IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 26, 2018, accepted October 20, 2018, date of publication November 14, 2018, date of current version December 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879891

Intrusion Detection Based on Parallel Intelligent Optimization Feature Extraction and Distributed Fuzzy Clustering in WSNs

ZHOUZHOU LIU[1,2], (Member, IEEE), WEI WEI[3], (Senior Member, IEEE), HAO WANG[4], YANGMEI ZHANG[1], QIANYUN ZHANG[1], AND SHINING LI[2]

[1]School of Electronic Engineering, Xi'an Aeronautical University, Xi'an 710077, China
[2]School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China
[3]School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China
[4]Norwegian University of Science and Technology, 710077 Aalesund, Norway

Corresponding author: Wei Wei (weiwei@xaut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871313, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2017JM6096, in part by the Xi'an Science and Technology Project under Grant 2017076CG/RC039(XAHK001), in part by the School Level Scientific Research Fund under Grant 2017KY1112, in part by the Scientific Research Program Funded by Shaanxi Provincial Education Department under Program 2013JK1139, in part by the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant 20136118120010, and in part by the Key Research and Development Program of Shaanxi Province under Grant 2018ZDXM-GY-036.

ABSTRACT Aiming at large-scale, high dimensional data, and variable intrusion behavior in wireless sensor networks (WSNs), an intrusion detection algorithm based on parallel intelligent optimization feature extraction and distributed fuzzy clustering for WSNs is proposed. First, in order to effectively reduce the data dimensionality and improve the robustness of the feature extraction process, the parallel intelligent optimization feature extraction framework is constructed on the basis of defining the optimal feature evaluation index, for which the theoretical analysis shows that the index can eliminate feature redundancy and maintain the diversity of original data. Second, the spider cluster optimization algorithm evolution rule is redefined by introducing local search and adaptive multi strategy update method, and it proves that the improved social spider optimization (ISSO) algorithm has global convergence. The ISSO is used to solve the feature extraction framework, and through the parallel feature subset selection process, the best feature combination is extracted. Finally, WSNs intrusion detection is carried out by using the best feature subset and the distributed fuzzy clustering technology, and intelligent iterative evolution method and adaptive clustering strategy are introduced in order to improve the fuzzy clustering algorithm performance. Experimental results show that the intrusion detection algorithm can effectively give the results of intrusion detection, and moreover, compared with the other detection algorithms, the intrusion detection rate is improved by about 13.1%, and the false detection rate is decreased by about 8.5%.

INDEX TERMS Feature extraction, fuzzy clustering, intrusion detection, social spider optimization algorithm, wireless sensor networks.

I. INTRODUCTION

With the rapid development of network communication, embedded computing and sensor technology [1], WSNs (wireless sensor networks), composed of a large number of sensor nodes which could do some compute and wireless communication, is widely used in engineering applications [2]. However, due to the vulnerability of WSNs to intrusion attacks such as black hole, wormhole and physical manipulation [3], its further development is restricted by network security [4]. Therefore, an efficient intrusion detection algorithm is of great importance for the WSNs to function properly.

Intrusion detection is used to detect various attacks quickly and accurately, and making responses immediately [5]. It is often categorized into two classes: misuse detection and anomaly detection. However, the latter one has received more concerns due to its ability to detect unknown attacks. Basically, the anomaly detection approaches can be distinguished between three main types: supervised, semi-supervised and unsupervised learning anomaly detection [6]. Among them, unsupervised learning identifies an access that can not to be modeled as normal operations as an attack without classify its type, more suitable for large-scale high dimensional WSNs intrusion detection. But such a "Yes or No" strategy will reduce the accuracy of the intrusion detection. Nevertheless, when dealing with the mass WSNs real-time data, feature selection can be used to optimize the WSNs data effectively by extracting feature subset with high resolution [8]. Due to its ability to reduce the data complexity while maximizing the accuracy of detection, the feature subset selection method has attracted increasing attentions, and SPFS (similarity preserving feature selection) is representative [9]. Hence, various feature extraction methods were proposed by scholars, such as Filter model, Wrapper model, Embedded model [10], PCA (principal component analysis), LDA (linear discriminant analysis) [11], etc. These methods usually adopt single variable evaluation rule to select feature subset, which may not be the optimal. Therefore, based on multi variable evaluation rule, researchers proposed new rules such as Pearson coefficient, maximum information compression index [12], maximum information coefficient (MIC) [13], etc. However, most of these multi-variable evaluation rules failed to measure the relationship between the redundancy of the feature subset and the diversity of the original data accurately. Consequently, multiple metrics fusion emerged as a popular tool to select feature subset. Sun *et al.* [14] combined the MIC with symmetric uncertainty measurement, effectively selected the feature subset by using Markov blanket method. Based on Pearson coefficient, Qiu *et al.* [15] introduced information gain metrics into feature extraction. Removing the irrelevant features, and then the redundant features, to achieve feature subset selection. Their method is also demonstrated to be effective by simulations. But these methods mentioned above need multiple steps to successfully select the feature subset, which in turn increased the algorithm's time complexity. After feature subset is selected, models such as neural network, rough set, support vector machine and cluster analysis can be used to detect intrusions. Because the boundary of normal behavior and intrusion behavior is sometimes difficult to define, intrusion detection with fuzzy clustering has drawn significant attention. Tang *et al.* [16] proposed an improved fuzzy clustering algorithm based on FCM (AGFCM) to detect anomaly intrusion behavior. Elhag *et al.* [6] used genetic algorithm to determine the optimal combination of the characteristic parameters, and fuzzy clustering method to detect intrusions. Nayak *et al.* [17] incorporated elicit teaching learning with the Fuzzy *c*-means clustering algorithm to obtain the improved fitness values of the cluster centers. However, the defects of fuzzy clustering (sensitive to the initial cluster center, easy to fall into local optimal, and the number of clusters needs to be determined beforehand, etc.) are still need to be solved.

With the rapid expansion of the WSNs' scale, and the exponentially explosive growth of data needs to be processed, the robustness of feature subset selection, the accuracy and promptness of intrusion detection are of great importance. Therefore, this paper proposed an intrusion detection algorithm based on parallel intelligent optimization feature extraction and distributed fuzzy clustering for WSNs. The major contributions of this paper include:

(1) We define a new feature evaluation index to eliminate the feature redundancy and maintain the diversity of the original data.

(2) We formulate our feature extraction framework, and utilize the improved social spider algorithm and parallel computing, to ensure the extraction of the best feature combination while improve the robustness of the feature selection.

(3) We propose an improved fuzzy clustering method for efficient and reliable WSNs intrusion detection.

II. PARALLEL INTELLIGENT OPTIMIZATION FEATURE EXTRACTION FRAMEWORK

A. FEATURE SUBSET EXTRACTION

In this paper, we use $\mathbf{D} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ to denote the data matrix, where $\boldsymbol{x}_i$ is data set and n is the number of data sets. For each data set $\boldsymbol{x}_i$, we use $f_{i1}, f_{i2}, \cdots, f_{im}$ to denote the m features, and $\boldsymbol{F}_i = (f_{i1}, f_{i2}, \cdots, f_{im})$ is the corresponding feature vector. We also use $\boldsymbol{C} = \left\{c_j\right\}_{j=1}^{c}$ to denote the c classes of the data matrix $\mathbf{D}$. Thus, feature subset extraction is to select k features from $\boldsymbol{F}_i = (f_{i1}, f_{i2}, \cdots, f_{im})$ to form a feature subset, which can provide almost the same classification and identification ability as the original data matrix.

Definition 1: Define feature extraction vector $\boldsymbol{P}$ as:

$$\boldsymbol{P} = \left(p_1, \cdots, p_j, \cdots, p_m\right), \quad p_j \in \{0, 1\}, \ \boldsymbol{P}^T\mathbf{1} = k \tag{1}$$

where $p_j = 1$ indicates that the corresponding feature is selected, otherwise $p_j = 0$. For data set $\boldsymbol{x}_i$, we have $\boldsymbol{F}_i\boldsymbol{P}^T = \sum_{j=1}^{k} \hat{f}_{ij}$, where $\left(\hat{f}_{i1}, \hat{f}_{i2}, \cdots, \hat{f}_{ik}\right)$ is the feature description after feature extraction.

Definition 2: Given $\boldsymbol{P}$, define feature extraction matrix $\boldsymbol{W}$ as:

$$\boldsymbol{W}_{m\times c} = \left(\boldsymbol{P}^T, \boldsymbol{P}^T \cdots, \boldsymbol{P}^T\right) = \begin{bmatrix} p_1 & p_1 & \cdots & p_1 \\ p_2 & p_2 & \cdots & p_2 \\ \vdots & \vdots & \ddots & \vdots \\ p_m & p_m & \cdots & p_m \end{bmatrix}$$
$$= \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \|\boldsymbol{W}\|_{2,0} = k \tag{2}$$

Since the value of p_i is either 0 or 1, $\boldsymbol{W}$ is also composed of 0 and 1. And $\boldsymbol{W}$ has and only has k rows of 1.

