Electrical Power and Energy Systems 91 (2017) 183–191

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Improved-ELM method for detecting false data attack in smart grid

Liqun Yang [a,1], Yuancheng Li [b,1], Zhoujun Li [a,*]

[a] State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, Chian
[b] School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China

ARTICLE INFO

Article history:
Received 9 November 2016
Received in revised form 13 February 2017
Accepted 23 March 2017

Keywords:
Smart grid
False data injection attack
Dimension reduction
Extreme learning machine (ELM)

ABSTRACT

Power grid is a complex system which closely links the power generation and power consumer through transmission and distribution networks. With the development of smart grid, smart grid is more open to external communication systems, it also has exposed some problems in the network attacks. A new false data injection attack (called the unobservable attack) that can bypass the traditional BDD and inject random errors into state estimation. We propose an improved extreme learning machine (ELM) for attack detection. The artificial bee colony (ABC) incorporates the thought of differential evolution algorithm (DE) to optimize ELM for improving detection precision. In this paper, Autoencoder is used to reduce the dimensionality of the measurement data, which makes the low-dimensional data information basically and fully represent high-dimensional data. We verify the performance of the proposed method on IEEE bus systems, and prove that the proposed method can effectively detect such unobservable attack.

1. Introduction

With the power grid changing to smart grid, it provides reliable, inexpensive and sustained power services. Accurate state estimation is of paramount importance to maintain normal operations of smart power grid. But the unobservable false data injection attack can bypass the bad data detection (BDD), make control center do a series of wrong decisions, and result in the imbalanced load distribution of power grid [1–3].

The generation capacity should closely follow the electricity consumption because electrical energy cannot be stored in large amount. The voltage mismatch and abnormal consumption will cause the migration of power. For example, load fluctuations in the power system will change the operating frequency and voltage level of the power grid [4–8]. Therefore, the Supervisory Control and Data Acquisition (SCADA) in the Energy Manage System (EMS) must intensively monitor and control the power system to ensure the operations of power system are safe and reliable. Accurate state estimation makes the monitored systems running in the best condition. Moreover, the constrained economic dispatch (SCED) based on estimated states, which can reschedule power output and reduce the total cost output of the system operation [9].

Different from other communication networks, the measurement data in smart grid is always passed to control center through the way of information exchange, and the erroneous measurements may interfere with the control center to make unreasonable state estimation. It can be seen that malicious attacks against state estimation cause a great threat to power grid. The different incentives of attack can be summed up as follows: (1) Disturbing electricity market order and stealing electricity of end users, which will bring social and economic impacts; (2) Disrupting electric system, which will result in the imbalances of power dispatch and control. Recent studies show that the attacker can inject the carefully synthetic false data into measurements, and bypass traditional detection to cause serious problems in power system. It would lead to a wider load reduction and block power system from entering safe operating status. Once the injected false data affects the state estimation results, the intelligent control algorithm may be misled and large scale regional power outages will be produced ultimately.

False data injection attack (FDIA) is one of the most threatened attack to smart grid, and it's a Cyber-Physics fusion attack [10,11]. On the one hand, the attacker can use web hacking techniques to invade the information and communication systems of smart grid. On the other hand, the attacker can determine which data of instruments should be tampered to achieve the purpose of destruction. Now, some methods are put forward to deal with FDIA, these

* Corresponding author at: 37 Xueyuan Road, HaiDian District, Beijing 100191, China.
E-mail address: lizj@buaa.edu.cn (Z. Li).
[1] Liqun Yang and Yuancheng Li contributed equally to this work and should be considered co-first authors.

http://dx.doi.org/10.1016/j.ijepes.2017.03.011
0142-0615/

methods can be divided into two categories: Detection based methods, and Active protection based methods. These former methods could discover the abnormal value by determining whether the measurements meet the historical distribution of the measured data [12–17], but if the attacker use measured data conformed to previous historical data distribution instead of current data, the detection method will be invalid. Active protection based methods protect specific sensors to resist the false data injection attacks in paper [18–21], but this method exists many shortcomings. Firstly, only in selectively specific sensors with protection the measurements are credible, so this method will increase the redundancy. Secondly, the protection methods may not have been in a state of security if the attacker penetrate the protection and modify the measurements, and the state estimation may be at great risk.

This paper focuses on the detection method of unobservable false data injection attack in smart grid. The detailed solutions are listed as below:

Inspired by the literature [22–24], this paper analyzes the feasibility of detecting the false data injection attacks using machine learning algorithms, and applies the two-layer optimal-based extreme learning machine (ELM) into unobservable false data injection attack detection.

Power system contains multiple power plants, substations, different voltage levels and a large number of end users. There are a lot of redundant measurements in power system. Using deep-learning algorithm-Autoencoder to reduce the dimensionality of the data (m-dimension to n-dimension, $n \leq m$), which leads to lower computation complexities and makes the measurements to be separable [25]. To verify the effectiveness of the two-layer optimal-based algorithm for attack detection, we simulate a lot of experiments for the IEEE-118 bus and IEEE-14 bus system.

The rest of this paper is organized as follows. Section 2 formulates and shows the attack mechanism. Section 3 transforms false data injection attack into classification problem, and analyzes the dimension reduction method. The two-layer optimal-based attack detection algorithm is proposed in Section 4. Section 5 gives the numerical results. Conclusion and future work are discussed in Section 6.

2. Attack model

2.1. Derivation of residual in DC state estimation

In state estimation, the measurements include the actual power of the injection node and the actual power of the transmission line. $z \in \mathbb{R}^{m\times 1}$ is the measurement includes m measure values. $x \in \mathbb{R}^{n\times 1}$is the state vector includes n state values (generally $m > n$). In DC state estimation, the phase angle of slack node can be set to 0, all voltage amplitudes are assumed to be 1, parallel components and branch impedance can be neglected. The phase angles of other nodes constitute the state vector x. The linear measurement function for DC state estimation is shown as:

$$z = Hx + e \tag{1}$$

H is the $m \times n$ Jacobian matrix. e is the measurement noise, and it is subject to Gaussian distribution. We convert the nonlinear relationship between measured data to a local linear relationship, then the measurement residual is described as:

$$r = z - h(\widehat{x}) \tag{2}$$

To get the state $\widehat{x}$ with minimum weighted sum of squared residuals. The function can be established as:

$$\min f(x) = [z - h(x)]^T R^{-1}[z - h(x)] \tag{3}$$

where R^{-1} is the weight matrix of m-dimension measurement value. In order to approximate the current operating status of power system, the $\widehat{x}$ can be solved by weighted least squares state estimation algorithm equation:

$$\widehat{x} = (H^T R^{-1} H)^{-1} H^T R^{-1} z \tag{4}$$

We linearize $\widehat{z}$ around the true state values x as follows:

$$\widehat{z} \approx h(x) + H(x - \widehat{x}) = h(x) + H\widetilde{x} \tag{5}$$

