Research Article

Electricity Network Security Monitoring Based on Bee Colony Algorithm

Wenzhi Su and Baolong Zhang

Jiyuan Vocational and Technical College, Jiyuan, Henan 459000, China

Correspondence should be addressed to Wenzhi Su; 1710211105@hbut.edu.cn

Received 28 June 2022; Revised 5 August 2022; Accepted 13 August 2022; Published 7 September 2022

Academic Editor: B. Madhavan

Copyright © 2022 Wenzhi Su and Baolong Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to effectively detect and discover network threats in the initial stage, this study proposes an electricity network security intrusion detection method based on feature selection. A heuristic feature selection algorithm based on the bee colony algorithm is proposed to overcome the shortcomings of existing feature evaluation methods. The algorithm uses average mutual information to measure the importance of features and more truly reflects the relationship between the selected features, the selected features, and the classification labels. Aiming at the problem that the algorithm is easy to fall into local optimization, a heuristic random search algorithm is proposed, which iteratively optimizes to generate smaller feature subsets, and improves the speed and accuracy of intrusion detection. The experimental results show that compared with the traditional algorithm, the proposed method can effectively evaluate the risk of attack path on the selected experimental data set, and the gap between the generation strategy and the optimal strategy is reduced by 71.3%, which enhances the practicability of the attack graph analysis method in a large-scale network environment. Conclusion. This method has good scalability and can be applied to large-scale network environments. It can effectively obtain attack paths that are more in line with the real threat situation in an acceptable time, so as to effectively find the network threats.

1. Introduction

With the wide application of cloud computing, the security problem of cloud computing has become increasingly prominent. The cloud environment not only has to face all the security threats faced by the traditional network environment, but also has the characteristics of resource virtualization, high dynamic, and sharing, which makes the cloud environment have more attack surfaces than the traditional network, so it also needs to face many new threats [1]. Network systems are faced with a variety of security threats every day, such as IP and port scanning, camouflage attacks, DDoS attacks, illegal privilege escalation, and so on. These means are not only destructive to the system but also may be used as the leading means of subsequent high-level threats, seriously endangering the system security. Therefore, effective detection and discovery of threats in the initial stage are of great significance to ensure system security. Intrusion detection system can provide an early warning function for the system in the face of threats. Since the intrusion detection technology was proposed, it has become a common electricity network security guarantee method and an important part of the electricity network security guarantee system. For an intrusion detection system, the correct rate of identifying intrusion and the detection speed are two important indicators to evaluate its performance [2]. Anomaly-based intrusion can extract features from incoming and outgoing network packets and system logs waiting for analysis data, and judge whether the samples belong to normal or intrusion according to the features. However, the more features are extracted, the better. Too many features will seriously affect the performance of the intrusion detection system. In fact, many features do not contribute or contribute little to the identification of samples. Cloud environment electricity network security monitoring can monitor the status and behavior of physical
nodes, virtual nodes, software, users, etc. of the cloud, integrate security tools such as firewall, vulnerability detection, intrusion detection, deep packet detection, collect data for fusion analysis, and present the results in a visual form to help administrators or users effectively understand the security status of the cloud and ensure the electricity network security of the cloud environment (as shown in Figure 1).

On this basis, the research on the emerging bee colony algorithm and other technologies has enabled it to adapt to the characteristics of the network and built a basic electricity network security guarantee system, which is of great development significance to improve electricity network security [3].

