IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 9, 2021, accepted January 9, 2022, date of publication January 18, 2022, date of current version February 1, 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3144015*

# Energy-Aware and Trust-Based Secure Routing Protocol for Wireless Sensor Networks Using Adaptive Genetic Algorithm

**YOUJIA HAN**[1], **HUANGSHUI HU**[1,2], **AND YUXIN GUO**[1]

[1]College of Computer Science and Engineering, Changchun University of Technology, Jilin 130012, China
[2]School of Artificial Intelligence, The Tourism College of Changchun University, Jilin 130607, China

Corresponding author: Huangshui Hu (huhs08@163.com)

This work was supported in part by the Science and Technology Development Project of Jilin Province under Grant 20200201009JC and Grant 20210201051GX.

**ABSTRACT** Due to their working environments, limited resources and communication characteristics, wireless sensor networks face some challenges including energy optimization and security enhancement to extend the network lifetime and guarantee the network security. Therefore, an energy-aware and trust-based routing protocol for wireless sensor networks using adaptive genetic algorithm called TAGA is proposed to not only resist common routing attacks and special trust attacks, but also minimize the energy consumption caused by data transmission. To this end, TAGA constructs the nodes' comprehensive trust values based on their direct trust values considering the volatilization and adaptive penalty factors, and indirect trust values with the filtering mechanisms. In addition, a novel threshold function is presented to select the optimal cluster heads, which considers the dynamic changes of the nodes' comprehensive trust values and residual energy. Finally, a genetic algorithm with adaptive crossover probability and mutation probability is applied to find the optimal secure routing for the cluster heads. The simulation results show that TAGA can reduce the number of packets discarded by malicious nodes when facing common attacks and special trust attacks, and effectively improve the energy efficiency compared to the relative secure routing protocols EOSR and IASR.

**INDEX TERMS** Wireless sensor networks, secure routing, comprehensive trust, direct trust.

## I. INTRODUCTION

Wireless sensor networks (WSNs) are comprised of multitudinous sensor nodes that are low-priced, low-power and miniature characteristics [1]–[3]. These nodes are data-centric responsible for collecting relevant data in the target area and transmitting the data to the sink node (base station or control center) in a single-hop or multi-hop manner [4], [5]. In recent years, wireless sensor networks have been widely spread in aerospace, industry, home, battlefield, and many other fields [6]–[8]. Usually, sensor nodes are deployed in harsh environments or unattended areas, which makes their routing protocols susceptible to all kinds of attacks [9]. Such attacks can be classified as external and internal attacks [10], [36]. To enable WSNs to operate in a healthy and secure environment, the security mechanisms based on cryptography and identity verification are proposed to resist external attacks on WSNs. However, such defense mechanisms cannot defend against attacks inside the network [11], [12]. This is because it is a prerequisite for realising these schemes that nodes in a network are cooperative and fully reliable [13]. Moreover, they require sophisticated calculations and large amounts of memory, which can incur higher energy overhead. Therefore, security mechanisms using trust have been proposed to solve internal attacks on WSNs have been shown to be feasible [14], [15].

Trust-based security mechanisms predict the behavior of the node in the next moment based on its historical behavior [16], [17]. They quantify the behavior of nodes by building models. The more good behaviors a node has, the higher the trust value and security will be. However, traditional trust-based security mechanisms also have some drawbacks, such as not being able to defend against multiple types of attacks simultaneously, not being fast enough to identify maliciousness, and high energy consumption. More importantly, there is a spear and a shield, and trust attacks (on-off and

The associate editor coordinating the review of this manuscript and approving it for publication was Ghufran Ahmed.

bad-mouth attacks) are designed to target trust-based security mechanisms. Its purpose is to destroy the trust evaluation mechanism, thereby paralyzing these defense mechanisms. Therefore, normal nodes will also be evaluated as malicious nodes. Therefore, how to resist both common attacks and trust attacks, as well as to find a secure and energy-saving route for the network, is one of the important issues that many researchers explore.

The routing protocol based on trust and adaptive genetic algorithm proposed in this paper resists common routing attacks and special trust attacks by building a trust model. For example, common attacks include black hole, selective forwarding, sinkhole and hello flooding attacks [18], [19]; trust attacks include on-off and bad-mouth attacks [20]. TAGA design the fitness function of an adaptive genetic algorithm (AGA) according to node comprehensive trust value, energy trust value and hop count as parameters to find secure and energy-saving routes. In addition, adaptive crossover probability, adaptive variation probability and random crossover mapping method are introduced in the AGA to enhance the diversity of the population, so as to accelerate the convergence of chromosomes and avoid the local optimum phenomenon.

The main contributions of this paper are as follows:

TAGA is developed by integrating trust security mechanism and AGA so that routing can take into account both security and energy saving.

TAGA is improving security by building an adaptive trust model to evaluate the comprehensive trust value of each node to resist common attacks and special trust attacks.

The cluster heads (CHs) selection threshold is improved according to the dynamic changes of trust value and energy, so as to avoid malicious nodes acting as CHs.

The rest of the paper is organized as follows: Section 2 presents the past work of many researchers; Section 3 describes the improved trust proposed in this paper; Section 4 presents the energy model of TAGA; Section 5 describes the pathfinding process of TAGA in detail; Section 6 verifies the performance of TAGA through simulation experiments; Section 7 concludes the paper.