Using $\boldsymbol{W}$ to extract feature from $\boldsymbol{D}$, we obtain feature subset $\widehat{\boldsymbol{F}} = \left\{\widehat{\boldsymbol{f}}_1, \widehat{\boldsymbol{f}}_2 \cdots \widehat{\boldsymbol{f}}_k\right\}$, as shown in Fig. 1.

From Fig.1 we can see that the extracted feature subset $\widehat{\boldsymbol{F}} = \left\{\widehat{\boldsymbol{f}}_1, \widehat{\boldsymbol{f}}_2 \cdots \widehat{\boldsymbol{f}}_k\right\}$ describes the features of the sample data

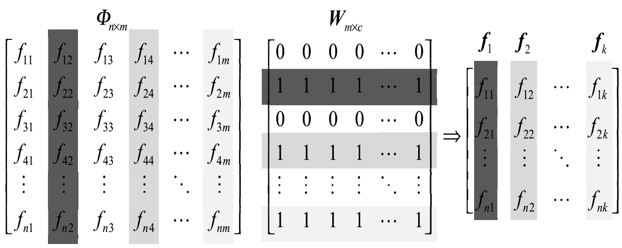


FIGURE 1. Feature subset extraction.

that corresponding to the non-zero elements of the feature extraction vector $\boldsymbol{P}$, which indicates that the feature subset is extracted from the data samples. To evaluate the feature subset $\widehat{\boldsymbol{F}}$, we have **Definition 3**.

Definition 3: The optimal feature evaluation index $\Theta(\boldsymbol{D})$ is defined as:

$$\Theta(\boldsymbol{D}) = \min_{\boldsymbol{P}} \left\| \frac{\boldsymbol{\psi}^T (\boldsymbol{\Phi W})(\boldsymbol{\Phi W})^T \boldsymbol{\psi}}{n^2} - \boldsymbol{A} \right\|_F^2 \tag{3}$$

where $\boldsymbol{A} = (a_{ij})_{c\times c}$ is inter class similarity matrix, $\boldsymbol{\psi} = (\varphi_{ij})_{n\times c}$ is feature and inter class correlation matrix. $\varphi_{ij} \in [0, 1]$ reflects the degree of correlation between sample $\boldsymbol{x}_i$ and the classification c_j. Here, we use the maximum information coefficient method [13] to solve φ_{ij}.

Maximum information coefficient the mutual information of $\boldsymbol{X} = \{x_i, i = 1, 2, \cdots, N\}$ and $\boldsymbol{Y} = \{y_i, i = 1, 2, \cdots, N\}$ is defined as:

$$MI(\boldsymbol{X}, \boldsymbol{Y}) = \sum_{x\in \boldsymbol{X}} \sum_{y \in \boldsymbol{Y}} p(x, y) \lg \frac{p(x, y)}{p(x)\, p(y)} \tag{4}$$

where $p(x, y)$ is the joint probability function of $\boldsymbol{X}$ and $\boldsymbol{Y}$, $p(x)$ and $p(y)$ are the marginal probability function of $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively. Divide the interval range of $\boldsymbol{X}$ into a segments, and $\boldsymbol{Y}$ into b segments. Thus, the joint space of $\boldsymbol{X}$ and $\boldsymbol{Y}$ is meshed into $a \times b$ grids. Estimate $p(x)$ and $p(y)$ with histogram to obtain $MI(\boldsymbol{X}, \boldsymbol{Y})_{a,b}$. Considering that there are various ways to mesh the joint space of $\boldsymbol{X}$ and $\boldsymbol{Y}$ into $a \times b$ grids, we define the maximum value of $MI(X, Y)$ as $MI(\boldsymbol{X}, \boldsymbol{Y})_{a,b}^{\max}$. Hence, the maximum information coefficient is defined as:

$$MIC(\boldsymbol{X}, \boldsymbol{Y}) = \max_{a\times b \le B(N)} \left\{ \frac{MI(\boldsymbol{X}, \boldsymbol{Y})_{a,b}^{\max}}{\log(a, b)} \right\} \tag{5}$$

where $B(N)$ is the maximum number of the meshed grids, usually we have $B(N) = N^{0.6}$. Since the maximum information coefficient is defined, we let $\varphi_{ij} = MIC(\boldsymbol{x}_i, c_j)$.

Conclusion 1: Feature subset selected by the optimal feature evaluation index $\Theta(\boldsymbol{D})$ defined in **Definition 3** can minimize the feature redundancy and optimize the original classification correlation information.

Proof: Perform standard centralization on all features in $\boldsymbol{\Phi}_{n\times m}$, $\sum_{i=1}^{n} f_{ij} = 0$, $\sum_{i=1}^{n} f_{ij}^2 = 1$. Let $\boldsymbol{A}' = n^2\boldsymbol{A}$, and $\boldsymbol{B} = \boldsymbol{\psi}^T (\boldsymbol{\Phi W})(\boldsymbol{\Phi W})^T \boldsymbol{\psi}$. Then, we have:

$$\begin{aligned}\Theta(\boldsymbol{D}) &= \min_{\boldsymbol{P}} \frac{1}{n^2} \left\| \boldsymbol{B} - \boldsymbol{A}' \right\|_F^2 \\ &= \min_{\boldsymbol{P}} \frac{1}{n^2} tr\left[(\boldsymbol{B} - \boldsymbol{A}')(\boldsymbol{B} - \boldsymbol{A}')^T \right] \\ \Rightarrow \Theta(\boldsymbol{D}) &= \min_{\boldsymbol{P}} \frac{1}{n^2} tr\left(\boldsymbol{B}^T\boldsymbol{B} + \boldsymbol{A}\prime T\boldsymbol{A}' - 2\boldsymbol{A}\prime T\boldsymbol{B} \right)\end{aligned} \tag{6}$$

Because $\boldsymbol{A}\prime T\boldsymbol{A}'$ is a constant matrix, if we want the value of $\Theta(\boldsymbol{D})$ to be the minimal, conditions $\min_{\boldsymbol{P}} tr(\boldsymbol{B}^T\boldsymbol{B})$ and $\max_{\boldsymbol{P}} tr(\boldsymbol{A}\prime T\boldsymbol{B})$ should be both satisfied. As for $\min_{\boldsymbol{P}} tr(\boldsymbol{B}^T\boldsymbol{B})$, we have

$$\begin{aligned}\min_{\boldsymbol{P}} tr\left(\boldsymbol{B}^T\boldsymbol{B}\right) &= \sum_{i,j=1}^{k} \left[\left(\widehat{\boldsymbol{f}}_i^T \boldsymbol{\psi}\right) \left(\widehat{\boldsymbol{f}}_j^T \boldsymbol{\psi}\right)^T \right]^2 \\ &= \sum_{i,j=1}^{k} \left(\widehat{\boldsymbol{f}}_i^T \left(\boldsymbol{\psi}\boldsymbol{\psi}^T\right) \widehat{\boldsymbol{f}}_j \right) \\ \Rightarrow \min_{\boldsymbol{P}} tr\left(\boldsymbol{B}^T\boldsymbol{B}\right) &= \sum_{i,j=1}^{k} \sum_{h}^{c} \left(\langle \widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h \rangle \times \langle \widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h \rangle \right)^2 \\ \Rightarrow \min_{\boldsymbol{P}} tr\left(\boldsymbol{B}^T\boldsymbol{B}\right) &= \sum_{i,j=1}^{k} \sum_{h}^{c} n^4 \sigma_{S_h}^4 \rho_{\widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h}^2 \rho_{\widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h}^2\end{aligned} \tag{7}$$

where $\boldsymbol{S}_h$ $(h = 1, 2, \cdots, c)$ represents the hth row of $\boldsymbol{\psi}_{n\times c}$, that is the corresponding elements of class c_h. $\sigma_{S_h}^2$ is the standard deviation of c_h, and $\rho_{\widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h}$ $(\rho_{\widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h})$ is the Pearson correlation coefficient of $\widehat{\boldsymbol{f}}_i$ and c_h. Equation (7) shows that $\sum_{h}^{c} \rho_{\widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h}^2 \rho_{\widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h}^2$ indicates the feature redundancy between $\widehat{\boldsymbol{f}}_i$ and $\widehat{\boldsymbol{f}}_j$. The value of $\min_{\boldsymbol{P}} tr(\boldsymbol{B}^T\boldsymbol{B}) = \sum_{i,j=1}^{k} \sum_{h}^{c} n^4 \sigma_{S_h}^4 \rho_{\widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h}^2 \rho_{\widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h}^2$ will achieve a minimum when $\min_{\boldsymbol{P}} \sum_{h}^{c} \rho_{\widehat{\boldsymbol{f}}_i, \boldsymbol{S}_h}^2 \rho_{\widehat{\boldsymbol{f}}_j, \boldsymbol{S}_h}^2$ reaches its smallest value. In other words, redundancy between different features is the minimum. For $\max_{\boldsymbol{P}} tr(\boldsymbol{A}\prime T\boldsymbol{B})$, we have:

