) -	2.9693e-1 (1.12e-2)
	5	9.8405e-1 (9.18e-3) +	1.0559e+0 (1.58e-2) -	1.0703e+0 (1.94e-2) -	1.3022e+0 (3.44e-2) -	1.0044e+0 (1.32e-2)
	8	3.0216e+0 (2.70e-2) +	3.3193e+0 (5.20e-2) -	3.8217e+0 (3.26e-1) -	3.5875e+0 (3.54e-2) -	3.0864e+0 (3.14e-2)
	10	4.3535e+0 (8.62e-2) -	4.3633e+0 (5.95e-2) -	5.4882e+0 (4.02e-1) -	5.1300e+0 (3.99e-1) -	4.1700e+0 (5.41e-2)
	15	8.9710e+0 (3.06e-1) -	9.3122e+0 (4.22e-1) -	1.3913e+1 (1.23e+0) -	1.9428e+1 (1.21e+0) -	8.3461e+0 (9.64e-2)
WFG6	3	2.7684e-1 (1.44e-2) +	2.7809e-1 (1.43e-2) +	3.2451e-1 (3.17e-2) =	4.1579e-1 (3.03e-2) -	3.2795e-1 (2.25e-2)
	5	1.0460e+0 (1.13e-2) =	1.0903e+0 (1.64e-2) -	1.1834e+0 (3.93e-2) -	1.3513e+0 (4.51e-2) -	1.0535e+0 (2.19e-2)
	8	3.1592e+0 (3.73e-1) +	3.3916e+0 (6.10e-2) -	3.9288e+0 (2.35e-1) -	3.7641e+0 (2.29e-1) -	3.2054e+0 (5.19e-2)
	10	4.4474e+0 (8.16e-2) -	4.4598e+0 (8.98e-2) -	5.7571e+0 (5.14e-1) -	5.2022e+0 (2.78e-1) -	4.2705e+0 (7.22e-2)
	15	9.1723e+0 (3.37e-1) -	9.4051e+0 (2.49e-1) -	1.4757e+1 (1.70e+0) -	1.6280e+1 (1.36e+0) -	8.4777e+0 (1.33e-1)
WFG7	3	2.3141e-1 (2.32e-3) +	2.4119e-1 (3.77e-3) +	2.7731e-1 (1.07e-2) +	4.5114e-1 (3.68e-2) -	2.9632e-1 (1.39e-2)
	5	9.9822e-1 (1.04e-2) +	1.0754e+0 (1.20e-2) -	1.0888e+0 (3.10e-2) -	1.4107e+0 (3.34e-2) -	1.0158e+0 (1.32e-2)
	8	3.0402e+0 (3.33e-2) +	3.2756e+0 (4.41e-2) -	3.9964e+0 (2.92e-1) -	3.7347e+0 (1.13e-1) -	3.0829e+0 (3.67e-2)
	10	4.4561e+0 (1.48e-1) -	4.4442e+0 (6.09e-2) -	5.8113e+0 (4.60e-1) -	5.4891e+0 (4.41e-1) -	4.2329e+0 (8.08e-2)
	15	9.9575e+0 (5.68e-1) -	1.0031e+1 (6.11e-1) -	1.4763e+1 (1.15e+0) -	1.6520e+1 (2.05e+0) -	8.6506e+0 (1.31e-1)
WFG8	3	3.2301e-1 (9.80e-3) +	3.2768e-1 (7.43e-3) +	3.7780e-1 (1.69e-2) -	5.0023e-1 (4.07e-2) -	3.5919e-1 (1.21e-2)
	5	1.1241e+0 (1.95e-2) +	1.1270e+0 (1.46e-2) +	1.3602e+0 (6.28e-2) -	1.3233e+0 (3.60e-2) -	1.1428e+0 (2.19e-2)
	8	3.5515e+0 (2.31e-1) -	3.3699e+0 (9.18e-2) -	4.3809e+0 (2.79e-1) -	4.8279e+0 (3.99e-1) -	3.3245e+0 (2.74e-2)
	10	4.8747e+0 (1.41e-1) -	4.9115e+0 (1.28e-1) -	6.1736e+0 (4.06e-1) -	6.4318e+0 (3.39e-1) -	4.6067e+0 (9.18e-2)
	15	1.0278e+1 (5.63e-1) -	1.0829e+1 (3.77e-1) -	1.5260e+1 (1.34e+0) -	1.7217e+1 (1.63e+0) -	9.1164e+0 (2.16e-1)
WFG9	3	2.4657e-1 (1.77e-2) +	2.4799e-1 (2.52e-2) +	2.9638e-1 (4.42e-2) =	3.7562e-1 (2.13e-2) -	3.0193e-1 (4.86e-2)
	5	9.9046e-1 (2.17e-2) -	1.0196e+0 (1.04e-2) -	1.1731e+0 (6.99e-2) -	1.2068e+0 (3.13e-2) -	9.7548e-1 (1.59e-2)
	8	3.1721e+0 (1.14e-1) -	3.1551e+0 (3.50e-2) -	4.0364e+0 (3.41e-1) -	3.5046e+0 (2.06e-1) -	3.0442e+0 (4.37e-2)
	10	4.5582e+0 (1.52e-1) -	4.3345e+0 (7.66e-2) -	5.4824e+0 (2.41e-1) -	5.3783e+0 (4.65e-1) -	4.2483e+0 (9.09e-2)
	15	9.5819e+0 (4.46e-1) -	9.4570e+0 (3.30e-1) -	1.2894e+1 (9.03e-1) -	1.7869e+1 (1.29e+0) -	8.5555e+0 (2.17e-1)
Best/All		16/45	3/45	0/45	5/45	23/45
+/-/=		18/23/4	15/28/2	4/37/4	6/37/2	