## II. RELATED WORK

In [21], a security and trust-aware routing scheme is proposed, which differs from other methods in that it acquires trust by using fuzzy logic. The fuzzy logic contains two inputs and one output. The source and destination are used as input to the fuzzy logic, and TU (trust and untrust) is used as output. The fuzzy logic is used to distinguish between normal and unnatural sensors, where unnatural sensors can be considered as internal attacks. For the routing of this scheme, they utilize the MDS-MAP (multidimensional scaling mapping) algorithm to decide the best path with less error. But this scheme requires the support of fuzzy system, Dijkstra algorithm, MDS-MAP algorithm, so the complexity is high. In [22], a trust-based secure directed diffusion routing protocol (TSDDR) is proposed to guarantee the confidentiality of data during transmission. The trust model of TSDDR considers two factors: direct trust value and energy consumption. In constructing the direct trust value, they use the Beta model with penalty function $(1-\beta/W)$ and tuning function $(1-1/(\alpha+\delta))$ to help construct the direct trust. Although the penalty function and tuning function are introduced in the direct trust model, the burden from the historical evaluation is not considered. In [5], an algorithm based on ant and trust called MPASR is proposed to alleviate the burden caused by historical evaluation by introducing $\eta$ (weakening factor), as well as to save energy. The trust type of MPASR is modeled as Beta distribution, and the comprehensive trust value is calculated with direct and indirect trust values. Unlike previous studies, MPASR filters out unnecessary third-party nodes in order to save energy when calculating indirect trust values. Then in the route establishment phase, MPASR integrates the trust value, residual energy and delay to obtain a comprehensive pheromone that is an important factor to improve the traditional ant colony algorithm, which in turn finds the most appropriate transmission route for each node in the network.

In [23], a hybrid optimization algorithm called Monarch-Cat Swarm Optimization (M-CSO) is proposed to ensure the effective security of the network. Nodes are identified as secure or not by calculating the tolerance constant according to trust, connectivity and Quality of Service. The opportunity routing of M-CSO is done by integrating the MBO (Monarch Butterfly Optimization) and the CSO (Cat Swarm Optimization), which balances the advantages and disadvantages of both algorithms. However, in the establishment of the trust model, the consideration is not comprehensive enough, which makes the evaluation not accurate enough. In [24], the Secure Quality of Service (QoS) aware Energy Efficient Routing (SQEER) protocol is proposed, which is designed based on trust and energy models. SQEER calculates three types of trust values to improve the security of network communication and the accuracy of evaluation. The protocol calculates the total trust value from direct and indirect trust values. QoS and trust are then applied together in an integrated way to the security CHs selection mechanism. Finally, the optimal path is selected based on path trust, energy and hop count. In [25], an algorithm combining trust and opportunistic routing called ETOR is proposed, which consists of two components. One uses tolerance constants to carefully select safety nodes, and the other selects some of these nodes to form a route. In the routing phase, a new hybrid fitness function is designed to select the optimal secure route. This function specifically involves parameters such as energy, trust, QoS, connectivity, distance, hop count and network traffic. ETOR is superior to M-CSO and SQEER in terms of energy efficiency and trust model design. In [37], an effective fuzzy path selection approach is proposed to reduce the impact of selective forwarding attacks. This approach is roughly divided into two phases: the first phase is when the compromised node is monitored; the second phase is when a new secure route is selected using fuzzy rules. This method takes the average link residual energy and hop count as inputs for fuzzy control and then obtains the candidate chance values for a particular route. Finally, a new disjoint secure route is selected to resist selective forwarding attacks.

In [26], base station (BS) controlled secure routing protocol (BSCSRP) is proposed. The CH selection mechanism of BSCSRP is based on the classical LEACH (Low Energy Adaptive Clustering Hierarchy) protocol. However, the CHs election threshold of BSCSRP only considers the residual energy of nodes. BSCSRP considers direct trust, indirect trust, packet drop trust and attribute trust when constructing a comprehensive trust model. The energy, distance and delay parameters in communication are used as the basis for path selection. But BSCSRP does not consider whether this CH is secure or not when selecting the CH. In [27], a multidimensional secure cluster routing scheme (MSCR) in hierarchical WSNs is proposed to participate in the selection of CHs based on several factors, including security. MSCR first constructs the trust synthesis, the environmental factors, and the security domain. Then, MSCR also takes LEACH protocol as the basic framework to improve the CHs election thresholds by introducing the energy domain, the distance domain, the environment domain and the security domain. Although MSCR contributes in the CHs election phase, all CHs of the network transmit their collected data to the BS in a single-hop manner. Therefore, applying MSCR to large-scale networks shows poor performance. In [28], a trust-based dynamic slicing mechanism (TDSM) is proposed to improve the performance of WSNs. This mechanism evaluates the corresponding trust degree based on data forwarding, latency and packet loss and then averages these three values. The node with the largest communication trust value is used as the CHs in the CHs selection phase; the node with a communication trust value less than 4 is the discard node; the rest are the member nodes. Finally, each node splits its data into different pieces and sends them to the corresponding nodes. The corresponding nodes mix the received slices with their own sensed data and sends them to the corresponding multi-hop link.

None of the above schemes can resist trust attacks. Once a malicious node launches a trust attack, many normal nodes with trust-based security mechanisms will be paralyzed. In [29], an energy-optimized secure routing algorithm (EOSR) is proposed, which is designed as a multi-factor routing method. The comprehensive trust of EOSR is a combination of direct and indirect trust values. In constructing the direct trust model, EOSR considers that the uncooperative behavior may be from malicious or normal nodes. Thus EOSR corrects the accuracy of the direct trust model. In addition, the deviation degree is introduced to filter false recommendations when calculating the indirect trust value. Finally, EOSR combines the AODV (Ad hoc On-demand Distance Vector Routing) protocol, trust, residual energy and hop count to calculate the combined path cost. Thus, the path with the smallest combined path cost is selected as the optimal route. In [30], Information-Aware Secure Routing in WSNs (IASR) is proposed. IASR calculates the direct attack probability based on the communication behavior of nodes and then calculates the indirect attack probability and filters the malicious evaluation value according to the direct attack probability. Finally, direct and indirect trust are simply integrated to get the integrated trust. Using the Dijkstra algorithm as the basis of pathfinding, the total state value of nodes in the whole path is calculated based on the trust value (attack probability) associated with the state (remaining energy and distance to the sink). Therefore, the optimal route with minimum cost can be found. Although these two schemes consider the impact of trust attacks, they do not consider the negative impact of historical evaluation on the current evaluation. In addition, the node with bad behavior is not penalized to speed them up to be identified. Therefore, these two schemes will lose a large number of packets when attacked.