2. Literature Review

For electricity network security monitoring, George and Ganesan proposed a multi granularity pub/sub mode cloud security monitoring architecture based on communication load prediction. On this basis, a multi granularity pub/sub mode cloud security monitoring architecture PSCSMA based on communication load prediction is proposed, which meets the scalability requirements of the monitoring architecture by dividing the monitoring domain, adopting the publish- or subscribe-based information interaction mode to meet the elastic requirements of the monitoring architecture, and meets the comprehensive requirements of the monitoring architecture through the multi-user characteristics of data collection agent and publish/subscribe mode [4]. Etinkaya, and others proposed a multi-granularity topic adaptive mechanism based on communication load prediction to balance the relationship between timeliness, accuracy, and adaptability of the monitoring architecture [5]. Tefek, and others proposed a method to generate an electricity network security reinforcement policy in cloud environment based on an attack graph. They put forward a risk assessment method that comprehensively considers the attack difficulty, attack consequence severity, path length, and the importance of target nodes. They search through heuristic algorithms to obtain the approximate maximum risk coefficient in the attack graph, avoiding the exponential time complexity problem of accurately searching the attack path [6]. Vajjha and Sushma proposed an electricity network security intrusion detection method in cloud environment based on feature selection. Aiming at the shortcomings of existing feature evaluation methods, a heuristic feature selection algorithm HFS-ACMI based on average conditional mutual information is proposed [7]. Wu and others proposed a regular expression grouping algorithm for efficient deep packet detection in cloud environment. After analyzing the state explosion problem of DFA merging regular expressions, the optimal grouping problem of regular expressions is reduced to the K-MAX cut problem of weighted undirected graphs [8]. Einy, and others proposed a heuristic random search algorithm for iterative optimization. Compared with other algorithms, the random search algorithm can generate a smaller feature subset, and the classifier constructed with this subset has faster detection speed and higher classification accuracy [9].

Therefore, taking into account the shortcomings of existing methods for evaluating traits, we propose an algorithm for selecting heuristic traits based on the bee colony algorithm. The algorithm uses average mutual information to measure the importance of features, and more truly reflects the relationship between the selected features, the selected features, and the classification labels. Aiming at the problem that the algorithm is easy to fall into local optimization, a heuristic random search algorithm is proposed, which iteratively optimizes to generate smaller feature subsets, and improves the speed and accuracy of intrusion detection.

3. Research Methods

3.1. Introduction to Artificial Bee Colony Algorithm. The ABC algorithm uses the location of the nectar source to represent the solution and the number of nectar source dusts to represent the fitness value of the solution [10, 11]. All bees are divided into three groups: working bees, follower bees, and research bees. Worked bees and companion bees make up half of the total bee colony. Mercenary bees are primarily responsible for finding and exchanging honey sources. The following bees are responsible for collecting honey in the hive according to the information provided by the hive bees. Exploration bees are responsible for randomly searching for new sources of honey. Replace the original honey source after leaving the honey source. Like other herd intelligence algorithms, the ABC algorithm is iterative [12]. After initiating a bee colony and nectar source, three processes are repeated to find the optimal solution to the problem: hiring bees, tracking bees, and studying bees. Each step is described as follows.

(a) Start the herd. To start the parameters of the ABC algorithm, these parameters are the number of nectar sources SN, the number that determines the limit of discarding the nectar source, and the number of iteration ends. In the standard ABC algorithm, the number of nectar sources SN is equal to the number of working bees and the number of following bees. The formula for generating a specific nectar source is as follows:

\[ x_{ij} = x_{\text{min}j} + \text{rand}(0, 1)(x_{\text{max}j} - x_{\text{min}j}), \]  

where \( x_{ij} \) represents the \( j \)-th dimension value of the \( i \)-th honey source \( x_i \), \( i \) is taken from \( \{1, 2, ..., sn\} \), and \( j \) is taken from \( \{1, 2, ..., d\} \); \( x_{\text{min}j} \) and \( x_{\text{max}j} \) represent the minimum and maximum values of dimension respectively. To initiate a nectar source, the above formula is used to assign a random value within all dimensional values of each nectar source and randomly generate the first nectar source of SN.

(b) Employment bee stage. In the stage of employing bees, employing bees use the following formula (2) to find new honey sources:

\[ v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}), \]
where \( x_k \) represents the neighborhood honey source, \( k \) is taken from \( \{1, 2, \ldots, s_n\} \), and \( k \) is not equal to \( i \). \( \phi_{ij} \) is a random number with a value of \([-1, 1]\).

After a new honey source is obtained through equation (2), the greedy algorithm is used to compare the fitness values of the old and new honey sources and select the best.