EFR-RR and MaOEA-RD achieve similar performance. According to the above results, our approach has better performance than other four algorithms on the DTLZ benchmark test.

In Table 5, MaOEA-ABC exceeds NSGA-III on 23 problems, while NSGA-III achieves better solutions than MaOEA-ABC on 18 problems. For the rest of 4 problems, both of them obtain similar performance. Compared to MaOEA-RD and PICEAg, our approach is better on 37 problems, but EFR-RR achieves better results on 4 problems. For the rest of 4 problems, both of them obtain similar performance. But PICEAg obtains better results on 6 problems. For WFG1, PICEAg is better than the other five algorithms. NSGA-III and EFR-RR achieve more convergence and diversity on DTLZ2, while MaOEA-ABC has better with 10 and 15 objectives. For WFG4-WFG8, NSGA-III achieves better performance with 3, 5 and 8 objectives, MaOEA-ABC is on 10 and 15 objectives. MaOEA-ABC is better than the other four algorithms on WFG3 and WFG9. From the above results, it can be found that the proposed MaOEA-ABC has better performance on the WFG benchmark set.

TABLE 8. The performance metrics of the different algorithm on SVM.

Algorithm		Accuracy (%)	NF	F-measure (%)	FAR (%)
Mean	EFR-RR	95.407	24	96.765	**2.108**
	NSGA-III	95.203	22	96.580	3.218
	MaOEA-RD	95.314	22	96.665	3.017
	PICEAg	95.482	21	96.786	2.810
	Our	**95.739**	**14**	**96.986**	2.112
Best	EFR-RR	**97.112**	35	**97.942**	5.179
	NSGA-III	95.774	29	96.996	3.893
	MaOEA-RD	95.950	26	97.119	3.947
	PICEAg	95.924	28	97,103	5.072
	Our	96.750	**18**	97.694	**2.679**
Worse	EFR-RR	92.200	16	94.652	**0.643**
	NSGA-III	**93.961**	16	95.740	2.482
	MaOEA-RD	93.050	17	95.173	1.429
	PICEAg	93.912	16	95.619	2.000
	Our	93.362	**10**	**95.964**	1.393

TABLE 9. The performance metrics of the different algorithm on KNN.

Algorithm		Accuracy (%)	NF	F-measure (%)	FAR (%)
Mean	EFR-RR	98.167	18.	98.702	0.788
	NSGA-III	98.564	19	98.976	0.862
	MaOEA-RD	98.561	23	98.974	0.885
	PICEAg	98.572	16	98.984	0.692
	Our	**98.578**	**11**	**98.988**	**0.622**
Best	EFR-RR	**98.974**	26	99.269	1.428
	NSGA-III	98.749	22	99.108	**1.071**
	MaOEA-RD	98.962	29	**99.260**	1.178
	PICEAg	98.962	26	99.259	1.321
	Our	98.712	**16**	99.083	0.982
Worse	EFR-RR	94.449	13	96.163	**0.464**
	NSGA-III	98.049	17	98.614	0.696
	MaOEA-RD	95.962	16	97.176	0.571
	PICEAg	95.499	10	96868	0.482
	Our	**98.362**	**9**	**98.833**	0.482

B. THE PERFORMANCE OF MAOEA-ABC ON MODEL

In this section, the proposed model is tested by different algorithms on diversity classifiers (KNN, SVM, NB). Meanwhile, the parameters of optimization algorithm are similar to section V.