All the acronyms mentioned in this paper are listed in Table 1.

**TABLE 1.** Definition of acronyms.

| Acronyms | Definition |
|---|---|
| WSNs | wireless sensor networks |
| AGA | adaptive genetic algorithm |
| CHs | cluster heads |
| MDS-MAP | multidimensional scaling mapping |
| TSDDR | a trust-based secure directed diffusion routing protocol |
| MPASR | A Multi-Attribute Pheromone Ant Secure Routing |
| M-CSO | Monarch-Cat Swarm Optimization |
| MBO | Monarch Butterfly Optimization |
| CSO | Cat Swarm Optimization |
| QoS | Quality of Service |
| SQEER | the Secure Quality of Service aware Energy Efficient Routing |
| QoS | Quality of Service |
| ETOR | An Energy-Aware Trust and Opportunity Based Routing Algorithm |
| BSCSRP | the Base Station Controlled Secure Routing Protocol |
| BS | base station |
| LEACH | Low Energy Adaptive Clustering Hierarchy |
| MSCR | multidimensional secure cluster routing |
| TDSM | trust-based dynamic slicing mechanism |
| EOSR | energy-optimized secure routing |
| AODV | Ad hoc On-demand Distance Vector Routing |
| IASR | Information-Aware Secure Routing |

## III. TRUST MODEL

Since a normal node is captured as a malicious node, security schemes based on cryptographic mechanisms cannot defend it. Therefore, human-to-human trust relationships are applied to WSNs to evaluate whether or not each node is captured as a malicious node. The higher a node is trusted, the higher its security. The watchdog mechanism is used for reconnaissance to obtain the data source of trust evaluation [19] and mainly monitors the nodes in the route for sending and receiving packets. In TAGA, hierarchical trust values are used to maintain the dynamic behavior of trust.

### A. DIRECT TRUST VALUE

The direct trust value is meant to be obtained by the nodes that personally monitor the behavior of their neighbors. In this paper, The direct trust model is constructed by monitoring the status of neighbor nodes receiving and sending data packets. The direct trust of node i evaluating neighbor node j can be expressed as:

$$D_{ij}^{t} = \psi * PT_{ij}^{t} + (1-\psi) * NT_{ij}^{t} \tag{1}$$

where $PT_{ij}^{t}$ denotes trust values that node i has evaluated node j in the past, i.e., historical trust values. It aims to limit the high trust values evaluated long time ago, thus improving the security, accuracy and speeding up identification in WSNs. $NT_{ij}^{t}$ indicates the trust value of node j evaluated by node i in the current evaluation cycle, and satisfies $NT_{ij}^{t} = R_{j}^{t} + S_{j}^{t}$. $\psi$ and $1-\psi$ are the measuring factor of historical and current trust values, $0 < \psi < 1$, whose values are set in terms of the specific circumstances. For fairness, the value of $\psi$ is usually set to 0.5. $R_j$ and $S_j$ denote under the penalty mechanism the ratio of the number of data received and sent by node j to the total number of packets, respectively, and are expressed as follows:

$$R_j = \frac{\gamma * re_j - rj_j}{me_j} \tag{2}$$

$$S_j = \frac{\gamma * se_j - us_j}{me_j} \tag{3}$$

where, $re_j$ and $se_j$ respectively represent the number of packets that j received and sent. $rj_j$ and $us_j$ denote the number of data that j rejected to received and sent, respectively. Message denotes the total number of packets received and sent by node j. $\gamma$ is the adaptive penalty factor. It serves to speed up the decline of trust value of malicious nodes. The expression formula of $\gamma$ is as follows:

$$\gamma = \frac{-a}{1+e^{-b*(BP_j+c)}} + 1 \tag{4}$$

$$BP_j = \frac{AB_j}{NB_j} \tag{5}$$

where $BP_j$ is the ratio of abnormal behavior $AB_j$ to normal behavior $NB_j$ in the last five trust evaluation cycles. $a$, $b$ and $c$ represent changeable parameters of $\gamma$, respectively, and control the strength of the penalty to node j. Their magnitudes depend on the specifics of the network. When node j is successfully caught at some point, $AB_j$ increases sharply, which leads to a decrease in $BP_j$. Under the action of $BP_j$, $\gamma$ decreases Correspondingly, which results in fast dropping current values. Therefore, the adaptive penalty factor $\gamma$ is helpful to improve the recognition speed.

In the perfect case, when a node becomes a malicious node, it can be identified immediately enough. However, in reality, it takes a period of evaluation cycle to identify it as a malicious node. Therefore, how to shorten the evaluation cycle is one of the issues discussed by many researchers. Malicious nodes are normal nodes with high trust value before they are captured, so the high historical trust value acts as a hindrance to the current evaluation. Therefore, the introduction of volatilization factor $\rho$ to reduce the effect of historical trust value is helpful to speed up the recognition speed. Its value is taken in the range of [0, 0.5]. The expression formula for the historical trust value is as follows.