(c) Follow the bee stage. The bee hiring phase ends and the bee compliance phase begins. At this stage, hired bees share information about the source of nectar on the dance floor. Following the bees will be more likely to analyze the data, control and extract nectar sources using a roulette strategy, and extract nectar sources of high fitness value. Following the bee extraction process is similar to hiring honey, using formula (2) to find a new source of nectar and leave a better adaptive. Nectar has a parametric test. When backing up the nectar update, the line is incremented by 0, otherwise the string is incremented by 1. Thus, the court can count how many times the source of nectar has not been renewed.

(d) Study the bee stage. If a nectar source has not been updated after many times of mining, and the trail value is too high and exceeds the predetermined threshold limit, then the nectar source needs to be discarded and the exploration bee stage is started. This reflects the negative feedback and volatility properties of self-organization in ABC. At this stage, the bee is randomly looking for a new nectar source to replace the destroyed nectar source using the following equation:

\[
    x_{ij} = x_{min_j} + \text{rand}[0, 1] \left( x_{max_j} - x_{min_j} \right),
\]

(3)

(a) ABC operation is simple and local search capabilities are weak. GA and DE are hybridized to create a new solution, but ABC is not. ABC’s new solution is based solely on the original solution (the old solution) and is simple to operate, suitable for local search frequency modulation, but it does not allow good information to spread rapidly to a population, leading to each mutation. Because it only changes one dimension of the original solution, and because the range of change is small, ABC’s local optimization capabilities are weak and integration speeds are slow, especially in solving constraint policies, complex functions, and integral functions [13].

(b) ABC has good exploration capabilities. The researcher can jump out of the original solution package and randomly find a new solution that completely replaces the old solution. This function reduces the dependence of the algorithm on the size of the herd and the impact of the initial solution package, ensures herd diversity, and prevents premature mergers, making ABC suitable for high-dimensional and multidimensional problems.

(c) The ABC parameters are low. In addition to the maximum number of cycles and the size of the population, the ABC algorithm has only one control parameter limit. The limit value, in turn, depends on the size of the population and the size of the problem, such as the limit = \( SN \times D \). Finally, ABC has two control parameters: maximum cycle number (MCN) and bee colony number (SN). For the ABC algorithm, the effects of some key parameters on the algorithm’s performance are summarized as follows:

(a) The performance of ABC is better when the number of initial colonies is more. However, the performance of ABC algorithm will not be improved when the number of bees reaches a certain amount. When the number of the bee colony is 50–100, a better convergence rate can be obtained. ABC algorithm does not need a large number of bees in high-dimensional optimization problems, and is suitable for solving high-dimensional problems [14].
3.2. Improved Artificial Bee Colony Algorithm. The artificial bee colony algorithm is a swarm intelligence optimization algorithm proposed in, which is based on the self-organizing simulation model of bee swarm intelligence. It was successfully applied to the numerical optimization of functions at first. The algorithm has the characteristics of simplicity, strong robustness, and strong global search ability.

Artificial bee colony algorithm simulates the real honey collecting behavior of bees by dividing the colony into collecting bees, observing bees, and reconnaissance bees. The position of each honey source (the decision variable of the function to be solved) in the program algorithm represents a feasible solution of the optimization problem, and the quality of the honey source corresponds to the quality of the corresponding feasible solution. Each group of solutions \( x_i (i = 1, 2, \ldots, d) \) is a dimensional vector (the number of decision variables of the optimization parameters). They are evaluated at the same time when they are generated. Then, honeybees are collected to generate a new honey source location and evaluate it according to the local honey source location information. If it is better than the initially generated honey source, it will be replaced. Otherwise, the original solution will not be changed [15]. In the second step, according to the quality of the honey source, the observation bees are dispatched to select the solution with good quality of the honey source with a high probability through the set "survival of the fittest" mechanism. At the same time, a group of new honey sources is also generated according to the local information disturbance at the honey source. If the quality of the new honey source is better than the old one, it will be replaced, and vice versa. The above two steps, such as collecting bees and observing the continuous circulation of bees, gradually improve the location of the honey source. After several times of selection, if the location of some honey source is still not improved, the reconnaissance bee operation will be carried out, the honey source will be abandoned, a group of substitutes will be found, and the evaluation will be carried out as before. The whole process is executed repeatedly, and the optimal solution of each step is recorded until it stops [17].