The measurement error covariance matrix can be expressed as:

$$E\widetilde{z}\widetilde{z}^T = H(E\widetilde{x}\widetilde{x}^T)H^T = H(H^T R^{-1} H)^{-1} H^T \tag{6}$$

where $\widetilde{z}$ is the error vector of the n-dimensional estimated state, and $\widetilde{x}$ is n-dimensional state. Residual equation can be expressed as:

$$r \triangleq z - \widehat{z} \tag{7}$$

2.2. Unobservable false data injection attack

Traditional BDD (bad data detection) method uses the residual-based method to detect false information, and the measurement residual can be computed as:

$$\begin{aligned} \|r\| &= \|z - \widehat{z}\| = \|h(x) + n - h(x) - H\widetilde{x}\| \\ &= \|(I - H(H^T R^{-1} H)^{-1} H^T R^{-1})n\| \leq \tau \end{aligned} \tag{8}$$

where I is unitary matrix, if measurement residual is less than the given threshold τ, the measurements can be thought of not suffering attack. Assuming that the attacker knows about the matrix H, z_i represents measurement value after being attacked, and $a = (a_1, \ldots, a_m)$ represents the injected attack vector, where the zero values of it represent no false data injected into measurements. The false measurement is $z_i = z + a$, and residual value can be computed from formula Eqs. (1), (6), and (8):

$$\begin{aligned} \|z_i - \widehat{z}\| &= \|z + a - H(H^T R^{-1} H)^{-1} H^T R^{-1}(z + a)\| \\ &\leq \|z - Hx\| + \quad \|a - Hc\| \leq \tau \end{aligned} \tag{9}$$

where c is interference factor when the system being attacked, it can be an arbitrary value. When $a = Hc$ that the attack vector is the linear combinations of selected Jacoby column vectors of the matrix H, an attacker could invade smart grid and tamper the system measurements to launch unobservable attacks and change the power system operating state without being detected[26,27].

3. Feasibility of machine learning for attack detection

This chapter proposes using machine learning method to detect the false data injection attack. According to the difference between the data with attack and the normal data, the data can be classified into a two different specific spaces. With the dimension of measurement vector become larger, which may lead to the occurrence of curse of dimensionality. Deep learning algorithm-Autoenchoder is utilized to reduce the data dimension, so the subsequently detection will become easier using the low-dimension data.

3.1. Classification using machine learning

The false data injection attack detection can be converted to a binary classification problem. Define the data sample set $S = \{s_i\}_{i=1}^n$, category tag value $Y = \{y_i\}_{i=1}^n$, $y_i \in \{1, -1\}$, and the training data set $TS = (s_i, y_i) \in S \times Y$ follows independent identical distribution of the simultaneous distribution P. It is assumed that the classification mark l_i, of a new data s_i', is predicted using the predictably computable function $l_i = f(s_i')$. Therefore, the problem

of unobservable false data injection attacks detection can be defined as:

$$l_i = \begin{cases} 1 & \text{if } i \neq 0 \\ -1 & \text{if } i = 0 \end{cases}$$

where i is the attack vector. If $l_i = 1$, the i-th measurement vector is attacked, otherwise, it is not attacked.

On the choice of classification algorithms, [24] proposed using SVM (Support Vector Machine), and k-Nearest Neighbor method for detection. Although SVM shows better generalization ability, its linear programming problems will increase the training time. KNN can build the complex decision space model of ultra polygon, but it spends large amount of computation and needs mass storage support. The goal of detecting the unobservable false data injection attacks is to get a higher detection accuracy, this paper improves the detection accuracy on the basis of the optimal ELM.

3.2. Dimensional reduction using Autoencoder

Autoencoder provides a nonlinear mapping method for the input and output spaces. The measurement dimension will be reduced when the number of hidden units is smaller than the input dimension. It contains two operations: forward compression operation $C_f(x)$ transforms the data space to the encode space, and reverses expansion operation $S_f(x)$ transforms the encode space to the data space. The purpose is to train the identity map meets $C_f(S_f(x)) = x$. Using RBM (restricted Boltzmann machine, see Fig. 1) in the process of training. The appropriate network parameters (weights and bais) can be obtained through gradient descent method [28].

Algorithm 1: Autoencoder

Pre-train weights using energy function

$$E(v, h; \theta) = -\sum_i b_i v_i - \sum_j b_j v_j - \sum_{i,j} v_i h_j w_{ij}$$

Update weights

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij} = w_{ij}(t) + \varepsilon(\langle v_i h_j \rangle^+ - \langle v_i h_j \rangle^-)$$

Unrolling process

The original RBM will be cut into multi-layer neural network when its pre-training is completed, as shown in Fig. 1, this paper doesn't elaborate this process here.

Fine tuning using cross entropy function

$$H_m = -\sum_{i=1}^{m} [t_i \log y_i + (1 - t_i) \log(1 - y_i)]$$

Adjust output weight

$$\Delta w_{ij} = -\alpha \frac{\partial H_m}{\partial w_{ij}} = \alpha(t_i - y_i) O_j$$

End

As shown above, the optimal weights will be output. Where $\theta = \{w_{ij}, b_i, b_j\}$ is the parameter vector; v_i is the state of the i-th visible unit; h_i is the state of the j-th hidden unit. Δw_{ij} is weight adjustment; w_{ij} is the weights between the i-th and j-th neurons; $w(t)$ is the weight of t-th step; ε is learning rate; $\langle v_i h_j \rangle^+$ and $\langle v_i h_j \rangle^-$ are the positive and negative average correlation. "Sigmoid" function is the activation function of the output layer and its domain ranges from 0 to 1, so we adopt the scale transformation method that the original sample measurements matrix Z_r can be normalized to the measurements matrix Zn, the formula is shown as follows:

$$Z_n = a + b \frac{Z_r - \min(Z_r)}{\max(Z_r) - \min(Z_r)} \tag{10}$$

where a and b are constant, and $\max(Z_r)$ and $\min(Z_r)$ are the maximum and minimum of the vector in each group, respectively. Autoencoder uses low dimension produced by neural networks to represent the high dimension input. Compared with the linear dimension reduction method PCA (Principal Component Analysis) whose linearity limits the extracted feature dimensions. Autoencoder overcomes the limitation with the inherent nonlinear neural network. Its principle corresponds to PCA when it has one hidden layer, so Autoencoder not only can maximize the interval between normal data and abnormal data, but also can choose the most important data to compress which will minimize the reconstruction error. Autoencoder's encoding and decoding processes are shown in Fig. 2.

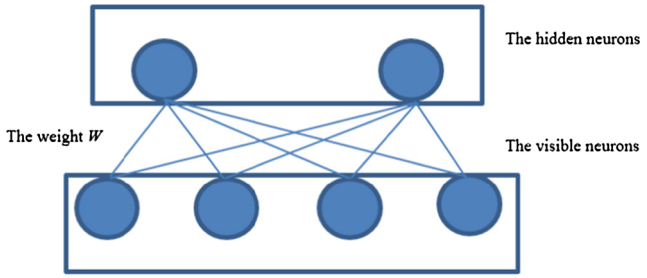

Fig. 1. The structure of RBM.