$$PT_{ij}^{t} = \rho * (D_{ij}^{t-1} + PT_{ij}^{t-1}) \tag{6}$$

### B. INDIRECT TRUST VALUE

Although nodes can be evaluated by direct trust, if the level of information interaction between nodes is not enough or is affected by the channel, direct trust is not accurate enough to measure the trustworthiness of a node [30]. Therefore, it is necessary to introduce indirect trust to enhance the accuracy of trust assessment. The indirect trust value is the trust value provided by the common nodes between the node and the target node. In other words, the direct trust value provided by the common trusted neighbor nodes of node i and node j is used to calculate the indirect trust value of node j. The indirect trust value of node i to node j is expressed as:

$$I_{ij}^{t} = \sum\nolimits_{k \in PB_h} \left(\phi_k * D_{ik}^{t} * D_{kj}^{t}\right) \tag{7}$$

where $PB_h$ stands for the set of trusted nodes that are jointly owned by i and j. The common neighbor nodes of i and j may be trusted or untrusted. Therefore it is necessary to filter the neighbor nodes. Moreover, if a third-party node k launches bad-mouth attacks, the trust mechanism can suffer a devastating disaster due to malicious evaluations. To avoid bad-mouth attacks and enhance the security of the trust mechanism, this paper uses Equation 8 to filter the false evaluations.

$$d_k^{t} = \sqrt{\frac{\sum_{B_x \in B} \left(\overline{D} - D_{kB_x}^{t}\right)^2}{l}} \tag{8}$$

where $\overline{D}$ represents the median that i evaluates node $B_x$. As shown in Figure 1, $B_x$ stands for the node jointly owned by i and third-party node k, $B_x \in B = [B_1, B_2 \cdots B_l]$, and $l$ is the number of common neighbor nodes. $d_k^t$ is related to whether the value recommended by k is reliable. Therefore, the recommendation threshold is set to guarantee the authenticity of the recommended values. If $d_k^t$ is greater than the threshold $\varepsilon$, the value recommended by node k is not adopted by node i and the abnormal behavior of k increases by $l$. Instead, node k is added to $PB_h$. The size of the threshold value $\varepsilon$ depends on the specific application; in this paper, $\varepsilon$ is set to 0.5. Note that the direct trust values recommended by the nodes in the set $PB_h$ are trusted, but it does not mean that these nodes are trusted themselves.

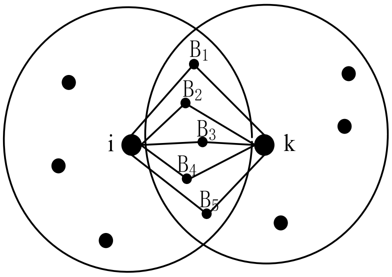


**FIGURE 1.** Common neighbors of node i and third-party node k.

In order to improve the accuracy of trust evaluation, the weight of node k is set as follows:

$$\phi_k = \frac{D_{ik}^{t}}{\sum_{k \in PB_h} D_{ik}^{t}} \tag{9}$$

where node k is from $PB_h$ to avoid malicious destruction of the weights.

### C. COMPREHENSIVE TRUST VALUE

The comprehensive trust value combines the direct and the indirect to ultimately express the basis of whether a node is trusted or not. The higher the comprehensive trust value, as shown in formula 10, the higher the security level and trustworthiness of the node.

$$C_{ij}^t = \begin{cases} (1-\eta) * D_{ij}^t + \eta * I_{ij}^t & \text{if } j \neq k \\ (1-\eta) * I_{ij}^t + \eta * D_{ij}^t & \text{else } j = k \end{cases} \tag{10}$$

$$\eta = \begin{cases} 0.5 & \text{if } d_k^t < \varepsilon \\ \left| D_{ij}^t - I_{ij}^t \right| / D_{ij}^t & \text{else } d_k^t \geq \varepsilon \end{cases} \tag{11}$$

where $\eta$ is their weight coefficient. According to Equation 11, if node k, the common neighbor of node i and node j, initiates bad-mouth attack, the calculation prefers the direct value. Conversely, for fairness, the weight coefficients for both direct and indirect are 0.5. The node that initiates the bad-mouth attack may be either a third-party node k or the evaluated. If it is the former, the calculation of the comprehensive trust value focuses more on the result of the evaluator's assessment. On the contrary, the direct and indirect trust values are considered in combination.

## IV. ENERGY MODEL

Usually, the nodes in WSNs are deployed in harsh and unattended environments, which prevents the nodes from replenishing their energy. Therefore, it is important to consider the energy consumption while enhancing the security of the network to avoid rapid energy depletion of the nodes. In this paper, we use a similar energy model as used in [30], [34]. The energy consumed by node i to send $s$-bits of data to node j at a distance d is as follows:

$$Es_j = \begin{cases} s * E_{elec} + s * \varepsilon_{fs} * d^2 & d < d_0 \\ s * E_{elec} + s * \varepsilon_{mp} * d^4 & d \geq d_0 \end{cases} \tag{12}$$

The energy consumed by node j to receive the $s$-bits data sent by node i is as follows:

$$Er_j = s * E_{elec} \tag{13}$$

where $E_{elec}$ denotes the energy cost of transmitting 1 bit by the transmitter; $\varepsilon_{fs}$ and $\varepsilon_{mp}$ denotes the energy cost of the free-space and multi-path fading models, respectively; Additionally, $d_0$ is the threshold value for an amplifier to adjust its power.

$$d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{mp}} \tag{14}$$

The energy consumption of $s$-bits data aggregation is:

$$E_{DA} = s * E_{pb} \tag{15}$$

The energy cost to fuse 1 bit of data is $E_{pb}$.

Assuming that the initial energy of each node is $E_0$, the remaining energy of each node is

$$RE_j = RE_j - Er_j - Es_j - E_{DA} \tag{16}$$

Thus, for j, the energy trust value is:

$$E_j = \frac{RE_j}{E_0} \tag{17}$$

## V. SECURE ROUTING

The third section describes the trust evaluation mechanism of TAGA, which aims to evaluate the security level of individual nodes. This section, on the other hand, aims at finding the safest and most energy-efficient route through an AGA to achieve secure routing in WSNs.