$$\begin{aligned}\max_{\boldsymbol{P}} tr(\boldsymbol{A}\prime T\boldsymbol{B}) &= (\boldsymbol{\Phi W})^T \boldsymbol{\psi} \boldsymbol{A}' \boldsymbol{\psi}^T (\boldsymbol{\Phi W}) \\ &= \sum_{i=1}^{k} \widehat{\boldsymbol{f}}_i^T \left(\boldsymbol{\psi} \boldsymbol{A}' \boldsymbol{\psi}^T \right) \widehat{\boldsymbol{f}}_i \\ \Rightarrow \max_{\boldsymbol{P}} tr(\boldsymbol{A}\prime T\boldsymbol{B}) &= \sum_{i=1}^{k} \widehat{\boldsymbol{f}}_i^T \left(\sum_{e=1}^{c} \sum_{r=1}^{c} \boldsymbol{S}_e a_{er} \boldsymbol{S}_r^T \right) \widehat{\boldsymbol{f}}_i\end{aligned} \tag{8}$$

Seen from (8), $\sum_{e=1}^{c} \sum_{r=1}^{c} \boldsymbol{S}_e a_{er} \boldsymbol{S}_r^T$ represents the inter class similarity matrix and inter class correlation matrix to the maximum extent, which means that the original classification correlation has been maintained to its best.

Proved.

From **Definition 1** and **Definition 2**, it is clear that there's mutual correspondence between feature extraction vector $\boldsymbol{P}$

and feature extraction matrix. When all the subscripts j that satisfy $p_j = 1$ in $\boldsymbol{P}$ are determined, the exact expression of $\boldsymbol{W}$ can be fixed and then the feature subset can be extracted. But, (3) is an integer programming problem [13], which is a NP-Hard problem. We use intelligent optimization to solve it in the following sections.

B. CONSTRUCTION OF FEATURE SUBSET EXTRACTION MODEL

Intelligent optimization provides a new way to solve complex optimization problems. By simulating the behavior of natural creatures or the change of the physical properties of materials, it solves the optimization problem through iterative evolutionary computation. For this reason, based on social spider optimization algorithm [18], we introduce improved social spider optimization algorithm (ISSO) into the process of solving $\Theta(\boldsymbol{D})$ to obtain the optimal feature extraction vector $\boldsymbol{P}$. Intelligent optimization, in its essence, is random search. It has no fixed optimization result. To enhance the robustness and reliability of feature extraction, parallel computing is also employed to extract Q ISSO features at the same time. Then feature subset selection strategy is applied to sort $\boldsymbol{P}_i, i = 1, 2, \cdots, Q$, and $\boldsymbol{P}_{best}$ with better robustness, and improved classification performance is achieved. $\boldsymbol{P}_{best}$ and distributed fuzzy clustering are used together to ensure efficient detection of intrusion behavior. The detailed procedure of WSNs intrusion detection is shown in Fig.2.

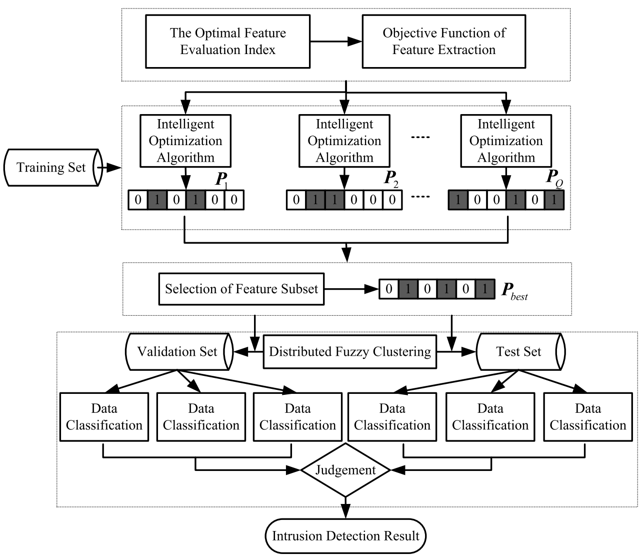


FIGURE 2. Flow chart of WSNs intrusion detection.

III. IMPLEMENTATION OF PARALLEL INTELLIGENT OPTIMIZATION FEATURE EXTRACTION

A. IMPROVED CLUSTERING SPIDER ALGORITHM

By simulating the cooperative behaviors of spiders, Cuevas proposed a novel swarm algorithm called the Social Spider Optimization (SSO) to solve optimization tasks. For SSO, the spider web of spider population is equivalent to the algorithm search space, and the spatial location of the spider represents a solution of the optimization problem. Through the constant co-evolution of female and male spiders, the problem is finally achieved. The SSO algorithm draws immediate attention due to its remarkable performance. However, SSO is easy to be trapped in the local optimization maximum, and has a low degree of convergence. Thus, based on SSO, we redefined the spider cluster optimization algorithm evolution rule by introducing local search and adaptive multi strategy update method. Firstly, an original particle swarm $\boldsymbol{G} = \{\boldsymbol{X}_i\}_{i=1}^N$ with the size of N is generated randomly in the solution space R^m. Each particle $\boldsymbol{X}_i = (x_{i1}, x_{i2}, \cdots, x_{im})$ represents a potential solution for the optimization. Then, perform iterative evolution on the particles. And finally, the optimization problem is solved. (Details of SSO can be found in related works.)

Adaptive Multi Strategy Update and Local Search: Researches have shown that the diversity of the population samples and the local extremum escaping ability are the key factors affecting the convergence performance of the intelligent optimization algorithm. Therefore, in the process of population evolution, various learning objects are given to the particles, so that they can adaptively adjust the updating strategy according to their evolution degree, so as to effectively maintain the diversity of the population samples.

Definition 4: In ISSO, the updating strategy of particle $\boldsymbol{X}_i(t)$ is defined as:

$$\boldsymbol{X}_i(t+1) = \begin{cases} \boldsymbol{X}_i(t) + \Delta, & r_1 \geq \kappa \\ SSO(\boldsymbol{X}_i(t)), & else \end{cases} \tag{9}$$

$$\Delta = \begin{cases} \omega_1 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_i(t)), & r_2 \leq \alpha_1 \\ \omega_2 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_{best}(t)) & \\ \quad + \omega_2 \otimes \left(\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_g(t)\right), & \alpha_1 < r_2 \leq \alpha_2 \\ \omega_3 \otimes \left(\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_j(t)\right), & \alpha_2 < r_2 < 1,\ i \neq j \end{cases} \tag{10}$$

where $SSO(\boldsymbol{X}_i(t))$ denotes that particle $\boldsymbol{X}_i(t)$ is iterative updated according to basic SSO, κ, α_1 and α_2 is the update control probability, r_1 and r_2 is (0,1) random, $A \leftrightarrow B$ represents A learn from B, $\omega_1 \otimes (\leftrightarrow)$ denotes learning degree control. The bigger the value of ω_1 is, the greater the impact of particle B has on the evolutionary direction of particle A. In order to further improve the convergence performance of IDSSO (improved discrete SSO), we divide $\boldsymbol{G}$ into multiple sub-populations $\boldsymbol{G}_j, j = 1, 2, \cdots, O$ according to particle fitness. Each sub-population set its α_1, α_2, ω_1, ω_2 and ω_3 adaptively, so as to adjust the learning object and learning degree dynamically. For sub-population $\boldsymbol{G}_j$, we have

$$\alpha_1(\omega_1) \propto \ln\left(\frac{\max_{X_i \in \boldsymbol{G}_j} f(X_i(t))}{f(X_{best}(t))} \frac{t}{T_{\max}} + 1\right)$$

$$\alpha_2(\omega_2, \omega_3) \propto \ln\left(2 - \frac{\max_{X_i \in \boldsymbol{G}_j} f(X_i(t))}{f(X_{best}(t))} \frac{t}{T_{\max}}\right) \tag{11}$$

in which $f(\cdot)$ is the objective function, $T_{\max}$ is the maximum number of iterations. From (11) we can see that, for $\boldsymbol{G}_j$ with better fitness, particles learn from themselves with a greater probability, which means evolution in the form of self-variation. And then, the deep search space of the algorithm is extended. Whereas $\boldsymbol{G}_j$ with worse fitness will result in particles learn with a larger probability from the population optimal solution and the historical optimal solution $\boldsymbol{X}_g(t)$, thus speeding up the evolution.