In section III, precision and recall are contradictory metrics to a certain extent. In order to balance the precision and comprehensiveness of the model, F-measure is used as the evaluation indicators as Eq. (12). Therefore, in order to better evaluate the performance of the model, this paper uses the following evaluation indicators to analyze the results.

$$\begin{cases} accuarcy_{IDS} = \dfrac{TP+TN}{TP+TN+FN+FP} \\ FAR_{IDS} = \dfrac{FP}{FP+TN} \\ NF = \text{the number of feature subsets selected} \\ F-measure = \dfrac{2 \cdot recall_{IDS} \cdot precision_{IDS}}{recall_{IDS} + precision_{IDS}} \end{cases} \tag{12}$$

In order to compare the effects of different classifiers on the model, finding a more suitable solution. Under differen classifiers, the proposed MaOEA-ABC and existing MaOEAs (including NSGA-III, EFR-RR, MaOEA-RD, and PICEAg) are used to solve the feature selection model for anomaly detection.

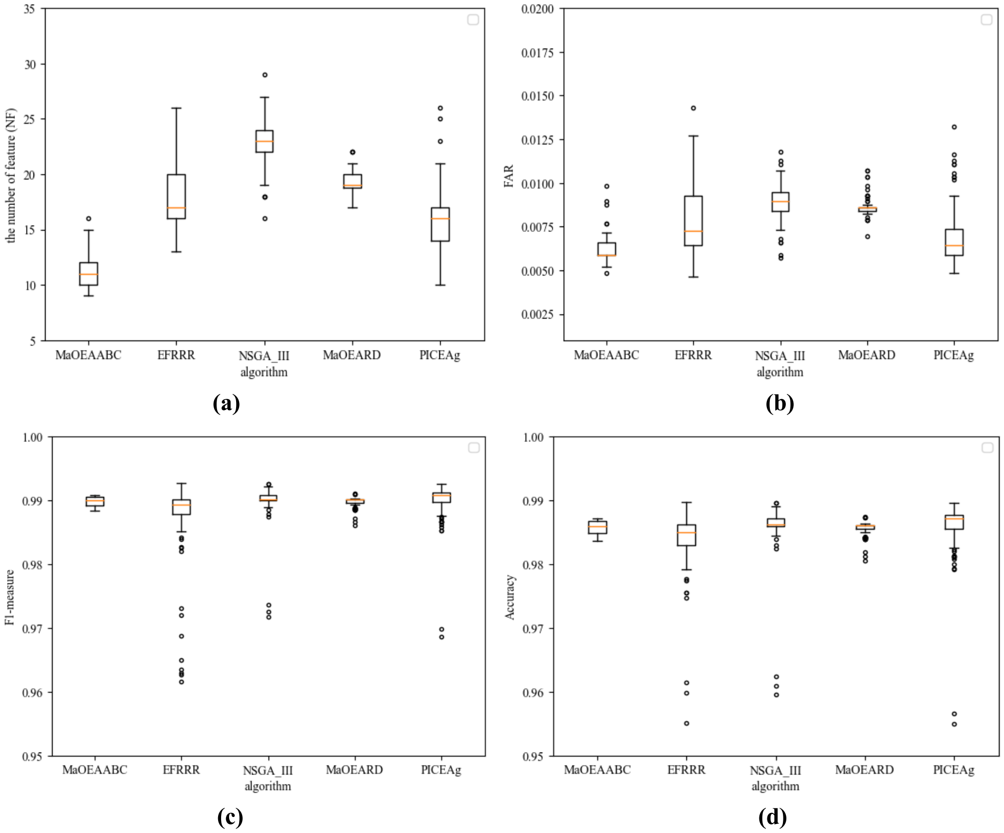
The SVM maps the data from the original sample space to the high-dimensional space through a non-linear mapping kernel function. Then construct an optimal classification hyperplane to distinguish the two types of samples, so that the distance between the two types of samples on the hyperplane is the largest. Since SVM adopts the principle of structure risk minimization, its generalization ability is strong. At the same time, it can solve the problem of insufficient data and nonlinear regression. Therefore, we use it as one of the contrast classifiers.

The classification principle of NB is to calculate the posterior probability by using the prior probability of an object. That is, select the class with the maximum posterior probability as the class to which the object belongs. For large-scale data training, the process of NB is simple and the speed of training is fast. Therefore, we also use it as one of the contrast classifiers.