### A. SELECTION OF CHs

TAGA is a hierarchical routing protocol that first selects the nodes with the highest combined capability as the CHs based on the comprehensive trust value and the remaining energy. Since the main function of the CHs are to receive and forward the data sent by the nodes in the cluster and to forward the data from other CHs, the node with high overall quality must be selected as the CH. In traditional trust-based secure routing protocols, only nodes with high security are considered as CHs, which is deficient. Because if a node does not have much-remaining energy but has a high-security level, then this node is bound to take the role of CH. So, in this case, these nodes are always selected as CHs, and then his energy decreases very quickly.

The secure routing in this paper is in terms of the framework of classical LEACH protocol, a hierarchical network topology with operational phases including cluster construction phase and stable data transmission phase. However, in these two phases, the selection of CHs and data transmission on the path make the WSNs full of crisis and energy wastage. Therefore, it is necessary to improve the CHs selection mechanism and path planning method. The improved CHs selection mechanism is shown in the following equation.

$$T(j) = \begin{cases} \dfrac{p}{1 - p(r \bmod (1/p))} * ((1-\varphi) * \overline{C_j^t} + \varphi * E_j') \\ \quad j \in G \\ 0 \quad j \notin G \end{cases} \tag{18}$$

where, $p$ is the percentage of the number of CHs in the network to the total number of nodes, $r$ is the number of rounds the network is running, and $G$ is the set of nodes that were not selected as CHs in the previous round and have residual energy greater than the average. In order to select high quality CHs with high security and sufficient energy, the average of the comprehensive trust value $\overline{C_j^t}$ and the min-max normalization of energy trust value $E_j'$ are introduced. $\varphi$ is adaptive weight. Traditional trust-based security schemes use fixed weights, which makes them inadequate to adapt to changes in the network. Since in the early stage of network operation, the comprehensive trust value of nodes has not yet reached the state of convergence, the comprehensive trust value is relatively volatile, which makes it poor to judge the security of nodes. Therefore, in the early stages of the network, energy trust should be favored more compared to the comprehensive trust. After the initial stage of the network, the assessed value of the nodes is stable and the energy in the network gradually decreases with the operation of the network. Therefore gradually focus on the role played by the comprehensive trust value in CHs selection. The formula for the weighting factor is as follows.

$$\varphi = \begin{cases} \mathrm{Q} & if\ r \leq r_1 \\ \dfrac{1-\sqrt{1+0.75*(\xi(r)-2)^2}}{\xi(r)-2} & else\ r > r_1 \end{cases} \tag{19}$$

where $Q$ is a constant, which takes values between 0.5 and 1. $r_1$ indicates the number of rounds the network is running when the standard deviation of the comprehensive trust value of the nodes is less than a specific threshold $\chi$. $\xi(r)$ indicates the total energy consumed by the network as it runs.

$$\xi(r) = \sum_{j=1}^{n} E''_j - \sum_{j=1}^{n} E_j \tag{20}$$

where $E''_j$ denotes the energy trust value of j when the number of running rounds is $r_1$.

### B. ROUTING BASED ON AGA

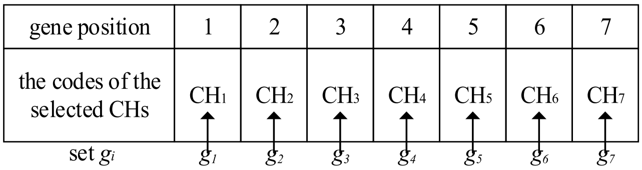
Since the single-hop approach is used in the stable phase of LEACH protocol, but this approach is not suitable for large-scale networks. However, if the multi-hop approach is used, the CHs need to forward the collected data through multiple nodes to the BS, which will also have a crisis. So the paths sought by the CHs must be safe and energy-efficient.

Currently, the research of AGA has been relatively mature and extensive [31], but few of them have been combined with trust mechanisms. Genetic manipulation involves four main processes: coding, selection, crossover, and mutation. In TAGA, real number coding is used instead of binary coding to represent chromosomes, and a new fitness function is designed to determine the quality of chromosomes [35]. In addition, random crossover and dynamic mutation methods from the literature [32], [33] are used to enhance the population diversity and improve the algorithm convergence.

#### 1) ENCODING

The AGA uses different coding methods will produce the effect, so the AGA is first faced with how to use the appropriate chromosome coding method. The most commonly used encoding methods are binary encoding and real encoding, and real encoding has the advantage of higher accuracy. Assuming that the network consists of N nodes, then each node is given a unique positive integer (or ID) between 1 and N, and the BS is denoted as N+1. Thus when $nh$ nodes are selected as CH, then the genes of this chromosome are composed of their respective codes, and the chromosome length is $nh$.

To expedite the aggregation of the AGA and obtain an efficient solution, the gene position of each source CH has its own corresponding set $g_i$, as shown in Figure 2. Only $CH_j$ that satisfies the following three conditions at the same time can be added to the set gene $g_i$ of source $CH_i$: (i) $CH_j$ is within the communication range of $CH_i$; (ii) the distance from $CH_i$ to the BS should be greater than the distance from $CH_j$ to the BS; (iii) the distance between $CH_i$ and $CH_j$ should be smaller than the distance from $CH_i$ to the BS.



| gene position | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| the codes of the selected CHs | $CH_1$ | $CH_2$ | $CH_3$ | $CH_4$ | $CH_5$ | $CH_6$ | $CH_7$ |
| set $g_i$ | $g_1$ | $g_2$ | $g_3$ | $g_4$ | $g_5$ | $g_6$ | $g_7$ |

**FIGURE 2. Chromosome structure diagram.**

#### 2) SELECTION OPERATOR

The reproduction of species in nature is a process of survival of the fittest. Similarly, in AGA, the operation of selecting high quality individuals and eliminating low quality individuals from a large number of individuals is called selection operator. At present, the roulette wheel selection method is the most common and simplest method of selection operator in AGA. The expression formula for the probability of each individual being selected is as follows:

$$P(x) = \frac{f(x)}{\sum\limits_{x=1}^{x=M} f(x)} \tag{21}$$

where $x$ and $f(x)$ represent individual $x$ and the fitness value of individual $x$, respectively. $M$ represents the size of the population.