Taking the minimum optimization problem as an example, the fitness of the nectar amount of the food source corresponding to the actual solution is the formula:

\[
\text{FIT}_i = \begin{cases} 
1, & \text{fit}_i \geq 0, \\
1 + \text{abs}(\text{fit}_i), & \text{fit}_i < 0.
\end{cases}
\]

Among them, the higher the quality of a group of honey sources, the greater the corresponding fitness. The probability of observing a bee selecting a group of honey sources is given by the following formula:

\[
P_i = \frac{\text{FIT}_i}{\sum_{i=1}^{n} \text{FIT}_i}.
\]

In the algorithm, roulette is used to realize the "survival of the fittest," so that the higher the fitness of the honey source, the greater the probability of being selected.

After bee collecting judges the advantages and disadvantages of the honey source, a new solution for the neighborhood is generated according to the following formula, and the evaluation is continued, which is given in the following formula:

\[
x_{ij} = x_{ij} + \varnothing(x_{ij} - x_{kj}),
\]

where \( \varnothing \) is a random number \((-1, 1), j = (1, 2, \ldots, d), \) and \( k(1, 2, \ldots, n) \) is randomly selected here, but it must be ensured that \( k \neq i \). In this way, the neighborhood of the original honey source is controlled, which can make the honey sources learn from each other and shorten the distance from the optimal solution. With the gradual approximation of the optimal solution, the range of the neighborhood will be gradually reduced, and the step size will be adaptively reduced. If a honey source has not been improved after a limited number of times, give up the position and perform a reconnaissance bee operation, which is given in the following formula:

\[
x_{ij} = x_{ij} + \text{rand}[0, 1](x_{j \max} - x_{j \min}).
\]

3.3. Modeling Steps of Stepwise Regression Based on Artificial Bee Colony Algorithm

Step 1: establish the dam deformation regression model according to the stepwise regression analysis model, and retain the regression load set variable factor;

Step 2: make the load set coefficient the decision variable of the objective function of the artificial bee colony, and make full use of the stepwise regression results to make the policy variable \( b \) randomly assigned in their respective definition domain \( (x_{j \max} - x_{j \min}) \), so as to reduce the blindness of the initial population and improve the optimization speed and efficiency;

Step 3: according to the actual solution problem, the residual sum of squares is taken as the objective function, which is transformed into the optimization problem of finding the function’s extreme value. If
there are \( s \) groups of observation samples, the actual displacement is \( y_i \), and the prediction result obtained by the model is \( \tilde{y}_i \); Then the objective function is formula:

\[
\min (g) = \sum_{i=1}^{s} (y_i - \tilde{y}_i)^2.
\] (8)

Since the large-scale deformation model is a minimization problem and non-negative, it is transformed into individual fitness. The smaller the target value, the higher the fitness and the greater the probability of being selected.

After the new solution is found in Step 4, the relevant statistical indicators of the model are calculated and compared with the original model. This process is repeated until the decision variables obtained from the optimization results reach satisfactory accuracy, as shown in Figure 2.

3.4. Risk Coefficient Measurement Method of Base Attack Path. Security vulnerabilities are common in the network environment, and a large number of new vulnerabilities are discovered every day, but it is often unrealistic to repair all vulnerabilities. The attack graph can well show the correlation between vulnerabilities. Through the analysis of the attack graph, it is a more practical method to select vulnerabilities with high risk and low repair difficulty to repair, so as to make the network environment have a relatively high degree of security as far as possible [18].

In order to obtain the electricity network security reinforcement strategy, we first need to know what possible attack paths the attacker has. In the attack graph, the algorithm of searching all attack paths has unavoidable exponential time complexity, and can not be applied to large-scale network environments [19]. A more practical method is to evaluate the risk of attack paths, search for attack paths that are more likely to be attacked, and block them in advance to ensure the relative security of the network.