Algorithm 1 ISSO Algorithm

Input: Number of particles N, number of sub-populations O, number of features m, update control probability κ, objective function $f(\cdot)$, maximum number of iterations $T_{\max}$;

Randomly generate an initial population in the solution space, $t \leftarrow 1$;
for $i = 1 : N$ **do**
  Calculate $f(\boldsymbol{X}_i(t))$ and update $\boldsymbol{X}_g(t)$;
end for
while the terminate condition is not satisfied **do**
  Sort all the particles according to their fitness, and divide them into O sub-populations;
  for $j = 1 : O$ **do**
    Update particles in $\boldsymbol{G}_j$ according to (9)-(12).
    If the updated particles are superior to the original particles, replace the originals with the updated and update $\boldsymbol{X}_g(t)$. Or else, leave it unchanged;
  end for
  Update $\boldsymbol{X}_{best}(t)$, $t \leftarrow t + 1$;
end while

Output: Global optimal solution $\boldsymbol{X}_{best}$.

The time complexity for ISSO to evolve once is $O(N \log Om)$, where $O(N)$ is the time complexity of the population initialization. Thus, the total time complexity of ISSO is $T_{\max} O(N \log Om) + O(N)$.

Conclusion 2: ISSO is a global optimization algorithm.

Proof: Viewed from the time sequence, population $\boldsymbol{G}(t)$ constructs a discrete time stochastic process $\{\boldsymbol{G}(1), \boldsymbol{G}(2), \cdots \boldsymbol{G}(t), \cdots\}$. From **Definition 4**, the current state of $\boldsymbol{G}(t)$ is only related to its previous state, that is

$$p\{G_t | G_1, \cdots, G_{t-1}\} = p\{G_t | G_{t-1}\} \tag{12}$$

Thus, $\{\boldsymbol{G}(1), \boldsymbol{G}(2), \cdots \boldsymbol{G}(t), \cdots\}$ is a Markov chain. Here we have:

$$\begin{aligned} &f(\boldsymbol{X}_{best}(1)) \\ &\quad \geq f(\boldsymbol{X}_{best}(2)) \geq \cdots \geq f(\boldsymbol{X}_{best}(t)) \geq \cdots \end{aligned} \tag{13}$$

$$\begin{aligned} &f(\boldsymbol{X}_{best}(1)) - f(\boldsymbol{X}_{best}) \geq f(\boldsymbol{X}_{best}(2)) - f(\boldsymbol{X}_{best}) \\ &\quad \geq \cdots \geq f(\boldsymbol{X}_{best}(t)) - f(\boldsymbol{X}_{best}) \geq \cdots \end{aligned} \tag{14}$$

For any positive ε, define N_ε, the range of $\boldsymbol{X}_{best}$, as $N_\varepsilon = \{\boldsymbol{X}_i(t) | f(\boldsymbol{X}_i(t)) - f(\boldsymbol{X}_{best}) < \varepsilon\}, \boldsymbol{X}_i(t) \in \boldsymbol{G}(t)$. We also

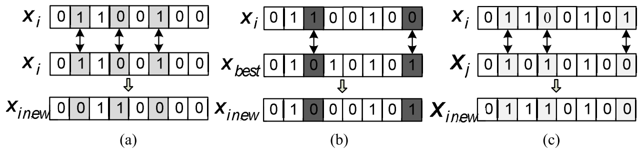


FIGURE 3. Particle update strategies of IDSSO. (a) Reverse strategy. (b) Replace strategy. (c) Exchage strategy.

define a sequence of random variables $\{\xi_1, \xi_2, \cdots, \xi_t, \cdots\}$, where ξ_t is expressed as

$$\xi_t = \begin{cases} 1, & \boldsymbol{X}_{best}(t) \in N_\varepsilon \\ 0, & \boldsymbol{X}_{best}(t) \notin N_\varepsilon \end{cases} \tag{15}$$

Let $P(\xi_t = 1) = p_t$, $P(\xi_t = 0) = 1 - p_t$, $\mathbb{Z}(t) = \frac{1}{t}\sum_{i=1}^{t} \xi_i$, then the expectation and variance of $\mathbb{Z}(t)$ can be written as

$$E(\mathbb{Z}(t)) = \frac{1}{t}\sum_{i=1}^{t} p_i, \quad D(\mathbb{Z}(t)) = \frac{1}{t^2}\sum_{i=1}^{t} p_i(1 - p_i) \leq \frac{1}{4t} \tag{16}$$

According to Chebyshev inequality,

$$P\{\mathbb{Z}(t) - E(\mathbb{Z}(t)) < \varepsilon\} \geq 1 - \frac{D(\mathbb{Z}(t))}{\varepsilon^2} \geq 1 - \frac{1}{4t\varepsilon^2} \tag{17}$$

Together with $P\{\mathbb{Z}(t) - E(\mathbb{Z}(t)) < \varepsilon\} \leq 1$, we have $\lim_{t\to\infty} P\{\mathbb{Z}(t) - E(\mathbb{Z}(t)) < \varepsilon\} = 1$, which indicates that $\{\xi_1, \xi_2, \cdots, \xi_t, \cdots\}$ is well convergent. When $t \to \infty$, $\boldsymbol{X}_{best}(t)$ is in an arbitrary small range around $\boldsymbol{X}_{best}$ with probability 1. Therefore, ISSO is a global optimization algorithm.

Proved.

B. IMPLEMENTATION OF FEATURE EXTRACTION BASED ON PARALLEL ISSO

Determining the exact form of feature extraction vector $\boldsymbol{P}$ is critical to feature extraction. So, we define the code of each ISSO particle as a corresponding extraction vector $\boldsymbol{P}$, written as $\boldsymbol{X}_i(t) = (x_{i1}, \cdots, x_{ij}, \cdots, x_{im})$, $x_{ij} \in \{0, 1\}$, $\boldsymbol{X}_i^T\mathbf{1} = k$. And select $\Theta(D)$ as the objective function $f(\cdot)$. According to **Definition 4** and the form of the particle code, the three update strategies in (10) are described in detail as follows.

Definition 4: Select $\lfloor m \times \omega_1 \rfloor$ bits out of the particle code $\boldsymbol{X}_i(t)$, and reverse these bits. This reverse strategy is defined as $\omega_1 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_i(t))$, as shown in Fig. 3(a).

Definition 5: If $\boldsymbol{X}_i(t)$ have $m'(0 \leq m' \leq m)$ bits code that are different from $\boldsymbol{X}_{best}(t)$, select $\lfloor m' \times \omega_2 \rfloor$ bits out of the m' bits, and replace the selected $\lfloor m' \times \omega_2 \rfloor$ bits in $\boldsymbol{X}_i(t)$ with the corresponding code in $\boldsymbol{X}_{best}(t)$. This replace strategy is defined as $\omega_2 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_{best}(t))$, as shown in Fig. 3(b).

Definition 6: Randomly select $\lfloor m \times \omega_3 \rfloor$ bits out of $\boldsymbol{X}_i(t)$, and exchange the these bits with the corresponding code in $\boldsymbol{X}_j(t)$. This exchange strategy is defined as $\omega_3 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_j(t))$, as shown in Fig. 3(c).

Selection of Parallel Feature Subset: Parallel computing is utilized in our algorithm to improve the robustness and reliability of feature extraction. Such strategy will enable us to run Q ISSO feature extraction processes simultaneously to calculate the optimal extraction vector set $\{\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_Q\}$.

Definition 7 (Evaluation of Classification): Use $\boldsymbol{P}_i, i \in \{1, 2, \cdots, Q\}$ to classify samples $\{\boldsymbol{x}_i\}_{i=1}^n$ in data matrix $\boldsymbol{D}$, and define the evaluation of classification as $DIV(\boldsymbol{P}_i)$:

$$DIV(\boldsymbol{P}_i) = \sum_{j=1}^{n}\left(\frac{\sum_{\boldsymbol{x}_z \in Mis_j}\|\boldsymbol{x}_j - \boldsymbol{x}_z\|}{n \times |Mis_j|}\right) - \sum_{j=1}^{n}\left(\frac{\sum_{\boldsymbol{x}_z \in Ne_j}\|\boldsymbol{x}_j - \boldsymbol{x}_z\|}{n \times |Ne_j|}\right) \quad (18)$$

where Mis_j and Ne_j are the different sample set and similar sample set of $\boldsymbol{x}_j$ respectively, $|Mis_j|$ and $|Ne_j|$ are the set scales.