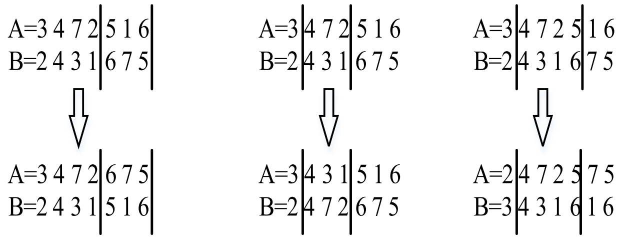
#### 3) CROSSOVER OPERATOR

The crossover operator means that two-parent individuals exchange corresponding genes to form the next generation. Its purpose is to enhance population diversity and promote population evolution. The traditional crossover operator randomly selects two individuals based on the crossover probability and randomly generates crossover positions for exchanging some genes.

To further enhance the population diversity and the search range, the random cross mapping method is introduced to break the limitation of the number of fixed crossover points in traditional genetic algorithms [32]. The random cross mapping method means that the crossover length is fixed as half of the chromosome length, but the crossover start point and endpoint are not fixed. Suppose the chromosome length is $L$ and the crossover length is:

$$L' = \left\lfloor \frac{L}{2} \right\rfloor \tag{22}$$

First, a starting crossover position is randomly selected, and then the end crossover point is extended $L'$ units backwards along the chromosome from the starting point. For example, two chromosomes A and B of length 7 (shown in Figure 3), then a crossover segment of crossover length 3 is generated under the random crossover mapping method. Figure 3 shows the three cases generated under this method.


A=3 4 7 2|5 1 6
B=2 4 3 1|6 7 5

A=3 4 7 2|6 7 5
B=2 4 3 1|5 1 6

A=3|4 7 2|5 1 6
B=2|4 3 1|6 7 5

A=3|4 3 1|5 1 6
B=2|4 7 2|6 7 5

A=3|4 7 2 5|1 6
B=2|4 3 1 6|7 5

A=2|4 7 2 5|7 5
B=3|4 3 1 6|1 6

**FIGURE 3.** The random crossover mapping method.

In addition to the above random cross mapping method, crossover probability and mutation probability can also increase population diversity. However, the fixed crossover probability preset by traditional genetic algorithms may lead to low population diversity in the early stages of the algorithm, which may lead to an early convergence situation. In the process of population evolution, different individuals behave differently, then the crossover probability of the population should also vary according to the fitness of each individual. The adaptive crossover probability [33] involving the number of evolutionary generations and the fitness values of the corresponding generations are shown below.

$$pc = \begin{cases} pc_{\min} - \dfrac{(pc_{\max} - pc_{\min})}{1 + e^{-it}} * \dfrac{f - f_{avg}^{it}}{f_{\max}^{it} - f_{avg}^{it}}, & if\ f \geq f_{avg}^{it} \\ pc_{\max}, & otherwise \end{cases} \tag{23}$$

where, $pc_{\min}$ and $pc_{\max}$ represents the maximum and minimum values of crossover probability, respectively; $f_{avg}^{it}$ and $f_{max}^{it}$ denote the mean and maximum of fitness values for all individuals in the $it$th generation, respectively. $f$ denotes the fitness value.

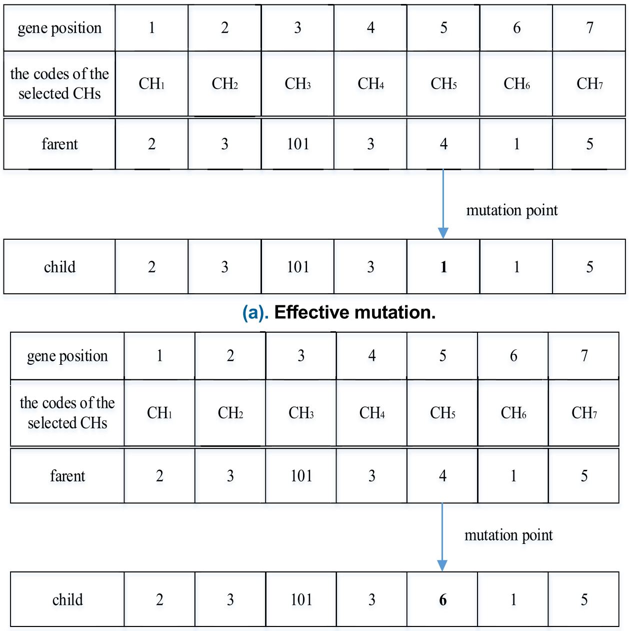
### 4) MUTATION OPERATOR

When the population diversity is low, all individuals tend to be optimal, which is likely to fall into local optimization in advance. So the mutation probability should be increased to break through the convergence state of individuals. The formula for adaptive mutation is as follows:

$$pm = \begin{cases} pm_{\max} - \dfrac{(pm_{\max} - pm_{\min})}{1 + e^{it}} * \dfrac{f - f_{avg}^{it}}{f_{max}^{it} - f_{avg}^{it}}, \\ \qquad if\ f \geq f_{avg}^{it} \\ pm_{\max}, \quad otherwise \end{cases} \tag{24}$$

where $pm_{\min}$ and $pm_{\max}$ represents the maximum and minimum values of mutation probability, respectively.