3.3. Brief of classification with ELM

This section briefly presents the Extreme Learning Machine (ELM) algorithm in data classification problem. ELM randomly generates input weights and hidden-layer bias, then the output weight can be obtained by analysis and calculation. In hidden layer, we choose "sigmoid" function as the activation function, and the output layer can choose linear activation function as the activation function [29–31].

Let $w_j \in \Re^d$ as the weights of the connection between i-th hidden layer nodes and the input nodes. $\beta_j \in \Re^m$ is the output weight vector that connects the i-th hidden layer node. b_j is the j-th hidden layer node bias. $\omega_j \cdot x_i$ is the inner product of ω_j and x_i. For n different training samples $\{x_i, y_i, i = 1, 2, \ldots, n\}$, $x_i \in \Re^d$, $y_i \in \Re^m$ correspond to m categories, category label is $y_s \in \{1, -1\}$ $(1 < s < m)$. If $y_k = 1$, and other elements in y_i are -1, this sample belongs to the j-th category. The model of ELM classifier contains L hidden layer nodes with activation function $h(\cdot)$ can be expressed as:

$$f_o(x_i) = \sum_{j=1}^{L} \beta_j h(w_j \cdot x_i + b_j), \quad i = 1, \ldots, n \tag{11}$$

$h(w_j \cdot x_i + b_j) = e^{\frac{-\|x_i - w_j\|}{2b_j^2}}$ is nonlinear radial basis activation function. The matrix form of Eq. (11) can be represented as:

$$M\beta = Y \tag{12}$$

where $Y = \{y_1; y_2; \ldots; y_n\} \in \Re^{n \times m}$, $\beta = \{\beta_1; \beta_2; \ldots; \beta_n\} \in \Re^{L \times m}$. M is the hidden layer output matrix, which can be expressed as:

$$M = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_n) \end{bmatrix} = \begin{bmatrix} h(w_1 \cdot x_1 + b_1) & \cdots & h(w_L \cdot x_1 + b_L) \\ \vdots & \ddots & \vdots \\ h(w_1 \cdot x_n + b_1) & \cdots & h(w_L \cdot x_n + b_L) \end{bmatrix} \tag{13}$$

where $h(x_i)$ is the output of hidden nodes, mapping D - dimensional input data x_i to L - dimensional feature space. The number of hidden layer neurons is far less than the number of training samples $(L \ll n)$. The output weights W_0 can be expressed as:

$$W_O = M^{\dagger} Y \tag{14}$$

where $M^{\dagger}$ is the generalized inverse matrix of M. The output function of the ELM classifier can be expressed as:

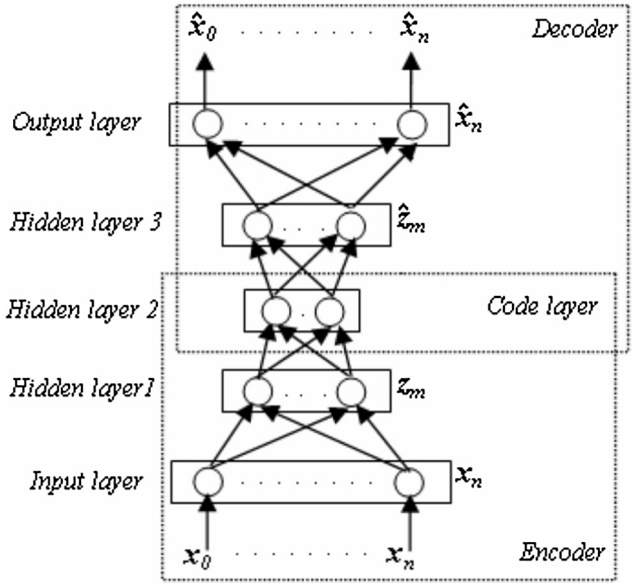

Fig. 2. Encoding and decoding processes.

$$f_o(x_i) = h(x_i)\beta = h(x_i)M^T\left(\frac{1}{\varepsilon} + MM^T\right)^{-1} Y \tag{15}$$

The predicted class label value for test sample x_j is $t_i = \arg\max_{s=1,2,\ldots,C} f_s(x_j)$. Where $f(x_j) = [f_1(x_j), \cdots, f_C(x_j)]^T$.

Appending non-negative value $1/\varepsilon$ to each element of the diagonal matrix MM^T improves the stability and generalization ability of the ELM classifier.

4. The proposed algorithm

From the respective of algorithm optimization, this paper applies the improved ELM into attack detection. The artificial bee colony (ABC) incorporates the thought of differential evolution algorithm (DE) to optimize the ELM's parameters and improve the detection precision.

4.1. Artificial bee colony optimization algorithm

Artificial bee colony algorithm is a bionics algorithm proposed by Karaboga [32–34]. In the ABC algorithm, the location of nectar represents the possible solution of optimization problems. Nectar amount stands for the quality of a feasible solution, which is called as fitness. Half of the swarm consists of employed bees, half by the onlookers.

The initial nectar (feasible solution) is a D-dimensional vector. D is equal to the number of optimization parameters. Set the population size as SN, the max times of nectar exploited as $\lim it$ and the max evolution number of the algorithm termination is MEN. According to the fitness value, employing bees find nectar to replace the original nectar with certain probability, which means modifying the original feasible solutions to generate new solutions. New nectar is generated by the lookers before forging. When the new nectar is better than the original one, the new one is exploited. The probability and the equation of generating new nectar are defined as Eqs. (16) and (17)

$$P_i = \frac{fitness_i}{\sum_{j=1}^{SN} fitness_j} \tag{16}$$

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{17}$$

where $fitness_i = \frac{1}{SN}\sum_{j=1}^{SN}\left|\frac{g_j - t_j}{g_j}\right|$ is the adaptation degree of i-the nectar, the bigger it is the more probability it will be selected. $k \in \{1,2,\ldots,SN\}$, $k \neq i$, and $j \in \{1,2,\cdots,D\}$ are all randomly selected values. ϕ_{ij} is a random number ranges from -1 to 1. x_{ij} represents the j -th parameter of the feasible solution of the i -th nectar location. v_i is candidate nectar location and x_k is adjacent position of the original nectar.

When the times of nectar exploiting reach foraging cycles predetermined limit, the employed bees become onlooker bees that randomly selects new honey source near the original source x_i. The equation of new nectar generation is defined as Eq. (18)

$$x_i = x^j_{\min} + rand(0,1)(x^j_{\max} - x^j_{\min}) \tag{18}$$

The max times of nectar exploited $\lim it$ will affect ABC's search capabilities, larger $\lim it$ will increase its exploiting capacity but will also weak its detectability.

4.2. Hybrid ABC-DE algorithm

From the Eq. (17), the search strategy of employed bees and onlooker bees only updates one element of nectar each time. This paper introduces mutation and crossover operation into employed bees phase. And add the accelerating evolution operation to increase the local search ability of the onlooker bees, and the optimal nectar source will be improved at last.