Equation (18) indicates that $DIV(\boldsymbol{P}_i)$ reflects the sample classification quality of $\boldsymbol{P}_i$. The larger the value of $DIV(\boldsymbol{P}_i)$, the better the classify quality. The pseudocode of parallel ISSO feature extraction based on $DIV(\boldsymbol{P}_i)$ (PISSOF) is listed below.

Algorithm 2 PISSOF Algorithm

Input: Number of particles N, number of sub-populations O, number of features m, update control probability κ, objective function $\Theta(\boldsymbol{D})$, maximum number of iterations $T_{\max}$, number of parallel feature extraction process Q;

1: Run Q ISSO optimization feature extraction procedures simultaneously, and get $\{\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_Q\}$;
2: **for** $i = 1 : Q$ **do**
3: According to $\boldsymbol{P}_i$, use SVM to classify data matrix $\boldsymbol{D}$;
4: Calculate $DIV(\boldsymbol{P}_i)$ according to (18);
5: **end for**
6: $\boldsymbol{P}_{best}$ is the maximum value of $DIV(\boldsymbol{P}_i)$;

Output: $\boldsymbol{P}_{best}$

IV. INTRUSION DETECTION BASED ON FEATURE EXTRACTION AND DISTRIBUTED FUZZY CLUSTERING

A. DISTRIBUTED FUZZY CLUSTERING

Fuzzy C-means clustering algorithm (FCM) classifies the data matrix through membership function. Suppose there are n samples in data set $\{\boldsymbol{x}_i\}_{i=1}^n$. FCM calculate the membership matrix $\boldsymbol{U} = [\mu_{ik}]_{C\times n}$ and cluster center $\boldsymbol{V} = \{\boldsymbol{v}_i\}$ through iteration to divide $\{\boldsymbol{x}_i\}_{i=1}^n$ into C sub-classes and make the clustering objective function minimal, here μ_{ik} is the membership degree of sample $\boldsymbol{x}_k$ to the ith sub-class. However, FCM is easy to fall into local optimum due to its local search strategy and sensitivity to initial value settings. Besides, the number of sub-classes C should also be set in advance for FCM. Hence, we present distributed fuzzy clustering to improve the fuzzy clustering algorithm performance through introducing intelligent iterative evolution method and adaptive clustering strategy.

Definition 8: In ISSO algorithm, coded particles as:

$$\boldsymbol{X}_i(t) = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_i, \cdots \boldsymbol{v}_C), \quad \boldsymbol{v}_i = (v_{i1}, v_{i2}, \cdots v_{ik}) \quad (19)$$

where $\boldsymbol{v}_i, i = 1, 2, \cdots, C$ is the ith cluster center, $(v_{i1}, v_{i2}, \cdots v_{ik})$ is the eigenvalues corresponding to non-zero elements of the optimal feature extraction vector $\boldsymbol{P}_{best}$. During the iterative process of the algorithm, particles are still updated by strategy defined in **Definition 4**. And for continuous optimization problem, the update strategy is written as:

$$\begin{cases} \omega_1 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_i(t)) = \omega_1 \times \boldsymbol{X}_i(t) \\ \omega_2 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_{best}(t)) = \omega_2 \times (\boldsymbol{X}_{best}(t) - \boldsymbol{X}_i(t)) \\ \omega_3 \otimes (\boldsymbol{X}_i(t) \leftrightarrow \boldsymbol{X}_j(t)) = \omega_3 \times (\boldsymbol{X}_j(t) - \boldsymbol{X}_i(t)) \end{cases} \quad (20)$$

Definition 9: Define the objective function of ISSO as:

$$\min f = 2\sum_{i=1}^{C}\sum_{j=1}^{n}\mu_{ij}^{\partial}\left[1 - K(\boldsymbol{x}_j, \boldsymbol{v}_i) + (\boldsymbol{v}_i, \boldsymbol{v}_i)\right]$$
$$K(\boldsymbol{x}_j, \boldsymbol{v}_i) = \exp\left(\frac{-\|\boldsymbol{x}_j - \boldsymbol{v}_i\|^2}{\sigma^2}\right) \quad (21)$$

where ∂ is fuzzy weighting exponent. The essence of (21) is to replace the traditional Euclidean distance by the Gauss kernel induced distance so as to expand the application scope of fuzzy clustering algorithms.

Distributed ISSO Optimization: To further increase the clustering performance, MPI parallel is used to run L ISSO optimization fuzzy clustering classification progresses simultaneously. After each iteration, the L progresses would lead to L optimum solutions $\{\boldsymbol{X}_{best}^i(t)\}_{i=1}^L$. If the inequality $\left|\max_{i=1:L} f(\boldsymbol{X}_{best}^i(t)) - \min_{i=1:L} f(\boldsymbol{X}_{best}^i(t))\right| \geq \tau$ is satisfied, solutions with poor-fitness will be substituted by N_{mpi} high-fitness solutions randomly selected from $\{\boldsymbol{X}_{best}^i(t)\}_{i=1}^L$.

Optimal Clustering: To automatically determine the clustering number C, clustering validity index $VS(\boldsymbol{U}, \boldsymbol{V})$ is defined as:

$$VS(\boldsymbol{U}, \boldsymbol{V}) = \frac{\left[\sum_{i=1}^{C}\sum_{j=1}^{n}\frac{\mu_{ik}d^2(\boldsymbol{x}_j, \boldsymbol{v}_i)}{n(i)}\right]\sqrt{\left(\frac{C+1}{C-1}\right)}}{\left[1 - \max_{i\neq j}\left(\max_{i\neq j}\left(\min\left(\mu_{ik}, \mu_{jk}\right)\right)\right)\right]} \quad (22)$$

$$d(\boldsymbol{x}_j, \boldsymbol{v}_i) = \sqrt{1 - \exp\left(-\beta\|\boldsymbol{x}_j - \boldsymbol{v}_i\|^2\right)},$$
$$\beta = \left(\frac{1}{n}\sum_{j=1}^{n}\left\|\boldsymbol{x}_j - \left(\frac{1}{n}\sum_{j=1}^{n}\boldsymbol{x}_j\right)\right\|^2\right)^{-1} \quad (23)$$

where, $n(i)$ is the ith cluster scale. From (22) and (23) we can see that $VS(\boldsymbol{U}, \boldsymbol{V})$ is able to effectively evaluate the homogeneity and separation of cluster sub-classes. The smaller the value of $VS(\boldsymbol{U}, \boldsymbol{V})$ is, the better the clustering performance.

Algorithm 3 Distributed ISSO Fuzzy Clustering Algorithm

Input: Number of particles N, number of sub-populations O, number of features m, update control probability κ, maximum number of iterations $T_{\max}$, fuzzy weighting exponent ∂, maximum clustering number $C_{\max}$, optimal feature extraction vector $\boldsymbol{P}_{best}$, number of parallel clustering processes L;

```
For each sample x_i, extract its optimum feature subset X_i = (f̂_i1, f̂_i2, ..., f̂_ik) according to the position of non-zero elements in P_best;
for C = 2 : C_max do
   for l = 1 : L do
      Initialization of populations G_l. According to Definition 8, coded every particle in G_l with k samples. Randomly select a particle to be X_best (0), and calculate the initial fitness value of each particle, t ← 1;
      while the terminate condition is not satisfied do
         Perform ISSO on every particle in population G_l once according to (20);
         for i = 1 : N do
            Use μ_ik = ||x_k − v_i||^(−2/(∂−1)) / ( Σ_{j=1}^{C} ||x_k − v_j||^(−2/(∂−1)) ) to calculate U, and calculate f (X_i (t)) according to (21);
            Update historical optimal solution for particle and optimal solution X^l_best (t) for population;
         end for
         if | max_{i=1:L} f (X^i_best (t)) − min_{i=1:L} f (X^i_best (t)) | ≥ τ then
            if X^l_best (t) belongs to the N_mpi poor solutions, substitute it with a better solution randomly selected from other populations;
         end if
         t ← t + 1;
      end while
   end for
   Collect the L optimal solutions {X^i_best (t)}_{i=1}^{L}, and choose the best fitted solution as cluster center V_c;
   Calculate VS_C (U, V) according to (22);
end for
The C corresponding to the minimal value of VS_C (U, V) is the optimal cluster number, the corresponding cluster center is the optimal V
```

Output: cluster center $\boldsymbol{V} = \{\boldsymbol{v}_i\}$

B. IMPLEMENTATION OF INTRUSION DETECTION IN WSNS

Perform **Algorithm 2** on data set $\left\{\boldsymbol{x}_i^{text}\right\}_{i=1}^{n}$, we obtain the optimal feature extraction vector $\boldsymbol{P}_{best}$. Then C cluster centers $\boldsymbol{V} = \left\{\boldsymbol{v}_j\right\}, j = 1, 2, \cdots, C$ based on $\boldsymbol{P}_{best}$ can be attained by **Algorithm 3**. Classify test data $\boldsymbol{x}_i^{text}$ to the nearest cluster center. When all the data have been classified, there are C classes $\boldsymbol{D}_c^{text}$, $c = 1, 2, \cdots, C$. Because the number of abnormal data is small, $\boldsymbol{D}_c^{text}$, which contain a small number of data, is very likely to be invaded. To determine whether the data is abnormal, suspected $\boldsymbol{D}_c^{text}$ is compared with validation data classes in sequence. $\boldsymbol{D}_c^{text}$ can be said to be abnormal if the following inequality is satisfied.