Table 8 shows the performance indicators of different algorithms on SVM. For the mean, our algorithm performs better on accuracy, NF and F-measure than other algorithms. For the best, EFR-RR algorithm has better performance on F-measure and accuracy. MaOEA-ABC is slightly worse than EFR-RR, but better than the other three algorithms. For the worse, our algorithm outperforms the other four algorithms

TABLE 10. The performance metrics of the different algorithm on NB.

Algorithm		Accuracy (%)	NF	F-measure (%)	FAR (%)
Mean	EFR-RR	85.791	20	90.053	7.961
	NSGA-III	78.605	17	83.830	20.261
	MaOEA-RD	85.356	19	89.832	7.542
	PICEAg	85.039	18	89.482	8.992
	Our	**86.135**	**10**	**90.694**	**3.482**
Best	EFR-RR	88.148	29	91.785	20.718
	NSGA-III	**88.648**	25	**92.153**	26.308
	MaOEA-RD	87.998	29	91.748	12.770
	PICEAg	87.698	28	91.468	**15.413**
	Our	86.885	**16**	90.942	15.89
Worse	EFR-RR	78.384	14	83.699	3.982
	NSGA-III	74.459	15	80.155	4.768
	MaOEA-RD	81.835	13	87.051	3.072
	PICEAg	82.185	13	87.032	5.161
	Our	**83.823**	**8**	**88.456**	**2.000**



(a)

(b)

(c)

(d)

FIGURE 5. The performance metrics value of different algorithms on KNN.

on NF and F-measure. Table 9 and table 10 respectively compare the performance indicators of different algorithms on KNN and NB. For the mean and the worst, our algorithm performs better than other algorithms. In the best, the five algorithms are not different.

Generally speaking, MaOEA-ABC has the best mean value in accuracy and NF when solving the proposed model in different classifiers. In additional, it is also surpass most algorithms both F-measure and FAR. Among them, the performance metrics is the best with KNN classifier and NB is the worst. At the same time, by comparing the extreme values (maximum value and minimum value) of the five algorithms on the four evaluation indexes, it can be seen that MaOEA-ABC has good convergence.

The results above show that KNN classifier has the best performance for the proposed model. In order to express the experimental results more intuitively, Fig.5 shows that box diagrams of all the solutions obtained through MaOEAs optimization. In Fig. 5, there are four black lines and one red line in each box respectively representing the maximum value, the first quartile value, the last quartile value, the minimum value and the median value of the data in this box. In additional, the circle in the figure represents the discrete individual.

In Fig (a), the NF of MaOEA-ABC algorithm is minimum, which means the removal of redundancy is greatest. PICEAg is similar to MaOEA-ABC, but more discrete. However, the NF of MaOEA-RD algorithm was between 16 and 20, with little overall difference. The redundancy removal effective of NSGA-III is not very good, and the maximum number of features reaches 30. Fig (b) reflects the distribution of FAR, MaOEA-RD and NSGA-III are relatively dispersed, indicating their poor stability. However, the other three algorithms perform better. In Fig (c), the F-measure of MaOEA-RD, PICEAg and MaOEA-ABC are all up to 99% overall, while the other two algorithms are between 96% and 99%. Finally, the accuracy dispersion of NSGA-III and EFR-RR is large, and the performance of other algorithms is similar.

Although the extreme values in the results of the MaOEA-ABC optimized model may be discrete individuals, the overall performance of the algorithm have a great effect in the four performance indicators. Due to each solution represents a set of coefficients and also represents a feature set of selected, the decision maker can select the appropriate subset of features for anomaly detection according to its own conditions. The above figure verifies that MaOEA-ABC based anomaly detection model has good performance in four indicators.

VI. CONCLUSION AND FUTURE WORK

This paper presents a many-objective hybrid anomaly detection model based on feature reduction. This model is optimized for NF, accuracy, precision, recall and FAR through three modules: data preprocessing, FS and anomaly detection. In order to better solve this model, we propose an integrated domination optimization algorithm (MaOEA-ABC) based on adaptive probability. It contains two key strategies, one is the integration strategy, which can choose the outstanding individuals with convergence and diversity in evolving. The other is an adaptive selection probability strategy, which can update the selection probability of each domination method according to the individual's pros and cons in the current environment to improve the algorithm's ability to search for the optimal solution. The proposed MaOEA-ABC optimization test results have good results on the five optimization goals. By comparing experiments with four existing MaOEAs, the results show that MaOEA-ABC not only has a good effect on the benchmarking problem, but also can detect more accuracy on the basis of selecting fewer features. In summary, the models and algorithms proposed have better performance for IDS. Since the classifier contains many parameters, it will affect the detection performance. In future work, we will work on improving the classifier to improve the efficiency of IDS detection.