To avoid invalid mutations, the variant code at each gene position must be chosen randomly from the corresponding set $g_i$. Table 2 shows the set corresponding to each gene position, for example the set corresponding to position 5 is $g_5 = [4, 1, 3]$. Figure 4(a) represents the effective mutation that the gene code at position 5 is mutated from 4 to 1. Figure 4(b) indicates the invalid mutation that 4 mutates to 6 at gene position 5.


| gene position | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| the codes of the selected CHs | CH1 | CH2 | CH3 | CH4 | CH5 | CH6 | CH7 |
| farent | 2 | 3 | 101 | 3 | 4 | 1 | 5 |

mutation point

| child | 2 | 3 | 101 | 3 | **1** | 1 | 5 |
|---|---|---|---|---|---|---|---|

(a). Effective mutation.

| gene position | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| the codes of the selected CHs | CH1 | CH2 | CH3 | CH4 | CH5 | CH6 | CH7 |
| farent | 2 | 3 | 101 | 3 | 4 | 1 | 5 |

mutation point

| child | 2 | 3 | 101 | 3 | **6** | 1 | 5 |
|---|---|---|---|---|---|---|---|

(b). Invalid mutation.

**FIGURE 4.** Gene mutation.

**TABLE 2.** Gene set.

| Gene set $g_i$ | Eligible next hop CHs |
|---|---|
| $g_1$ | 3, 2 |
| $g_2$ | 3, 101 |
| $g_3$ | 2, 101 |
| $g_4$ | 1, 3 |
| $g_5$ | 4, 1, 3 |
| $g_6$ | 1, 2 |
| $g_7$ | 5, 6 |

### 5) FITNESS FUNCTION

Each chromosome is potentially the optimal solution, but the chromosome with the highest fitness value at the end of the iteration is the optimal route. As shown in Figure 5, assuming that the third row represents the chromosome with the highest fitness, the optimal route for the source cluster head $CH_1$ is $OR_1 = \{1, 3, 101\}$. Similarly, the optimal path for source cluster head CH7 is $OR_7 = \{7, 6, 2, 101\}$.

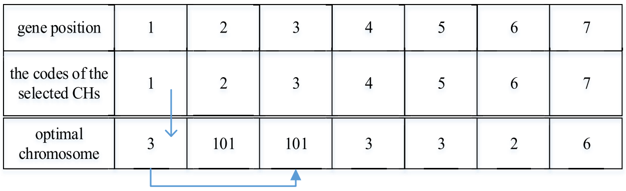

| gene position | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| the codes of the selected CHs | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| optimal chromosome | 3 | 101 | 101 | 3 | 3 | 2 | 6 |

**FIGURE 5.** Routing path represented by chromosome.

TAGA mainly uses two dimensions, the average comprehensive trust value on the link and the energy consumed by CH, to compose the fitness function so as to find a path with high security and small energy cost at the same time. The fitness function for the $k$th gene position in a certain chromosome is as follows:

$$f\left(OR_k\right)=\frac{\sum_{CH_i\in OR_k,i=1}^{H_k-1}C_{CH_iCH_{i+1}}^{t}}{H_k-1}+1-\frac{E'_{OR_k}-\mathrm{E}'_{OR\_\min}}{E'_{OR\_\max}-E'_{OR\_\min}}\tag{25}$$

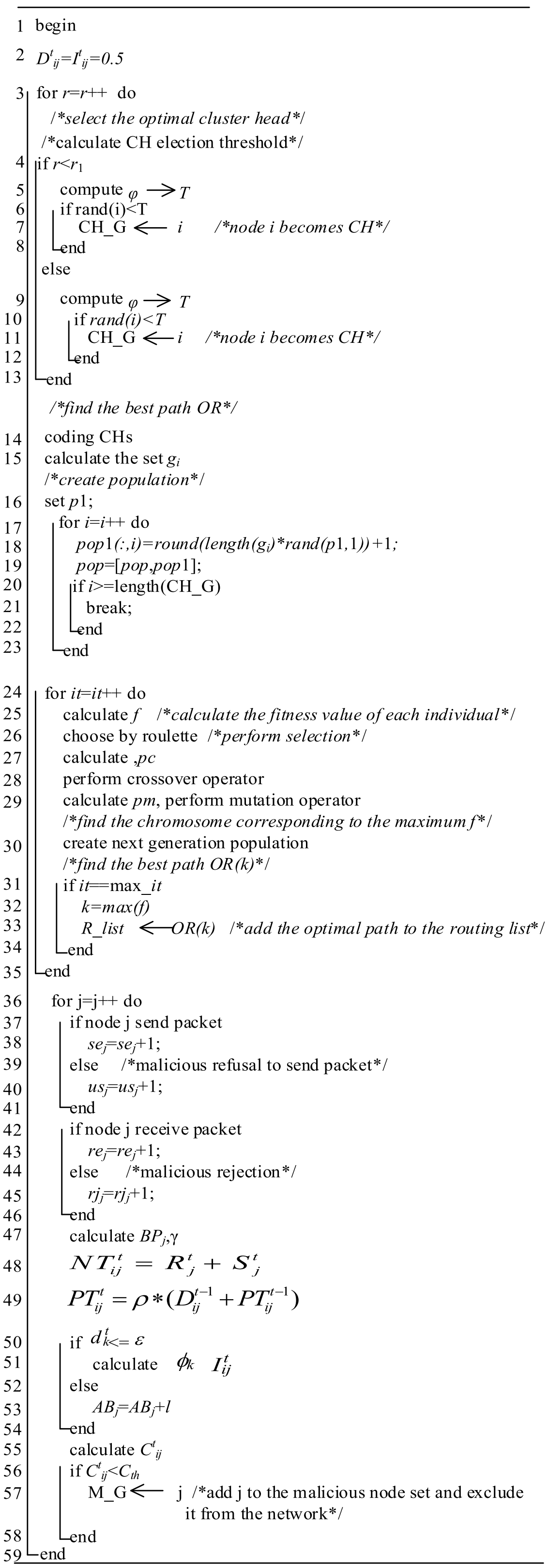
where $OR_k$ represents the link of the $CH_k$ that occupies the $k$-th gene position; $H_k$ is the number of hops of this link; $C_{CH_iCH_{i+1}}^{t}$ denotes the comprehensive trust value that the previous hop $CH_i$ evaluates the next hop $CH_{i+1}$ on this link; $E'_{OR_k}$ denotes the energy consumption on the link $OR_k$; $E'_{OR\_\max}$ and $E'_{OR\_\min}$ represent the maximum and minimum energy consumption of all links in all chromosomes, respectively. Therefore, the fitness function of the chromosome is:

$$f=\sum_{k=1}^{nh}f\left(OR_k\right)\tag{26}$$

where $nh$ is the length of the chromosome.