For the employed bee, the crossover operator can find all optimal locations. Operations on all elements instead of the operations on one element, in order to increase the diversity of each variation individual $x_i(t)$, the variation individuals can be generated as follows:

$$\begin{cases} v_i(t) = (v_i^1(t), \ldots, v_i^D(t)) \\ v_i^j(t) = x_{r1}^j(t) + F \times (x_{r2}^j(t) - x_{r3}^j(t)) \end{cases} \tag{19}$$

where $i = 1,2,\cdots,SN$, $j = 1,2,\cdots,D$, and $r1, r2, r3 \in \{1,2,\cdots SN\}$ ($i \neq r1 \neq r2 \neq r3$). $v_i^j(t)$ is the j -th component value of the i -th variation individual in the t-th generation population. $F \in (0,2)$ is the scaling factor of evolution parameter. We use the crossover operator to crossover individuals, then the experimental individual $u_i(t)$ can be achieved by $x_i(t)$ and $v_i(t)$.

$$u_i^j(t) = \begin{cases} v_i^j(t), & \text{rand}(0,1) \le CR \text{ or } j = j\_rand \\ x_i^j(t), & else \end{cases} \tag{20}$$

where $CR \in (0,1)$ is the crossover factor of evolution parameter. Comparing the fitness of the testing individual with that of the original individual, we can choose the individual with the better fitness as an individual of the new generation.

$$x_i(t+1) = \begin{cases} x_i(t), & fit(x_i(t)) < fit(u_i(t)) \\ u_i(t), & else \end{cases} \tag{21}$$

In ABC-DE, the onlooker bees assess the quality of nectar through the feedback information by employed bees. The idea of ABC may provide more opportunities of local search for the individuals with better evolution. The number of individual accelerating evolution is computed as follows:

$$T_i = SN * P_i \tag{22}$$

In order to get the optimal individual, the number of individual evolution and the upper limit of evolution times can be defined as $\lim it = SN * D$. According to Eq. (19), the randomly qualified individual will be regenerated by onlooker bees to replace the abandoned individuals, and the number of individual accelerating evolution is set to 0. For clarity, Fig. 4 presents the flowchart of the ABC-DE algorithm.

The schematic diagram for the classification and ELM parameter optimization is shown in Fig. 3. First, the Autoencoder reduces n

independent measurements from the available data set. Then we will find optimal parameters (number of hidden nodes and input weights) such that the validation performance of the ELM binary classifier improved.

Algorithm 2: ABC-DE-ELM

Input:- Training dataset D
Output:- Classification result
Step 1) Initialization:
Step 1.1) Set the index of generation $Gen = 0$, *generate random target vector* $\overrightarrow{v_{i,gen}}$, $i = 1, \ldots SN$.
Step 1.2) Randomly initialize a population of SN individuals x_i. *Initialize the parameter* $F, CR, \lim it$ *of DE.*
Step 1.3) For each vector $\overrightarrow{v_{i,gen}}$, *run ELM classifier on the training dataset and compute the corresponding objective function* $g(\overrightarrow{v_{i,gen}})$. *Set the number of function evalutions* $FES = SN$.
Step 2) ABC employed by DE
Step 2.1) Evaluate the fitness for each individual in Pi .
Step 2.2) for i=1 to SN, do Mutation; Crossover; Selection
Step 3) Set $FES = SN + FES$
Step 4) Go to step 6) if FES geq $\lim it$, otherwise set $Gen = Gen + 1$ and return to step 2.
Step 5) Select the optimal vector $\overrightarrow{v_i^*}$ corresponding to the minimum objective function $g(\overrightarrow{v_i^*})$.
Step 6) Train the ABC-DE-ELM using the optimal parameter vector $\overrightarrow{v_i^*}$,and compute the decision function for test sample according to the ELM classification theory.
End

5. Experimental results

This part selects the IEEE 118-bus and IEEE 14-bus systems to evaluate the performance of the proposed method. Using MATPOWER to simulate the operation of the power network [35–37], and collecting active measurement data from the testing systems, there are 304 and 54 kinds of measurements in the two IEEE bus systems, respectively, that is to say, there are 304 and 54 features. Assume that all load on the bus satisfy the uniform distribution $[0.8L_0, 1.2L_0]$, where L_0 is the base load of power system.

5.1. False data injection attack generating

According to the attack model, when the attack vector satisfied $a = Hc$, false measurement values and normal measurement values satisfy: $Z_a = Z + a = H(\theta + c) + e$, the attacker can successfully bypass the traditional threshold detection. This paper focuses on IEEE-14 bus and IEEE-118 bus systems, and we define the tampered state value as $\theta_a = \theta + c$ and set c = 1.0. Fig. 5 and Fig. 6 show the histogram of false measurements with or without attack.

Fig. 5 and Fig. 6 show measurements of different power system are very close 0. When the false data injected into power system, the histograms of false measurements are different from that of normal measurements. It can be found that the attack will affect the distribution of the measurements.

5.2. The performance of Autoencoder in dimension reduction

To verify the training effect of initial weights in training process and final reduction results in fine-tuning process, we select 600 samples from the 900 normal measurements of IEEE 118-bus as the training samples, and the rest as test samples. Select Mean Squared Error (MSE) as the evaluation standard during measurement pre-training process and fine-tuning process, and MSE can be defined as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Z_{data} - Z_{recon})^2 \tag{23}$$

where N is the size of training sample or testing sample. Z_{data} is training sample data or testing sample data. Z_{recon} is the different reconstructed data during pre-training process of RBM, or the reconstructed data during fine-tuning process. The pre-training process is composed of four layers of RBM, extracting "codes" from data sets according the structure "304–500–350–150–15". The topmost hidden layer of RBM has real-valued status and it is decided by the Gaussian function of the unit, and its mean value is determined by the input of visible RBM logic units. Generally, the probability is $\sigma(b_j + \sum_j v_i w_{ij})$ when the state of neuron j on the hidden-layer set to 1, and $\sigma(x)$ represents Sigmoid logical function, b_j is deviation, v_i and w_{ij} represents the status value and the weight value of hidden-layer, respectively. After determining the state of the hidden layer, producing reconstructed data through setting the state of the visible layers to 1 according probability $\sigma(b_i + \sum_j h_j w_{ij})$. The state of the hidden layer will be updated several times to mediate the weights for obtaining better initial weight. This experiment eventually reduces the data dimension to 15, and achieves reconstruction error of different layers RBM in the pre-training process after 10 epochs, when the number of iterations in pre-training process is set to greater than or equal to 2, the reconstruction error become stable, and we can get the initial weights used for fine-tuning, as shown in Fig. 7. To visualize the difference of measurement data after dimension reduction, this paper analyzes the 2-dimensional data, as shown in Fig. 8, the invasive data has been identified. The red circle represents illegal measurements, others as legal measurements.