In order to measure the attacker’s willingness to attack on different attack paths, it is necessary to evaluate the severity of all vulnerabilities on the attack path. To solve this problem, CVSS will give a base score for each known vulnerability. The score is calculated by such indicators as an access method, attack complexity, authentication, confidentiality impact, integrity impact, and effectiveness impact. The higher the vulnerability risk, the higher the score. The value range is [0,10]. It can be seen that, as a general vulnerability scoring standard, the CVSS basic score comprehensively considers the attack difficulty and the severity of the consequences, which can better reflect the attacker’s willingness to attack different vulnerabilities [20]. Therefore, the CVSS basic score is used as a reference to explain the calculation method of the attack path risk coefficient. In the process of attackers’ multi-step attack on the network, each step of the attack corresponds to a security vulnerability. For any atomic attack \( a \), let \( r(a) \) be the risk factor of atomic attack \( a \), then the formula can be given as:

\[
r(a) = \frac{\text{CVSSBase score}(a)}{10}. \tag{9}
\]

For any attack path \( l \), \( r(l) \) is the risk factor of the attack path, which is defined as the product of the risk factors of all exploits on \( l \), then formula can be given as:

\[
r(l) = \prod_{a \in l} r(a). \tag{10}
\]

The risk coefficient of the attack path reflects the relative degree of the willingness of the attacker to choose the path, which provides a basis for evaluating the threat degree of the attack path.

4. Result Analysis

To verify and illustrate the effectiveness of the proposed method, a simple example network is built, as shown in Figure 3.

A network is a general office network that consists of three servers: a file server (FS), a database server (DB), and a web server (WS). The three servers are connected to the Internet via a loose firewall, making FS easily accessible to anyone with user privileges, and DB and WS are key nodes. For convenience, we call FS Host 0, DB Host 1, and WS Host 2. Network vulnerabilities are shown in Table 1.

Before using the proposed method, the attack graph should be constructed first. Since the research focus of this study is to put forward the network security reinforcement strategy through the analysis of the attack graph, the manual method is used here to build the attribute attack graph AG of the network. The key attributes in the attack graph AG are root (1) and root (2). First, add the subsequent atomic attack \( a_1, a_2, r(a_1) = r(a_2) = 1 \) for root (1) and root (2) respectively, and then add a common result attribute \( v_k \) for \( a_1, a_2 \), which turns the problem into finding the maximum risk coefficient attack path of attribute \( v_k \) in the AG. The number of iterations of the ant colony (as) algorithm T50, the number of ants \( k = 4 \), \( \omega = 2 \), and other parameter settings refer to the common settings of the ASRank model, as shown in Table 2. Where \( \omega^\beta \) is the solution constructed by greedy algorithm, and \( m \) is the number of steps of the solution.

After calculation, the maximum risk factor attack path \( l_{\text{max}} \) in AG is (sshd BOF (1) --- \( a_1 \)), and the risk factor \( r_{\text{max}} = 0.9 \). Experiments show that the proposed method can effectively evaluate the risk of attack path.

Based on the above discussion, 10 groups of data are used to compare the performance of the bee colony algorithm and the weighted-greedy (WG) algorithm. Each group of data includes the number of sets \( m \), the number of elements \( n \), the cost of each element \( W_i \), the elements contained in each set and the optimal solution of each group of problems [21, 22]. For 10 groups of data, WG algorithm and bee colony algorithm are used to solve them respectively. The parameters of the bee colony algorithm are set as follows: the number of iterations \( t \) is set to 100, \( a = 1, \beta = 2, \rho = 0.5, Q = 1, r_0 = \omega^\beta/m \), the result is shown in Figure 4.

Figure 4 shows the average time that the two algorithms run on each set of data. WG algorithm takes less than 1
second, which is obviously less than the bee colony algorithm. Although the running time of the bee colony algorithm is long, it is still within a reasonable range. Combined with the previous experimental results, the results calculated by the WG algorithm are not ideal. In practical application, the reinforcement strategy given by the WG algorithm will cause greater repair costs to users, and the result of the bee colony algorithm is closer to the optimal solution, which can effectively reduce the required security reinforcement cost and obtain a better reinforcement strategy and better security guarantee in the case of limited resources. For an actual network, the time required to generate the security
reinforcement strategy is only within a reasonable range. It is obviously more practical to be able to calculate a better solution and save the repair cost for users as much as possible. Therefore, for the network security reinforcement of a large-scale network environment, the bee colony algorithm has greater practical application value.