REFERENCES

[1] F. Sabahi, "Cloud computing security threats and responses," in *Proc. IEEE 3rd Int. Conf. Commun. Softw. Netw.*, May 2011, pp. 245–249.

[2] D. Lim, Y.-S. Ong, Y. Jin, B. Sendhoff, and B.-S. Lee, "Efficient hierarchical parallel genetic algorithms using grid computing," *Future Gener. Comput. Syst.*, vol. 23, no. 4, pp. 658–670, May 2007.

[3] G. Chen, G. Sun, Y. Xu, and B. Long, "Integrated research of parallel computing: Status and future," *Sci. Bull.*, vol. 54, no. 11, pp. 1845–1853, Jun. 2009.

[4] J. Soldatos, M. Serrano, and M. Hauswirth, "Convergence of utility computing with the Internet-of-Things," in *Proc. 6th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.*, Jul. 2012, pp. 874–879.

[5] M. I. Khan, W. N. Gansterer, and G. Haring, "In-network storage model for data persistence under congestion in wireless sensor network," in *Proc. 1st Int. Conf. Complex, Intell. Softw. Intensive Syst. (CISIS)*, Vienna, Austria, Apr. 2007, pp. 10–12.

[6] G.-F. Xiang, H. Jin, D.-Q. Zou, and X.-G. Chen, "Virtualization-based security monitoring," *J. Softw.*, vol. 23, no. 8, pp. 2173–2187, Sep. 2012.

[7] V. Balamurugan and R. Saravanan, "Enhanced intrusion detection and prevention system on cloud environment using hybrid classification and OTS generation," *Cluster Comput.*, vol. 22, no. S6, pp. 13027–13039, Nov. 2019.

[8] Z. Cui, F. Xue, S. Zhang, X. Cai, Y. Cao, W. Zhang, and J. Chen, "A hybrid BlockChain-based identity authentication scheme for multi-WSN," *IEEE Trans. Services Comput.*, to be published.

[9] S. Garg, K. Kaur, G. Kaddoum, S. H. Ahmed, and D. N. K. Jayakody, "SDN-based secure and privacy-preserving scheme for vehicular networks: A 5G perspective," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8421–8434, Sep. 2019.

[10] J. Hu, X. Yu, D. Qiu, and H.-H. Chen, "A simple and efficient hidden Markov model scheme for host-based anomaly intrusion detection," *IEEE Netw.*, vol. 23, no. 1, pp. 42–47, Jan. 2009.

[11] A. Anitha and V. Vaidehi, "Context based application level intrusion detection system," in *Proc. Int. Conf. Netw. Services (ICNS)*, 2006, p. 16.

[12] J. Anderson, "Computer security threat monitoring and surveillance," vol. 11, no. 2, pp. 65–68, Jan. 1980.

[13] X. Wang, T.-L. Huang, and X.-Y. Liu, "Research on the intrusion detection mechanism based on cloud computing," in *Proc. Int. Conf. Intell. Comput. Integr. Syst.*, Oct. 2010, pp. 125–128.

[14] S. Roschke, F. Cheng, and C. Meinel, "Intrusion detection in the cloud," in *Proc. 8th IEEE Int. Conf. Dependable, Autonomic Secure Comput. (DASC)*, Chengdu, China, Dec. 2009, pp. 12–14.

[15] N. Chergui and N. Boustia, "Contextual-based approach to reduce false positives," *IET Inf. Secur.*, vol. 14, no. 1, pp. 89–98, Jan. 2020.

[16] L. Li and G. Lee, "DDoS attack detection and wavelets," *Telecommun. Syst.*, vol. 28, nos. 3–4, pp. 435–451, Mar. 2005.

[17] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: Classification and state-of-the-art," *Comput. Netw.*, vol. 44, no. 5, pp. 643–666, Apr. 2004.

[18] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "DDoS attacks in cloud computing: Issues, taxonomy, and future directions," *Comput. Commun.*, vol. 107, pp. 30–48, Jul. 2017.

[19] C.-J. Chung, P. Khatkar, T. Xing, J. Lee, and D. Huang, "NICE: Network intrusion detection and countermeasure selection in virtual network systems," *IEEE Trans. Dependable Secure Comput.*, vol. 10, no. 4, pp. 198–211, Jul. 2013.

[20] Z. Ji, Q. Gao, and W. Hai, "Outlier detection for high-dimensional data streams," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2015.