We use Autoencoder and PCA to reduce the experimental data to the same dimension, compare the squared reconstruction error of both the training MSE and the test MSE to evaluate performance of Autoencoder. We achieve the average reconstruction error through fine-tuning 10 times for the training process and testing process. As shown in Table 1, when the dimension drops to 15, minimum reconstruction error of training and test data is 0.331 and 0.339, respectively.

5.3. ABC-DE-ELM based false data injection detection

We select "sig" function as ELM's activation function finally. The ABC searches for the best input weight and bias values such that the analytically calculated output weight in the ELM classifier achieves better performance. Assume not changing the normal variables, we improve the accuracy of attack detection as far as possible. Using 5 standard measures, such as precision, False Positive Rate (FPR), Recall, F value and True Positive Rate (TPR) to evaluate the performance of the proposed method, the 5 performance measures are defined as follows:

$$precision = \frac{|TP|}{|TP| + |FP|}, \quad Recall = \frac{|TP|}{|FN| + |TP|}, \quad FPR = \frac{|FP|}{|FP| + |TN|}$$

$$TPR = \frac{|TP|}{|TP| + |TN|}, F = \frac{|\text{Precision}| \times |Recall| \times 2}{|\text{Precision}| + |Recall|},$$

where TP is the number of legitimate measurement which is correctly classified as legitimate, TN is the number of illegal measure-

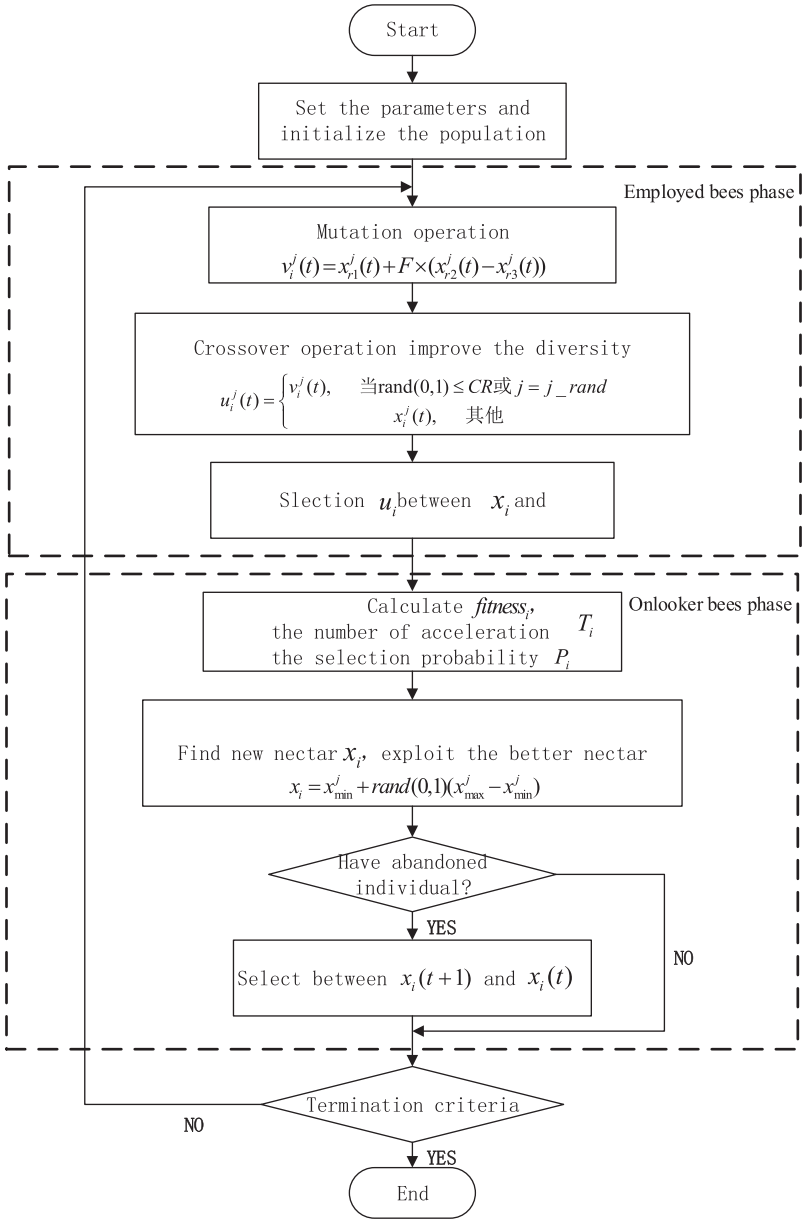

Fig. 3. Schematic diagram of attack detection using ABC-DE-ELM.

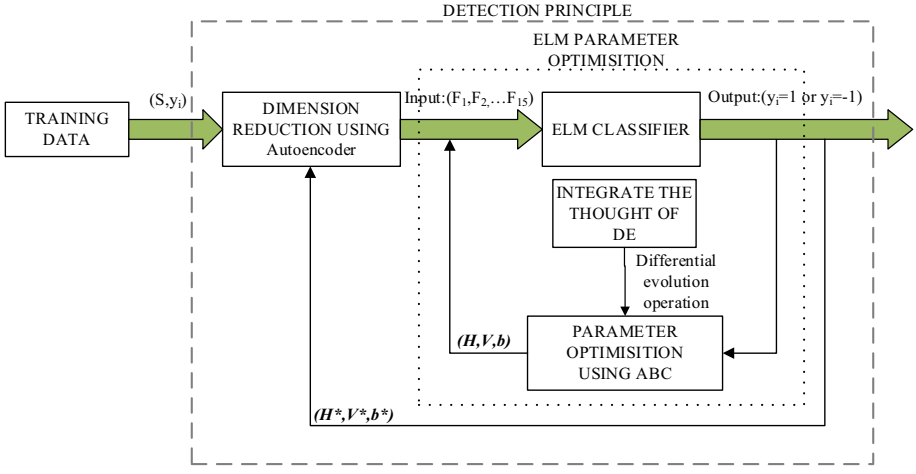

Fig. 4. The flowchart of the ABC-DE algorithm.