5. Conclusion

This article proposes an algorithm for selecting heuristic traits based on the bee colony algorithm. The algorithm uses average mutual information to measure the importance of features, and more truly reflects the relationship between the selected features, the selected features, and the classification labels. Focusing on the fact that the algorithm is easily accessible to local optimization, it offers a heuristic random search algorithm that seeks to generate subgroups of iterative functions over and over again, improving the speed and accuracy of attack detection. Combined with the approximate maximum risk coefficient, network scale, network nature and actual demand, the danger threshold is set, the search process is limited to the threshold of danger to reduce the complexity of the algorithm. Experimental results show that this method is highly scalable and can be used in large network environments. Compared with the existing methods, it can effectively obtain attack paths that are more in line with the real threat situation in an acceptable time. On the selected experimental data set, compared with the traditional algorithm, this method reduces the gap between the generation strategy and the optimal strategy by 71.3% and enhances the practicability of the attack graph analysis method in large-scale network environments.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by (1) Training Program for Young Backbone Teachers in Colleges and Universities of Henan Province, 2020YB0655, Research on quality monitoring and evaluation of Teachers Quality Improvement Program in Higher Vocational Colleges; (2) Training Plan for Young Backbone Teachers in Colleges and Universities of Henan Province, 2018GGJS259, Development of Lingkang Yun-cheng Platform Based on VUE; (3) Project of Jiyuan Vocational and Technical College, JYZY-2021-95, Construction and Research of College Network Space Security Experience Platform; and (4) 2022 Humanities and Social Science Research Project of Universities in Henan Province, 2022-ZDJH-00152, Research on Red Culture Network Communication.

References

[1] X. Liu, J. Liang, D. Y. Liu, R. Chen, and S. M. Yuan, "Weapon-target assignment in unreliable peer-to-peer architecture based on adapted artificial bee colony algorithm," *Frontiers of Computer Science*, vol. 16, no. 1, Article ID 161103, 2022.
[2] X. Zong, A. Liu, C. Wang, Z. Ye, and J. Du, "Indoor evacuation model based on visual-guidance artificial bee colony algorithm," *Building Simulation*, vol. 15, no. 4, pp. 645–658, 2022.
[3] P. Lakshminarayana and T. V. Sureshkumar, "Automatic generation and optimization of test case using hybrid cuckoo search and bee colony algorithm," *Journal of Intelligent Systems*, vol. 30, no. 1, pp. 59–72, 2020.
[4] T. George and V. Ganesan, "An effective technique for tuning the time delay system with pid controller-ant lion optimizer algorithm with ann technique," *International Journal of COMADEM*, vol. 23, no. 1, pp. 39–48, 2020.
[5] M. B. Etinkaya and H. Duran, "A detailed and comparative work for retinal vessel segmentation based on the most

| Table 1: Vulnerabilities in the network. |
| CVE   | Access | Protocol | CVss score | Host |
|-------|--------|----------|------------|------|
| CVE2012-2999 | Remote | Ftp | 6.8 | 1.2 |
| CVE2008-3234 | Remote | Ssh | 6.5 | 1 |
| CVE2004-1161 | Remote | Ssh | 7.5 | 2 |
| CVE2013-4512 | Local | Ssh | 4.7 | 1.2 |
| CVE2008-6899 | Remote | Ssh | 9.0 | 1 |

| Table 2: ASRank model settings. |
|------|------|-----|-----|-----|
| $a$  | $b$  | $p$ | $\tau_0$ | $Q$ |
| 1    | 2    | 0.5 | $w^d/m$ | 1   |

Figure 4: CPU running time of various algorithms under different scale attacks.
effective heuristic approaches,” *Biomedical Engineering/Biomedizinische Technik*, vol. 66, no. 2, pp. 181–200, 2021.