$$\min_{H=1:C'} \left| \frac{\sum\limits_{\boldsymbol{x}_i^{text} \in \boldsymbol{D}_c^{text}} \left(\boldsymbol{x}_i^{text} - \boldsymbol{v}_c^{text}\right)}{\left|\boldsymbol{D}_c^{text}\right|} - \frac{\sum\limits_{\boldsymbol{x}_j \in \boldsymbol{D}_H} \left(\boldsymbol{x}_j - \boldsymbol{v}_H\right)}{\left|\boldsymbol{D}_H\right|} \right| \leq \theta \tag{24}$$

where C' is the number of validation data clusters, $\boldsymbol{v}_c^{text}$ and $\boldsymbol{v}_H$ are the cluster center of test data classes $\boldsymbol{D}_c^{text}$ and validation data classes $\boldsymbol{D}_H$ respectively, $|\boldsymbol{D}_H|$ is the scale of data in each class.

V. EXPERIMENTAL ANALYSIS AND SIMULATION RESULTS

A. PERFORMANCE OF PARALLEL ISSO FEATURE EXTRACTION ALGORITHM

In this part, real data is used to verify the performance of PISSOF we proposed in this paper, they are Ionosphere (ION), Hill-Valley (Hil), Arrythmia (Ary), Madelon, Multiple-features (Mfe), Breast, ORL10P and Dexter selected from [8] (the parameters of the 8 real data are shown in Table 1). Typical feature extraction algorithms, ReliefF, mRMR, SPEC [7], CFS [13], CIP [8], FCBF [13] and FSBR [14], are also simulated as comparative algorithms of PISSOF. Two classifiers, KNN and SVM, are utilized individually to carry out classification test for feature subsets obtained from PISSOF and the aforementioned 7 feature extraction algorithms on MATLAB. Each algorithm runs 50 times. And evaluation analysis is based on average correct rate of classification $CAC(\boldsymbol{P}_i)$ and average $DIV(\boldsymbol{P}_i)$ respectively.

TABLE 1. Test data.

	m	n	c
ION	34	351	2
Hil	100	1212	10
Ary	279	452	16
Madelon	500	2000	2
Mfe	769	3428	10
Breast	9216	84	5
ORL10P	10304	100	10
Dexter	20000	300	2

Parameters in PISSOF are set as follows: $N = 200$, $O = 10$, $\kappa = 0.25$, $T_{\max} = 500$, $Q = 5$. Under the same parallel computing framework, perform discrete particle swarm optimization algorithm (DPSO) and discrete artificial bee colony algorithm (DABC) separately to solve the feature extraction model. Table 2 and Table 3 are the

TABLE 2. Comparison of *CAC* (P_i).

data	SVM(/%)								KNN(/%)							
	ReliefF	mRMR	SPEC	CFS	CIP	FCBF	FSBR	PISSOF	ReliefF	mRMR	SPEC	CFS	CIP	FCBF	FSBR	PISSOF
ION	88.42	86.07	89.57	90.15	91.20	90.87	89.96	**92.04**	80.74	81.27	84.44	83.75	**88.29**	86.73	80.91	87.48
Hil	58.12	60.15	69.27	57.23	77.14	76.89	69.82	**79.43**	55.41	53.74	59.84	60.17	56.38	51.49	50.22	**67.94**
Ary	68.75	69.44	70.28	66.53	70.15	71.08	70.38	**74.21**	77.43	75.61	74.28	70.49	78.24	74.35	76.42	**80.19**
Madelon	63.77	64.21	65.10	60.24	66.79	65.39	64.02	**69.88**	66.11	67.21	60.54	68.15	66.97	61.72	62.37	**70.24**
Mfe	96.02	94.21	97.62	95.11	97.16	98.27	98.04	**98.71**	94.01	93.27	91.11	94.56	95.66	97.28	94.86	**98.17**
Breast	90.36	94.11	95.20	94.13	95.55	94.81	94.78	**97.56**	94.28	91.56	90.18	94.37	95.27	94.11	93.10	**96.14**
ORL10P	95.23	96.71	97.24	95.08	90.73	91.24	96.34	**98.80**	91.24	91.53	92.77	96.37	95.42	91.08	91.07	**97.82**
Dexter	91.27	90.55	93.88	94.17	**95.41**	90.51	91.74	92.81	90.47	92.11	91.24	94.00	94.17	95.37	94.11	**96.83**

TABLE 3. Comparison of *DIV* (P_i).

data	ReliefF	mRMR	SPEC	CFS	CIP	FCBF	FSBR	PISSOF
ION	113.47	98.14	44.82	76.71	42.84	33.46	47.15	**29.47**
Hil	0.97	0.82	0.76	0.54	**0.17**	0.55	0.41	0.19
Ary	5.74	4.21	3.49	2.74	1.84	2.41	3.47	**1.07**
Madelon	8.10	7.54	9.24	7.55	6.94	7.61	8.04	**4.33**
Mfe	22.17	**10.13**	14.82	33.14	29.05	23.37	20.74	19.82
Breast	99.45	107.52	111.46	124.08	76.34	86.17	99.30	**67.24**
ORL10P	44.17	29.71	19.42	17.33	20.18	21.46	33.01	**15.73**
Dexter	44.13	27.06	19.44	21.97	34.18	36.81	16.44	**10.52**

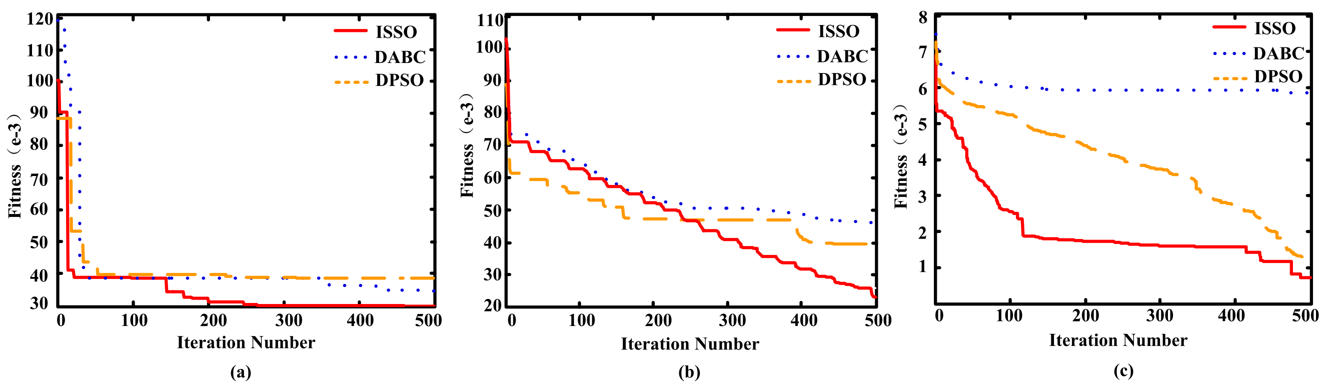


FIGURE 4. Comparison of the feature extraction convergence curves. (a) Test data Ion. (b) Test data Mfe. (c) Test data Ary.

comparison results of *CAC* $(\boldsymbol{P}_i)$ and *DIV* $(\boldsymbol{P}_i)$ for different feature extraction algorithms. Fig. 4 gives a comparison of the convergence curves between ISSO and the other two discrete algorithms.

For SVM, PISSOF has a better overall performance than the other 7 feature extraction algorithms, as shown in Table 2. Except for test data Dexter, the classification accuracy of PISSOF is the best. The *CAC* $(\boldsymbol{P}_i)$ of PISSOF has increased by 8.9%~15.6% compared with the other 7 feature extraction algorithms. Similarly to KNN, besides test data ION, the PISSOF has the best classification rate. The comparison results indicate that the feature subset selected by PISSOF has better classification performance. From Table 3, it can be concluded that the redundancy between extracted features has been effectively reduced by PISSOF, much lower than ReliefF, SPEC, FCBF, FSBR, etc. Obviously, CIP and mRMR also has small inter-feature redundancy, which is a direct result of their starting point of feature extraction, reduce the redundancy. To sum up, unilateral reduction of redundancy does not necessarily improve the success rate of feature subset classification. Only by reducing the redundancy while preserving the diversity of the original samples, can the classification and identification performs better.