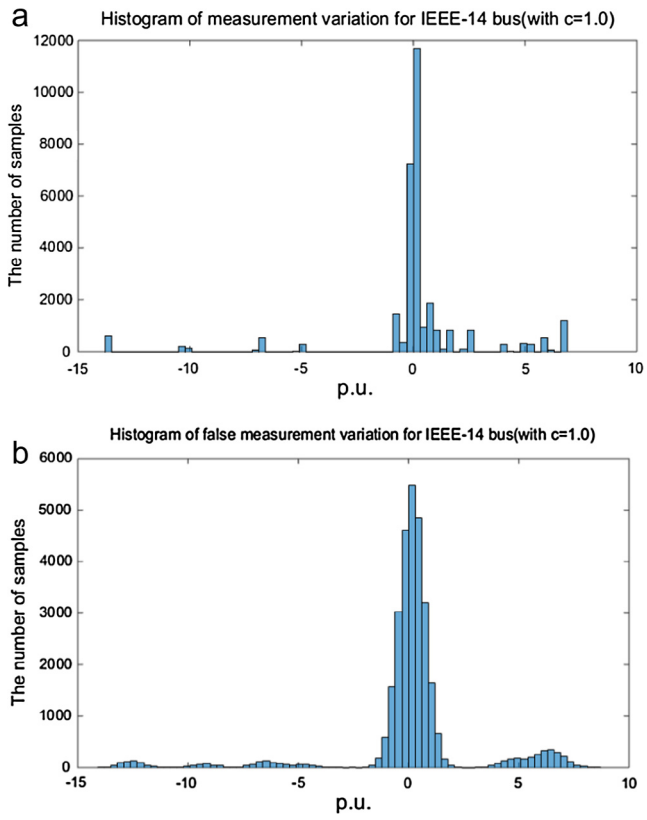

Fig. 5. (a) IEEE-14 bus system without false data. (b) IEEE-14 bus system with false data.

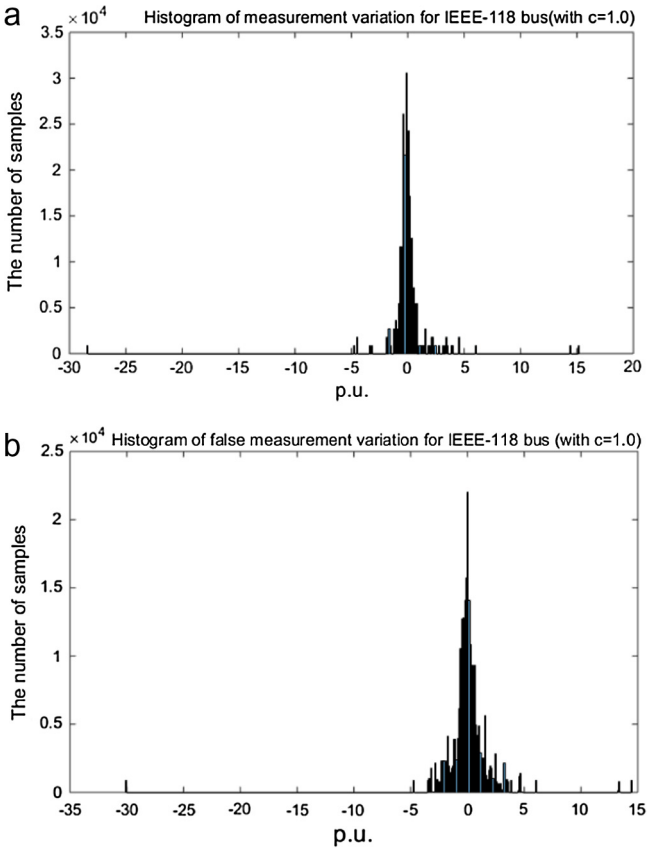

Fig. 6. (a) IEEE-118 bus system without false data. (b) IEEE-118 bus system with false data.

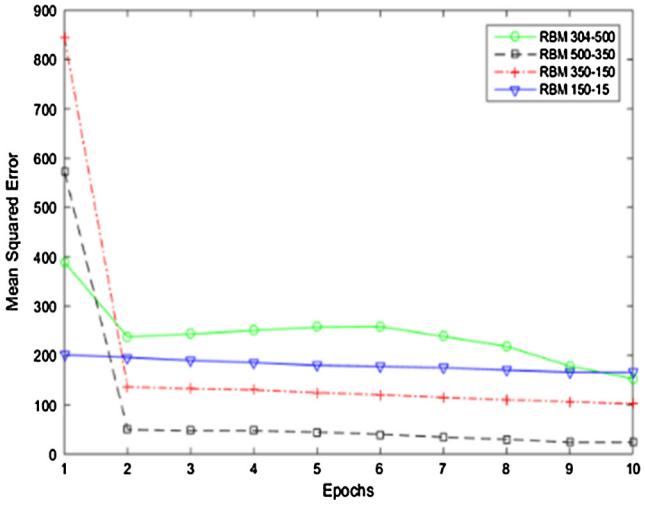

Fig. 7. Reconstruction Error for RBM pre-training.

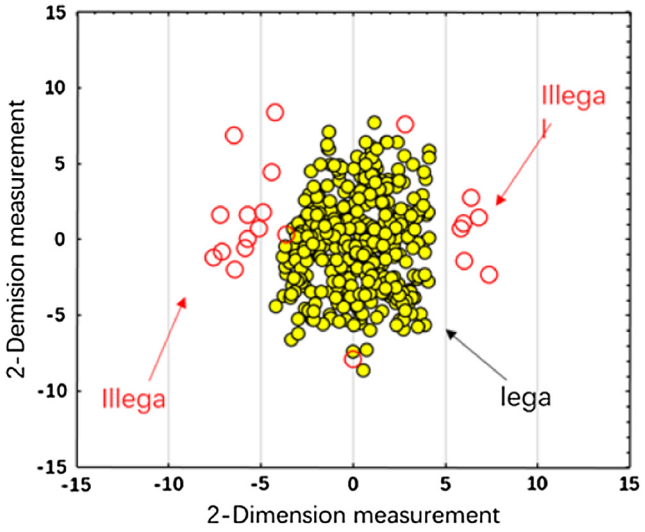

Fig. 8. Dimension reduction into R^2 space.

ment which is correctly classified as illegal, FN is the number of legal measurements, which is misclassified as illegal, FN is the number of illegal measurement which is misclassified as legal, and F value is used to comprehensively reflect the whole performance of algorithm.

To verify the superiority of the proposed method, and the practicality of the selected dimension reduction method. The final dimension is reduced to 5 by Autoencoder. We analyze the receiver operating characteristics curve (ROC) of different methods, then compare with the detection method with different dimension reduction methods. We select 1800 measurement samples as experimental data, and the training samples with the testing samples in a 2 to1 ratio.

Fig. 9 and 10 demonstrate that the proposed method can accurately identify illegal measurement data with lower error detection rate. In the IEEE118-bus system, when the False Positive Rate (FPR) is 10.53%, the True Positive Rate (TPR) of three methods are 94.8%, 88.8% and 76%, respectively. In the IEEE14-bus system, when the FPR is 11.6%, the TPR of three methods are 95.5%, 88.8 and 76.8%, respectively. The proposed algorithm is superior to ABC-ELM and ELM, and has a good generalization ability in different IEEE-bus measurements.

Table 1
MSE comparison among different dimensionality reduction methods.