[6] M. F. Tefek, “Artificial bee colony algorithm based on a new local search approach for parameter estimation of photovoltaic systems,” *Journal of Computational Electronics*, vol. 20, no. 6, pp. 2530–2562, 2021.

[7] H. Vajjha and P. Sushma, “Techniques and limitations in securing the log files to enhance network security and monitoring,” *Solid State Technology*, vol. 64, no. 2, pp. 1–8, 2021.

[8] D. Wu, “A network security posture assessment model based on binary semantic analysis,” *Soft Computing*, vol. 6, pp. 1–8, 2022.

[9] S. Einy, C. Oz, and Y. D. Navaei, “The anomaly- and signature-based ids for network security using hybrid inference systems,” *Mathematical Problems in Engineering*, vol. 2021, no. 9, 10 pages, Article ID 6639714, 2021.

[10] X. Zhou, J. Lu, J. Huang, M. Zhong, and M. Wang, “Enhancing artificial bee colony algorithm with multi-elite guidance,” *Information Sciences*, vol. 543, pp. 242–258, 2021.

[11] K. Zhou, Y. Wen, W. Wu, Z. Ni, T. Jin, and X. Long, “Cloud service optimization method based on dynamic artificial ant-bee colony algorithm in agricultural equipment manufacturing,” *Mathematical Problems in Engineering*, vol. 2020, no. 1, 11 pages, Article ID 9134695, 2020.

[12] A. Sharma, G. Rathee, R. Kumar et al., “A secure, energy- and sla-efficient (seese) e-healthcare framework for quickest data transmission using cyber-physical system,” *Sensors*, vol. 19, no. 9, p. 2119, 2019.

[13] M. Bradha, N. Balakrishnan, S. Suvi et al., “Experimental, computational analysis of Butein and Lanceolatin for natural dye-sensitized solar cells and stabilizing efficiency by IoT,” *Environment, Development and Sustainability*, vol. 24, 2021.

[14] X. Liu and Z. Ahmadi, “H$_2$O and H$_2$S adsorption by assistance of a heterogeneous carbon boron nitrogen nanocage: computational study,” *Main Group Chemistry*, vol. 21, no. 1, pp. 185–193, 2021.

[15] Z. Huang and S. Li, “Reactivation of learned reward association reduces retroactive interference from new reward learning,” *Journal of Experimental Psychology Learning Memory and Cognition*, vol. 48, no. 2, pp. 213–225, 2022.

[16] R. Huang and X. Yang, “Analysis and research hotspots of ceramic materials in textile application,” *Journal of Ceramic Processing Research*, vol. 23, no. 3, pp. 312–319, 2022.

[17] D. Trachanatzi, M. Rijakis, M. Marinaki, Y. Marinakis, and N. Matsatsinis, “Distance related: a procedure for applying directly artificial bee colony algorithm in routing problems,” *Soft Computing*, vol. 24, no. 12, pp. 9071–9089, 2020.

[18] A. Fx, B. Cmp, D. Hlc, E. Yz, A. Ys, and G. Hao, “Training feed-forward artificial neural networks with a modified artificial bee colony algorithm—sciedirect,” *Neurocomputing*, vol. 416, pp. 69–84, 2020.

[19] S. Famila, A. Jawahar, A. Sariga, and K. Shankar, “Improved artificial bee colony optimization based clustering algorithm for smart sensor environments,” *Peer-to-Peer Networking and Applications*, vol. 13, no. 4, pp. 1071–1079, 2020.

[20] X. Wang, H. Yu, Y. Lin, Z. Zhang, and X. Gong, “Dynamic equivalent modeling for wind farms with dfigs using the artificial bee colony with k-means algorithm,” *IEEE Access*, vol. 8, pp. 173723–173731, 2020.

[21] Q. Pu, C. Xu, H. Wang, and L. Zhao, “A novel artificial bee colony clustering algorithm with comprehensive improvement,” *The Visual Computer*, vol. 38, no. 4, pp. 1395–1410, 2022.