Fig. 4 shows that ISSO has better convergence precision and efficiency. Multiple evolution updating strategies and

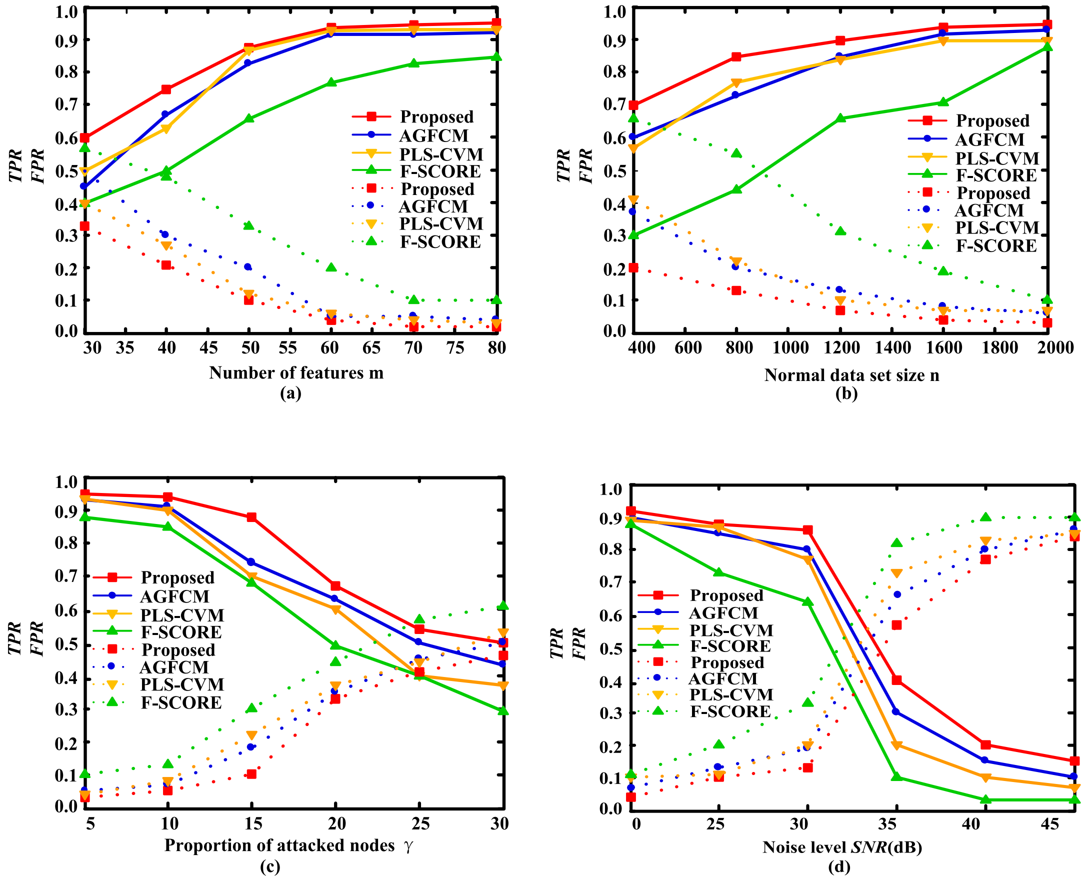


FIGURE 5. Comparison of *TPR* and *FPR*.

adaptive learning enables ISSO to avoid the local optimum for higher accuracy.

B. PERFORMANCE OF WSNS INTRUSION DETECTION ALGORITHM

200 nodes were deployed on a $100 \times 100m$ ground. Select LEACH and IEEE802.15.4 as the routing protocol and MAC protocol respectively. Continuously collect network data in a given time. Parameters of the simulation are listed in Table 4. The data feature index is composed of network basic connection, network connection content and network flow. And packet loss rate, message transmission frequency, message reception frequency, energy consumption rate and sensor measurement value are also investigated as main feature index. For each intrusion simulation, we randomly select different number of nodes as the attacked nodes. Attack types are mainly wormhole, black hole, and flooding. The experimental data is divided into normal data set and test data set that were attacked.

Similar to the WSNs intrusion detection algorithm proposed in this paper, AGFCM [15], PLS-CVM [20]–[27] and classical F-Score classify and detect data on the basis of feature subset extraction. So we compare the detection performance of our algorithm and the three algorithms for further analysis. We simulate the four algorithms under different feature number m, normal data set size n, proportion of attacked nodes γ and noise level SNR. Each algorithm runs 20 times. Comparisons and analysis were between average detection rate TPR and average false detection rate FPR. Fig. 5 is the comparison result, and Table 5 lists the results of feature subset selection and the time cost.

TABLE 4. Parameters for WSNs simulation.

Parameter	Value
Bit rate/Kbps	250
Data acquisition interval/s	120
Packet size/Byte	128
Transmission power/mW	1

TABLE 5. Comparison of feature subset selection result and time cost.

		AGFCM	PLS-CVM	F-Score	Algorithm proposed		
					$L=5$	$L=10$	$L=15$
m=40	k	23	21	17	13	12	14
	t /s	11.38	15.42	8.77	12.46	13.01	11.98
m=80	k	41	46	38	21	18	20
	t /s	22.14	30.82	18.37	24.08	25.37	29.14

Fig. 5(a) is the TPR and FPR curve over m when $n = 2000$, $\gamma = 5\%$, $SNR = 20db$. As it can be seen, the TPR of the four algorithms increased with the increase of m, while the FPR decreased. Especially when $m \geq 70$, the TPR of the algorithm we propose will reach beyond 95%, better than the other three algorithms. Fig. 5(b) is the TPR and FPR curve over n when $m = 70$, $\gamma = 5\%$, $SNR = 20db$. With the increase of n, the TPR of our algorithm improves obviously and the FPR decreases sharply, which means that our algorithm is superior than the other three algorithms on detection performance. And we have the conclusion that the participation of normal validation set is helpful for the improvement of detection accuracy. Fig. 5(c) is the TPR and FPR curve over γ when $m = 70$, $n = 2000$, $SNR = 20db$. It is clear that with more nodes being attacked, TPR declines. But even when 30% of the nodes were attacked, the TPR of our algorithm is still above 50%, rather better than the other 3 algorithms. Fig. 5(d) is the TPR and FPR curve over SNR when $m = 70$, $n = 2000$, $\gamma = 5\%$. When the noise level is low, all the four algorithms perform well. However, when the SNR exceeds 35dB, all of the algorithms suffer dramatic decline of detection accuracy, but our algorithm declines with a slower tendency compared with the other three algorithms, which means our algorithm is more robust against noise.

As Table 5 shows, for different m, the extracted feature subset size k of our algorithm are 13, 12, 14, 21,18, 20 respectively, the whole change is not very obvious, while the other algorithms' k almost doubled, indicating that the algorithm we proposed has better stability. The time cost of our algorithm is better than PLS-CVM, much the same as that of AGFCM, for they all adopts fuzzy clustering technique. It also can be seen that increase the number of parallel operations cannot necessarily guarantee a significantly increased time cost of our algorithm, which means that our algorithm has a high calculation efficiency [28]–[31].

C. SIMULATION RESULTS ANALYSIS

In summary, PISSOF feature extraction algorithm and intrusion detection algorithm has better performance.

1. Feature subset obtained by PISSOF classifies data set more precisely. The introduction of the optimal feature evaluation index reduced the inter-feature redundancy while maintaining the diversity of original data. And the parallel ISSO further improved the solving efficiency of feature subset extraction model.

2. The intrusion detection algorithm we present in this paper can effectively detect intrusions, and the intrusion detection rate is improved by about 13.1%, and the false detection rate is decreased by about 8.5%. The performance improvement is because the best feature subset and adaptive clustering strategy makes the algorithm more easily to identify intrusion behavior.

3. For intrusion detection of high dimensional complex data, the intrusion detection rate can be raised by the increase of the size of normal validation data set and the number of feature sets.

VI. CONCLUSION AND FUTURE SCOPE

This paper investigates the large-scale complex WSNs intrusion detection, and proposes an intrusion detection algorithm based on parallel intelligent optimization feature extraction and distributed fuzzy clustering in WSNs. The algorithm utilizes techniques such as feature extraction, intelligent optimization and fuzzy clustering, combined with the actual demand of WSNs intrusion detection, to improve the detection performance. Simulation experiments also verify the effectiveness of the proposed algorithm from many aspects. However, the WSNs experiment deployment in this paper is relatively stable, and the intrusions that happened were mainly deliberately set in advance. Therefore, in our future work, we intend to consider actual deployment of WSNs, and focus on the success rate of small sample detection.