[21] S. Deng, A.-H. Zhou, D. Yue, B. Hu, and L.-P. Zhu, "Distributed intrusion detection based on hybrid gene expression programming and cloud computing in a cyber physical power system," *IET Control Theory Appl.*, vol. 11, no. 11, pp. 1822–1829, Jul. 2017.

[22] Z. Cui, L. Du, P. Wang, X. Cai, and W. Zhang, "Malicious code detection based on CNNs and multi-objective algorithm," *J. Parallel Distrib. Comput.*, vol. 129, pp. 50–58, Jul. 2019.

[23] Y. Chetouani, "A non-linear auto-regressive moving average with exogenous input non-linear modelling and fault detection using the cumulative sum (Page-Hinkley) test: Application to a reactor," *Int. J. Comput. Appl. Technol.*, vol. 32, no. 3, pp. 187–193, 2008.

[24] Ł. Saganowski, M. Choraś, R. Renk, and W. Hołubowicz, "A novel signal-based approach to anomaly detection in IDS systems," in *Proc. Int. Conf. Adapt. Natural Comput. Algorithms*, 2009, pp. 527–536.

[25] J. Jabez and B. Muthukumar, ''Intrusion detection system (IDS): Anomaly detection using outlier detection approach,'' *Procedia Comput. Sci.*, vol. 48, pp. 338–346, Jan. 2015.

[26] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, ''A hybrid deep learning-based model for anomaly detection in cloud datacenter networks,'' *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 3, pp. 924–935, Sep. 2019.

[27] N. Marir, H. Wang, G. Feng, B. Li, and M. Jia, ''Distributed abnormal behavior detection approach based on deep belief network and ensemble SVM using spark,'' *IEEE Access*, vol. 6, pp. 59657–59671, 2018.

[28] S. K. Dey, M. M. Rahman, and M. R. Uddin, ''Detection of flow based anomaly in OpenFlow controller: Machine learning approach in software defined networking,'' in *Proc. 4th Int. Conf. Electr. Eng. Inf. Commun. Technol. (iCEEiCT)*, Sep. 2018, pp. 416–421.

[29] S. K. Dey and M. M. Rahman, ''Effects of machine learning approach in flow-based anomaly detection on software-defined networking,'' *Symmetry*, vol. 12, no. 1, p. 7, 2020.

[30] Y. Zhang, X. Chen, L. Jin, X. Wang, and D. Guo, ''Network intrusion detection: Based on deep hierarchical network and original flow data,'' *IEEE Access*, vol. 7, pp. 37004–37016, 2019.

[31] H. Liu and L. Yu, ''Toward integrating feature selection algorithms for classification and clustering,'' *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005.

[32] M. N. Sudha, S. Selvarajan, and M. Suganthi, ''Feature selection using improved lion optimisation algorithm for breast cancer classification,'' *Int. J. Bio-Inspired Comput.*, vol. 14, no. 4, p. 237, Jan. 2019.

[33] M. S. Gharajeh, ''T*: A weighted double-heuristic search algorithm to find the shortest path,'' *Int. J. Comput. Sci. Math.*, vol. 10, no. 1, p. 58, Jan. 2019.

[34] G. Li and P. Ge, ''A novel computation method for 2D deformation of fish scale based on SURF and N-R optimisation,'' *Int. J. Comput. Sci. Math.*, vol. 10, no. 2, pp. 203–213, 2019.

[35] T. H. Jiang and C. Zhang, ''Adaptive discrete cat swarm optimisation algorithm for the flexible job shop problem,'' *Int. J. Bio-Inspired Comput.*, vol. 13, no. 3, pp. 199–208, 2019.

[36] K. O. Okosun, O. Rachid, and N. Marcus, ''Optimal control strategies and cost-effectiveness analysis of a malaria model,'' *Biosystems*, vol. 111, no. 2, pp. 83–101, Feb. 2013.

[37] J. D. Sweetlin, H. K. Nehemiah, and A. Kannan, ''Computer aided diagnosis of drug sensitive pulmonary tuberculosis with cavities, consolidations and nodular manifestations on lung CT images,'' *Int. J. Bio-Inspired Comput.*, vol. 13, no. 2, pp. 71–85, 2019.

[38] X. Cai, P. Wang, L. Du, Z. Cui, W. Zhang, and J. Chen, ''Multi-objective three-dimensional DV-hop localization algorithm with NSGA-II,'' *IEEE Sensors J.*, vol. 19, no. 21, pp. 10003–10015, Nov. 2019.