Reduced dimension	Autoencoder		PCA	
	Training MSE	Test MSE	Training MSE	Test MSE
5	3.31e−01	3.39E−01	4.74e−01	9.17−01
10	3.31e−01	3.34E−01	9.42e−01	1.83E+00
15	3.31e−01	3.39E−01	2.73e+00	1.41E+02
20	3.32e−01	3.41E−01	1.87e+02	3.64E+02

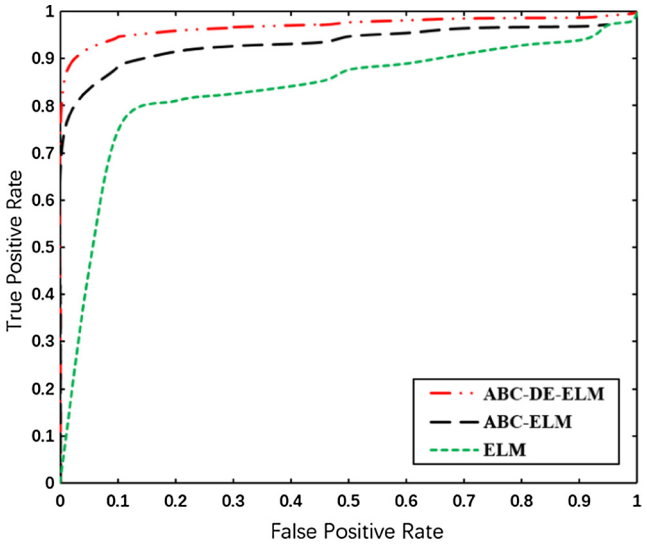

Fig. 9. The performance of the proposed algorithm for IEEE118-bus. ROC curves of EML, ABC-ELM, ABC-DE-ELM, respectively.

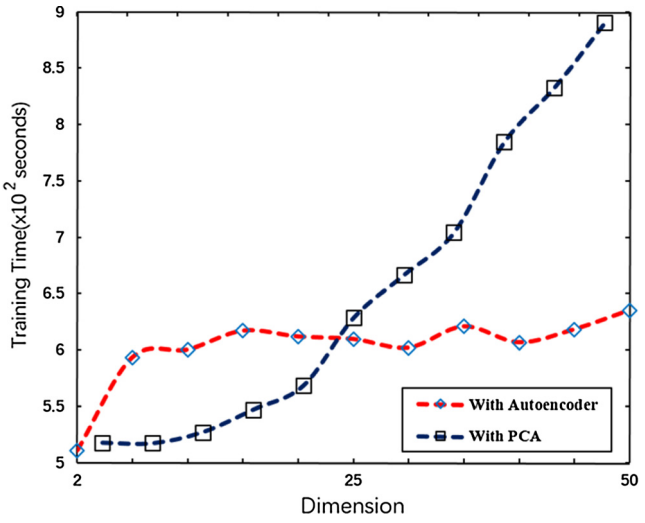

Fig. 11. Training time versus reduced measurement dimension for ABC-DE-ELM classifier with Autoencoder and PCA.

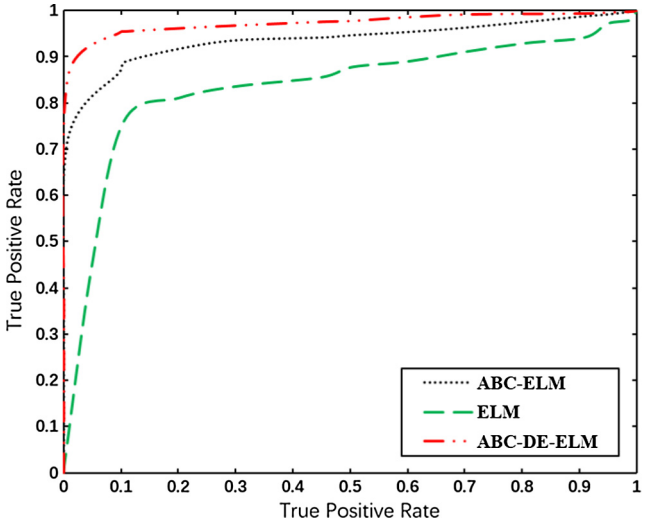

Fig. 10. The performance of the proposed algorithm for IEEE14-bus. ROC curve of ELM, ABC-ELM, and ABC-DE-ELM, respectively.

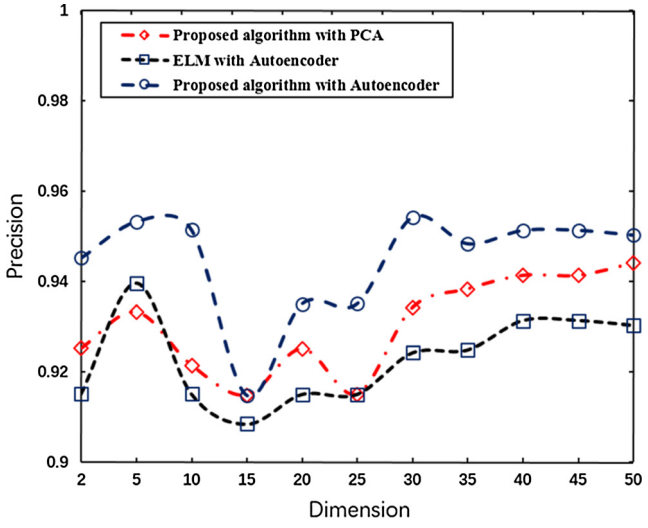

Fig. 12. Testing precision comparison of the proposed algorithm with Autoencoder and PCA.

As shown in Figs. 11and 12, the IEEE118-bus data is reduced to 2–50 dimension by Autoencoder and PCA. Use the reduced measurement to train ABC-DE-ELM classifier, when the dimension ranges from 2 to 25, the dimension method based on Autoencoder can save little time than that based on PCA, but with the dimension getting lower, the time difference is obvious. In terms of precision, the proposed method with Autoencoder is more superior than other methods. When the data dimension is reduced to 5, the detection accuracy is 95.3%, while the detection accuracy with PCA is 93.9%. At the end of our paper, we use Weka (Wikato Environment for Knowledge Analysis tool) to compare with the existing machine learning methods, the experimental results are shown in Table 2.

Table 2
Comparison of different classifiers.

Classifer	TP rate	FP rate	Precision	Recall	F-score
Native bayes	0.853	0.182	0.824	0.886	0.854
RBF network	0.925	0.076	0.924	0.925	0.924
Decision stump	0.881	0.105	0.894	0.905	0.899
SVM	0.934	0.091	0.9108	0.923	0.917
ELM	0.953	0.068	0.933	0.958	0.945
ABC-DE-ELM	0.964	0.047	0.953	0.961	0.957

6. Conclusion and future work

This paper firstly constructs attack vector, then simulates the normal and false measurements. We use the generated active power measurements as experimental data. The deep learning algorithm is used to reduce the dimensionality of data, which not only normalizes the experiment data, but also makes data show the good divisibility in low dimension space. In order to improve the performance of detection methods, we blend the thought of differential evolution into artificial bee colony algorithm. Using ABC-DE to optimize the weights and the parameters of ELM classifier, we verify the proposed detection method is superior to other machine learning methods. This paper opens a new era of using ELM with deep learning method to detect the false data injection attack, and the relevant work will be furtherly carried out in the future.