REFERENCES

[1] T. Wang, Z. A. Bhuiyan, G. J. Wang, A. Rahman, J. Wu, and J. N. Cao, "Big data reduction for a smart city's critical infrastructural health monitoring," *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 128–133, Mar. 2018.
[2] A. Aziz, "Efficient compressive sensing based technique for routing in wireless sensor networks," *INFOCOMP J. Comput. Sci.*, vol. 12, no. 1, pp. 1–9, 2013.
[3] L. N. Wang and T. T. Wang, "Application of network intrusion detection based on two-steps fuzzy clustering," *Microelectron. Comput.*, vol. 31, no. 3, pp. 70–74, 2014.
[4] X. Li, H. Wang, H.-N. Dai, Y. Y. Wang, and Q. Zhao, "An analytical study on eavesdropping attacks in wireless nets of things," *Mobile Inf. Syst.*, vol. 2016, no. 1, pp. 1–10, 2016.
[5] T. Wang *et al.*, "Fog-based storage technology to fight with cyber threat," *Future Gener. Comput. Syst.*, vol. 83, pp. 208–218, Jun. 2018.
[6] S. Elhag, A. Fernández, A. Bawakid, S. Alshomrani, and F. Herrera, "On the combination of genetic fuzzy systems and pairwise learning for improving detection rates on intrusion detection systems," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 193–202, 2015.
[7] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 91, nos. 1–2, pp. 273–324, 1997.
[8] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.
[9] Z. Zhao, L. Wang, H. Liu, and J. Ye, "On similarity preserving feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 619–632, Mar. 2013.
[10] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington, DC, USA, 2003, pp. 856–863.
[11] D.-C. Feng, F. Chen, and W.-L. Xu, "Detecting local manifold structure for unsupervised feature selection," *Acta Autom. Sin.*, vol. 40, no. 10, pp. 2253–2261, 2014.
[12] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 301–312, Mar. 2002.
[13] D. N. Reshef *et al.*, "Detecting novel associations in large data sets," *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.
[14] G. L. Sun *et al.*, "Feature selection method based on maximum information coefficient and approximate Markov blanket," *Acta Autom. Sin.*, vol. 43, no. 5, pp. 795–805, 2017.
[15] L. K. Qiu *et al.*, "Feature selection algorithm based on redundancy analysis," *J. Beijing Univ. Posts Telecommun.*, vol. 40, no. 1, pp. 36–41, 2017.
[16] C. Tang, P. Liu, S. Tang, and Y. Xie, "Anomaly intrusion behavior detection based on fuzzy clustering and features selection," *J. Comput. Res. Develop.*, vol. 52, no. 3, pp. 718–728, 2015.
[17] J. Nayak, B. Naik, D. P. Kanungo, and H. S. Behera, "A hybrid elicit teaching learning based optimization with fuzzy c-means (ETLBO-FCM) algorithm for data clustering," *Ain Shams Eng. J.*, vol. 9, no. 3, pp. 379–393, 2018.
[18] E. Cuevas, M. Cienfuegos, D. Zaldívar, and M. Pérez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider," *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6374–6384, 2013.

[19] A. Ma, Y. Zhong, and L. Zhang, "Adaptive multiobjective memetic fuzzy clustering algorithm for remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4202–4217, Aug. 2015.

[20] L.-Y. Wu, S.-L. Li, X.-S. Gan, and M.-H. Wang, "Network anomaly intrusion detection CVM model based on PLS feature extraction," *Control Decis.*, vol. 32, no. 4, pp. 755–758, 2017.

[21] W. Wei, X. Fan, H. Song, X. Fan, and J. Yang, "Imperfect information dynamic Stackelberg game based resource allocation using hidden Markov for cloud computing," *IEEE Trans. Services Comput.*, vol. 11, no. 1, pp. 78–89, Jan./Feb. 2018, doi: 10.1109/TSC.2016.2528246.

[22] W. Wei, H. Song, W. Li, P. Shen, and A. Vasilakos, "Gradient-driven parking navigation using a continuous information potential field based on wireless sensor network," *Inf. Sci.*, vol. 408, pp. 100–114, Oct. 2017, doi: 10.1016/j.ins.2017.04.042.

[23] W. Wei *et al.*, "GI/Geom/1 queue based on communication model for mesh networks," *Int. J. Commun. Syst.*, vol. 27, no. 11, pp. 3013–3029, Nov. 2014.

[24] W. Wei, Z. Sun, H. Song, H. Wang, X. Fan, and X. Chen, "Energy balance-based steerable arguments coverage method in WSNs," *IEEE Access*, vol. 6, pp. 33766–33773, Mar. 2017, doi: 10.1109/ACCESS.2017.2682845.

[25] W. Wei, H. Song, H. Wang, and X. Fan, "Research and simulation of queue management algorithms in ad hoc networks under DDoS attack," *IEEE Access*, vol. 5, pp. 27810–27817, 2017, doi: 10.1109/ACCESS.2017.2681684.

[26] W. Wei, X. Fan, H. Song, and H. Wang, "Video tamper detection based on multi-scale mutual information," *Multimedia Tools Appl.*, pp. 1–18, Sep. 2017, doi: 10.1007/s11042-017-5083-1.

[27] W. Wei and Y. Qi, "Information potential fields navigation in wireless ad-hoc sensor networks," *Sensors*, vol. 11, no. 5, pp. 4794–4807, 2011.

[28] W. Wei, X.-L. Yang, B. Zhou, J. Feng, and P.-Y. Shen, "Combined energy minimization for image reconstruction from few views," *Math. Problems Eng.*, vol. 2012, Oct. 2012, Art. no. 154630.

[29] W. Wei, X.-L. Yang, P.-Y. Shen, and B. Zhou, "Holes detection in anisotropic sensornets: Topological methods," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 10, p. 135054, Oct. 2012.

[30] W. Wei, Y. Qiang, and J. A. Zhang, "A bijection between lattice-valued filters and lattice-valued congruences in residuated lattices," *Math. Problems Eng.*, vol. 2013, Jul. 2013, Art. no. 908623.

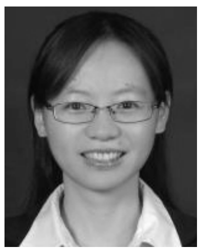
[31] W. Wei, H. M. Srivastava, Y. Zhang, L. Wang, P. Shen, and J. Zhang, "A local fractional integral inequality on fractal space analogous to Anderson's inequality," *Abstract Appl. Anal.*, vol. 2014, Jun. 2014, Art. no. 797561.

ZHOUZHOU LIU received the bachelor's degree in information engineering, the master's degree in telecommunication engineering, and the Ph.D. degree in information engineering from Northwestern Polytechnical University, China, in 2004, 2007, and 2016, respectively. He is currently a Professor and the Head of the Sensor network and Intelligent Computing Laboratory, Xi'an Aeronautical University. His current research interests include performance evaluation in mobile communications, random access in mobile radio networks, and ad hoc and sensor networks.

WEI WEI (SM'17) received the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 2005 and 2011, respectively. He is currently an Associate Professor with the School of Computer Science and Engineering, Xi'an University of Technology. He ran many funded research projects as principal investigator and technical members. He has published around 100 research papers in international conferences and journals. His research interest is in the area of wireless networks, wireless sensor networks application, image processing, mobile computing, distributed computing, and pervasive computing, Internet of Things, and sensor data clouds. He is a senior member of CCF. He is a TPC member of many conferences and regular reviewer of IEEE TPDS, TIP, TMC, TWC, and many other Elsevier journals.

HAO WANG received the B.Eng. and Ph.D. degrees in computer science in 2000 and 2006, respectively. **He** is currently an Associate Professor and the Head of the Big Data Laboratory, Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Norway. His research interests include wireless communication and security, big data analytics and industrial Internet of Things, high performance computing, and safety-critical systems.

YANGMEI ZHANG received the B.S. degree from the Air Force Engineering University of Communication Engineering in 2004, the M.S. degree from the Xi'an University of Science and Technology of Signal and Information Processing in 2008, and the Ph.D. degree in communication and information engineering from Northwestern Polytechnical University, Xi'an, China, in 2017. She is currently a Lecturer with Xi'an Aeronautical University. Her main research interests are signal detection and signal processing.



QIANYUN ZHANG received the B.S. degree in measurement technology and instruments from Nanchang Hangkong University in 2007 and the M.S. degree in measurement technology and instruments from the Nanjing University of Aeronautics and Astronautics in 2011. She is currently a Lecturer with Xi'an Aeronautical University. Her research area is structural health monitoring signal and information processing.

SHINING LI received the B.Eng. and Ph.D. degrees in computer science in 1992 and 2005, respectively. He is currently a Professor and the Head of Computer Systems and Microelectronics, Northwestern Polytechnical University. His research interests include wireless communication and security, big data analytics and industrial Internet of Things, and intelligent computing.

• • •