[39] P. Wang, J. Huang, L. Xie, Z. Cui, and J. Chen, ''A Gaussian error correction multi-objective positioning model with NSGA-II,'' *Concurrency Comput. Pract. Exper.*, vol. 32, no. 5, p. e5464, 2020.

[40] O. S. Younes, ''Modelling and analysis of TCP congestion control mechanisms using stochastic reward nets,'' *Int. J. Comput. Sci. Math.*, vol. 10, no. 4, pp. 390–412, 2019.

[41] R. Chouder and N. Benhamidouche, ''New exact solutions to nonlinear diffusion equation that occurs in image processing,'' *Int. J. Comput. Sci. Math.*, vol. 10, no. 4, pp. 364–374, 2019.

[42] Z. Chai, Y.-L. Li, Y.-M. Han, and S.-F. Zhu, ''Recommendation system based on singular value decomposition and multi-objective immune optimization,'' *IEEE Access*, vol. 7, pp. 6060–6071, 2019.

[43] Z. Cui, X. Xu, F. Xue, X. Cai, Y. Cao, W. Zhang, and J. Chen, ''Personalized recommendation system based on collaborative filtering for IoT scenarios,'' *IEEE Trans. Services Comput.*, to be published.

[44] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, ''A fast and elitist multiobjective genetic algorithm: NSGA-II,'' *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[45] Z. Cui, J. Zhang, D. Wu, X. Cai, H. Wang, W. Zhang, and J. Chen, ''Hybrid many-objective particle swarm optimization algorithm for green coal production problem,'' *Inf. Sci.*, vol. 518, pp. 256–271, May 2020.

[46] F. Xue and H. Tang, ''Cuckoo search algorithm with different distribution strategy,'' *Int. J. Bio-Inspired Comput.*, vol. 13, no. 4, p. 234, 2019.

[47] X. Cai, Y. Niu, S. Geng, J. Zhang, Z. Cui, J. Li, and J. Chen, ''An undersampled software defect prediction method based on hybrid multi-objective cuckoo search,'' *Concurrency Comput. Pract. Exper.*, vol. 32, no. 5, p. e5478, 2020.

[48] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, ''A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems,'' *Expert Syst. Appl.*, vol. 42, no. 5, pp. 2670–2679, Apr. 2015.

[49] J. F. Jiang, ''Feature selection based on multi-objective evolutionary algorithm for intrusion detection,'' *Comput. Eng. Appl.*, 2010.

[50] S. Maza and M. Touahria, ''Feature selection for intrusion detection using new multi-objective estimation of distribution algorithms,'' *Int. J. Speech Technol.*, vol. 49, no. 12, pp. 4237–4257, Dec. 2019.

[51] M. Li, S. Yang, and X. Liu, ''Bi-goal evolution for many-objective optimization problems,'' *Artif. Intell.*, vol. 228, pp. 45–65, Nov. 2015.

[52] R. Jiao, C. Li, R. Wang, and S. Zeng, ''Constrained optimisation by solving equivalent dynamic loosely-constrained multiobjective optimisation problem,'' *Int. J. Bio-Inspired Comput.*, vol. 13, no. 2, p. 86, 2019.

[53] Z. Cui, J. Zhang, Y. Wang, Y. Cao, X. Cai, W. Zhang, and J. Chen, ''A pigeon-inspired optimization algorithm for many-objective optimization problems,'' *Sci. China Inf. Sci.*, vol. 62, no. 7, Jul. 2019, Art. no. 070212.

[54] Z. Cui, Y. Cao, X. Cai, J. Cai, and J. Chen, ''Optimal LEACH protocol with modified bat algorithm for big data sensing systems in Internet of Things,'' *J. Parallel Distrib. Comput.*, vol. 132, pp. 217–229, Oct. 2019.

[55] B. Xu, J. Liu, T. Ma, Y. Xue, and B. Zhao, ''Multi-objective classification based on NSGA-II,'' *Int. J. Comput. Sci. Math.*, vol. 9, no. 6, pp. 539–546, 2018.

[56] H. Bostani and M. Sheikhan, ''Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems,'' *Soft Comput.*, vol. 21, no. 9, pp. 2307–2324, May 2017.

[57] P. Viola and W. M. Wells, III, ''Alignment by maximization of mutual information,'' in *Proc. IEEE Int. Conf. Comput. Vis.*, 2002, vol. 24, no. 2, pp. 16–23.

[58] F. Amiri, M. R. Yousefi, C. Lucas, A. Shakery, and N. Yazdani, ''Mutual information-based feature selection for intrusion detection systems,'' *J. Netw. Comput. Appl.*, vol. 34, no. 4, pp. 1184–1199, Jul. 2011.