Acknowledgements

This work was supported in part by the National High Technology Research and Development Program of China under Grant (No. 2015AA016004), the National Natural Science Foundation of China (No. 61370126, No. 61672081 and No. U1636211), and the Fund of the State Key Laboratory of Software Development Environment (No. SKLSDE-2015ZX-16).

References

[1] Li H, Lai L, Zhang W. Communication requirement for reliable and secure state estimation and control in smart grid. IEEE Trans Smart Grid 2011;2(3):476–86.
[2] Shahidehpour M, Yamin H, Li Z. Market operations in electric power systems. New York: Wiley; 2002.
[3] Kim J, Tong L. On topology attack of a smart grid: Undetectable attacks and countermeasures. IEEE J Sel Areas Commun Jul. 2013;31(7):1294–304.
[4] Wang W, Xu Y, Khanna M. A survey on the communication architectures in smart grid. Comput Netw Oct. 2011;55(15):3604–29.
[5] Liu Y, Reiter MK, Ning P. False data injection attacks against state estimation in electric power grids. In: Proc 16th ACM conf comput commun security, Chicago, IL, USA; Nov. 2009. p. 21–32.
[6] Choi T-I, Lee KY, Lee DR, Ahn JK. Communication system for distribution automation using CDMA. IEEE Trans Power Del 2008;23(2):650–6.
[7] Yuan Y, Li Z, Ren K. Quantitative analysis of load redistribution attacks in power systems. IEEE Trans Parallel Distrib Syst Sep. 2012;23(9):1731–8.
[8] Wang W, Lu Z. Survey cyber security in the smart grid: survey and challenges. Comput Netw 2013;57(5):1344–71.
[9] Yuan X, Li Z. Local load redistribution attacks in power systems with incomplete network information. IEEE Trans Smart Grid 2014;5(4):230–41.
[10] Anwar A, Mahmood A. Cyber security of smart grid infrastructure, the state of the art in intrusion prevention and detection. USA: CRC Press; 2014. p. 514. http://dx.doi.org/10.1201/b16390-9.
[11] Anwar A, Mahmood A. Vulnerabilities of smart grid state estimation against false data injection attack. In: Renewable energy integration. Springer; 2014. p. 411–28.
[12] Manandhar K, Cao X, Hu F, Liu Y. Detection of faults and attacks including false data injection attack in smart grid using Kalman filter. IEEE Trans Smart Grid 2014;1(4):370–9.
[13] Yu Z, Chin W. Blind false data injection attack using PCA approximation method in smart grid. IEEE Trans Smart Grid 2015;6(3):1219–26.
[14] Li S, Yilmaz Y, Wang X. Quickest detection of false data injection attack in wide-area smart grids. IEEE Trans Smart Grid 2014;6(6):2725–35.
[15] Gu C, Jirutitijaroen P, Motani M. Detecting false data injection attacks in AC state estimation. IEEE Trans Smart Grid 2015;6(5):2476–83.
[16] Chen P, Yang S, McCann J, et al. Detection of false data injection attacks in smart-grid systems. IEEE Commun Mag 2015;53(2):206–13.
[17] Sedghi H, Jonckheere E. Statistical structure learning to ensure data integrity in smart grid. IEEE Trans Smart Grid 2015;6(4):1924–33.
[18] Deng R, Xiao G, Lu R. Defending against false data injection attacks on power system state estimation. IEEE Trans Ind Inf 2015;PP(99):1.
[19] Hao J, Piechocki R, Kaleshi D, et al. Sparse malicious false data injection attacks and defense mechanisms in smart grids. IEEE Trans Ind Inf 2015;11(5):1198–209.
[20] Yang Q, Yang J, Yu W, et al. On false data-injection attacks against power system state estimation: modeling and countermeasures. IEEE Trans Parall Distrib Syst 2013;25(3):717–29.
[21] Kim T, Poor H. Strategic protection against data injection attacks on power grids. IEEE Trans Smart Grid 2011;2(2):326–33.
[22] Mohammad E, Liu L, Nguyen N, et al. Detecting stealthy false data injection using machine learning in smart grid. IEEE Syst J 2014;PP(99):1–9.
[23] Ozay M, Esnao I, Yarman Vural F, et al. Machine learning methods for attack detection in the smart grid. IEEE Trans Neural Netw Learn Syst 2015;PP(99):1.
[24] Anwar A, Mahmood A, Shah Z. A data-driven approach to distinguish cyber-attacks from physical faults in a smart grid. In: Pro CIKM, New York, NY, USA; Nov. 2015.
[25] Hu C, Hou X, Lu Y. Improving the architecture of an autoencoder for dimension reduction. In: Pro. IEEE UIC, Sorrento Peninsula, Italy; May. 2014. p. 855–8.
[26] Anwar A, Mahmood A, Pickering M. Modeling and performance evaluation of stealthy false data injection attacks on smart grid in the presence of corrupted measurements. J Comput Syst Sci Feb. 2017;83(1):58–72.
[27] Anwar A, Mahmood A. Stealthy and blind false injection attacks on SCADA EMS in the presence of gross errors. In: IEEE int conf PESGM, Boston, MA; 2016. p. 1–5.
[28] Savich A, Moussa M. Resource efficient arithmetic effects on RBM neural network solution quality using MNIST, In: Pro IEEE ReConFig; Nov. 2011. p. 35–40.
[29] Li W, Chen C, Su H, et al. Local binary patterns and extreme learning machine for hyperspectral imagery classification. IEEE Trans Geosci Rem Sens 2015;53(7):3681–93.
[30] Cheng C, Tay W, Huang G. Extreme learning machines for intrusion detection. In: IEEE int conf IJCNN; Jun. 2012. p. 1–8.
[31] Yoan M, Sorjamaa A, Bas P, et al. OP-ELM, optimally pruned extreme learning machine. IEEE Trans Neural Networks 2009;21(1):158–62.
[32] Civicioglu P, Besdok E. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif Intell Rev 2013;39(4):315–46.
[33] Yang L, Sun X, Peng L, et al. An agent-based artificial bee colony (ABC) algorithm for hyperspectral image endmember extraction in parallel. IEEE J Sel Top Appl Earth Obs Rem Sens 2015;8(10):4657–64.
[34] Shrivastava A, Dubey M, Kumar Y. Design of interactive artificial bee colony based multiband power system stabilizers in multimachine power system. In: IEEE int conf CARE; 2013. p. 1–6.
[35] Zimmerman R, Murillo-Sanchez C, Thomas R. MATPOWER steady-state operations, planning and analysis tools for power systems research and education. IEEE Trans Power Syst 2011;26(1):12–9.
[36] Giani A, Bitar E, Garcia M, et al. Smart grid data integrity attacks. IEEE Trans Power Syst 2013;4(3):1244–53.
[37] Ozay M, Esnaola I, Yarman Vural FT, Kulkarni SR, Poor HV. Sparse attack construction and state estimation in the smart grid: centralized and distributed models. IEEE J Sel Areas Commun 2013;31:1306–18.