[59] K. Zheng, X. Wang, B. Wu, and T. Wu, ''Feature subset selection combining maximal information entropy and maximal information coefficient,'' *Int. J. Speech Technol.*, vol. 50, no. 2, pp. 487–501, Feb. 2020.

[60] X. Wang, B. Guo, Y. Shen, C. Zhou, and X. Duan, ''Input feature selection method based on feature set equivalence and mutual information gain maximization,'' *IEEE Access*, vol. 7, pp. 151525–151538, 2019.

[61] Y. Li, J.-L. Wang, Z.-H. Tian, T.-B. Lu, and C. Young, ''Building lightweight intrusion detection system using wrapper-based feature selection mechanisms,'' *Comput. Secur.*, vol. 28, no. 6, pp. 466–475, Sep. 2009.

[62] M. Elarbi, S. Bechikh, A. Gupta, L. Ben Said, and Y.-S. Ong, ''A new decomposition-based NSGA-II for many-objective optimization,'' *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 7, pp. 1191–1210, Jul. 2018.

[63] Y. Yuan, H. Xu, B. Wang, and X. Yao, ''A new dominance relation-based evolutionary algorithm for many-objective optimization,'' *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 16–37, Feb. 2016.

[64] Q. Zhang and H. Li, ''MOEA/D: A multiobjective evolutionary algorithm based on decomposition,'' *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[65] K. Deb and H. Jain, ''An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints,'' *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014.

[66] Z. Cui, Y. Chang, J. Zhang, X. Cai, and W. Zhang, ''Improved NSGA-III with selection-and-elimination operator,'' *Swarm Evol. Comput.*, vol. 49, pp. 23–33, Sep. 2019.

[67] D. H. Deshmukh, T. Ghorpade, and P. Padiya, ''Improving classification using preprocessing and machine learning algorithms on NSL-KDD dataset,'' in *Proc. Int. Conf. Commun., Inf. Comput. Technol. (ICCICT)*, Jan. 2015, pp. 1–6.

[68] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, ''Scalable test problems for evolutionary multiobjective optimization,'' in *Evolutionary Multiobjective Optimization* (Advanced Information and Knowledge Processing). 2005, pp. 105–145.

[69] B. Li, K. Tang, J. Li, and X. Yao, ''Stochastic ranking algorithm for many-objective optimization based on multiple indicators,'' *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp. 924–938, Dec. 2016.

[70] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, ''Reference point specification in inverted generational distance for triangular linear Pareto front,'' *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 961–975, Dec. 2018.

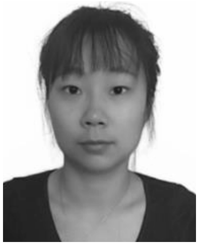
[71] S. Jiang, Y.-S. Ong, J. Zhang, and L. Feng, "Consistencies and contradictions of performance metrics in multiobjective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2391–2404, Dec. 2014.

[72] Z. He and G. G. Yen, "Many-objective evolutionary algorithm: Objective space reduction and diversity improvement," *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 145–160, Feb. 2016.

[73] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 180–198, Apr. 2016.

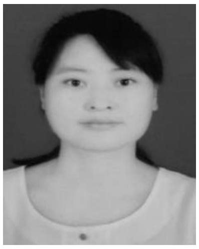
[74] R. Wang, R. C. Purshouse, and P. J. Fleming, "Preference-inspired coevolutionary algorithms for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 474–494, Aug. 2013.

ZHIXIA ZHANG is currently pursuing the master's degree in computer science and technology with the Taiyuan University of Science and Technology, China. Her research interests include computational intelligence and combinatorial optimization.



JIE WEN is currently pursuing the master's degree in computer science and technology with the Taiyuan University of Science and Technology, China. Her research interests include computational intelligence and combinatorial optimization.

JIANGJIANG ZHANG is currently pursuing the master's degree in computer science and technology with the Taiyuan University of Science and Technology, China. His research interests include computational intelligence and combinatorial optimization.



XINGJUAN CAI received the Ph.D. degree in control science and engineering from Tongji University, China, in 2017.

She is currently an Associate Professor with the School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan, China. Her research interest includes bio-inspired computation and application.

LIPING XIE received the Ph.D. degree in control theory and control engineering from the Lanzhou University of Technology, in 2010. She is currently a Professor with the Institute of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan, China. She has published more than 30 international journal articles and conference papers. Her current research interests include swarm intelligence and swarm robotics.