### C. THE PSEUDO-CODE OF TAGA

The pseudo-code of TAGA is as follows:



```
begin
$D^t_{ij}=I^t_{ij}=0.5$
for $r=r{+}{+}$ do
   /*select the optimal cluster head*/
  /*calculate CH election threshold*/
  if $r<r_1$
    compute $\varphi \longrightarrow T$
    if rand(i)<T
      CH_G $\longleftarrow$ $i$      /*node i becomes CH*/
    end
  else
    compute $\varphi \longrightarrow T$
      if $rand(i)<T$
        CH_G $\longleftarrow$ $i$    /*node i becomes CH*/
      end
  end
   /*find the best path OR*/
  coding CHs
  calculate the set $g_i$
  /*create population*/
  set $p1$;
    for $i=i{+}{+}$ do
      $pop1(:,i)=round(length(g_i)*rand(p1,1))+1;$
      $pop=[pop,pop1]$;
      if $i>=$length(CH_G)
        break;
      end
    end

  for $it=it{+}{+}$ do
    calculate $f$   /*calculate the fitness value of each individual*/
    choose by roulette  /*perform selection*/
    calculate ,$pc$
    perform crossover operator
    calculate $pm$, perform mutation operator
    /*find the chromosome corresponding to the maximum f*/
    create next generation population
    /*find the best path OR(k)*/
    if $it==$max_$it$
      $k=max(f)$
      $R\_list$ $\longleftarrow OR(k)$   /*add the optimal path to the routing list*/
    end
  end
   for j=j++ do
     if node j send packet
       $se_j=se_j+1$;
     else    /*malicious refusal to send packet*/
       $us_j=us_j+1$;
     end
     if node j receive packet
       $re_j=re_j+1$;
     else    /*malicious rejection*/
       $rj_j=rj_j+1$;
     end
     calculate $BP_j$,γ
     $NT^t_{ij} = R^t_j + S^t_j$
     $PT^t_{ij}=\rho*(D^{t-1}_{ij}+PT^{t-1}_{ij})$

     if $d^t_{k}<=\varepsilon$
        calculate  $\phi_k$  $I^t_{ij}$
     else
        $AB_j=AB_j+l$
     end
     calculate $C^t_{ij}$
     if $C^t_{ij}<C_{th}$
       M_G$\longleftarrow$  j /*add j to the malicious node set and exclude
                             it from the network*/
     end
end
```

**FIGURE 6.** The pseudo-code of TAGA.

### D. TIME COMPLEXITY OF TAGA

The proposed TAGA uses comprehensive trust values to evaluate the security of individual nodes and adaptive genetic algorithms to evaluate the security of links, thus improving the network's ability to cope with attacks. Therefore its time complexity can be expressed as O(TAGA) = O(time complexity of the trust model + time complexity of AGA). The trust model of TAGA is to construct the comprehensive trust by combining direct and indirect trust values. Assuming that the network has n nodes, the time complexity of constructing direct trust is O(n*s), where s is the number of neighbor nodes. Since indirect trust is obtained by multiplying the corresponding direct trust, the time complexity of indirect trust is O (n * s). Similarly, the time complexity of comprehensive trust is O(n*s). So, O(time complexity of the trust model)=O(n*s). Encoding chromosomes with length k during the execution of the adaptive genetic algorithm produces a time complexity of O(k). The time complexity that AGA generates the initial population is O(k*p),where p is the population size. The average time complexity of computing the fitness value is O(p*k$^2$), while it is O(p*k) and O(p*k$^2$) in the best and worst cases, respectively. The time complexity of the crossover and variation operations are O(p/2) and O(p), respectively. After performing I iterations, the time complexity of AGA is O(k+k∗p+I*(k$^2$+3/2)*p), which is O(I*p*k$^2$). The number of CHs in the network is much less than the number of nodes, so the time complexity of AGA can be simplified to O(n$^2$). Therefore the overall time complexity is O(TAGA)=O(n*s+n$^2$). Since s is less than n, O(TAGA)=O(n$^2$).

## VI. SIMULATION

This section focuses on verifying the performance of TAGA by MTALB in the face of different attacks and comparing it with IASR and EOSR. We randomly deploy 100 nodes in an area of 100*100$m^2$. The trust threshold is set to 0.35. The simulation time is in rounds. It is assumed that some internal nodes are caught as malicious nodes and then launch the attacks, when the network is running to the 100th round. Other simulation parameters are set in Table 3.

**TABLE 3. Simulation parameters.**

| Parameter | Value |
|---|---|
| $E_0$ | 1J |
| Control packet size | 1000 bits |
| Data packet size | 4,000 bits |
| Number of nodes | 100,1000 |
| Area size | 100m*100m,1000m*1000m |
| Initial trust value | 0.5 |
| BS position | (0,0) |
| $E_{pb}$ | 5nj/bit |
| $\chi$ | 0.08 |
| $p$ | 0.1 |
| Population size | 100 |
| Number of iterations | 50 |
| $pc_{\max}, pc_{\min}$ | (0.8,0.5) |
| $pm_{\max}, pm_{\min}$ | (0.05,0.001) |

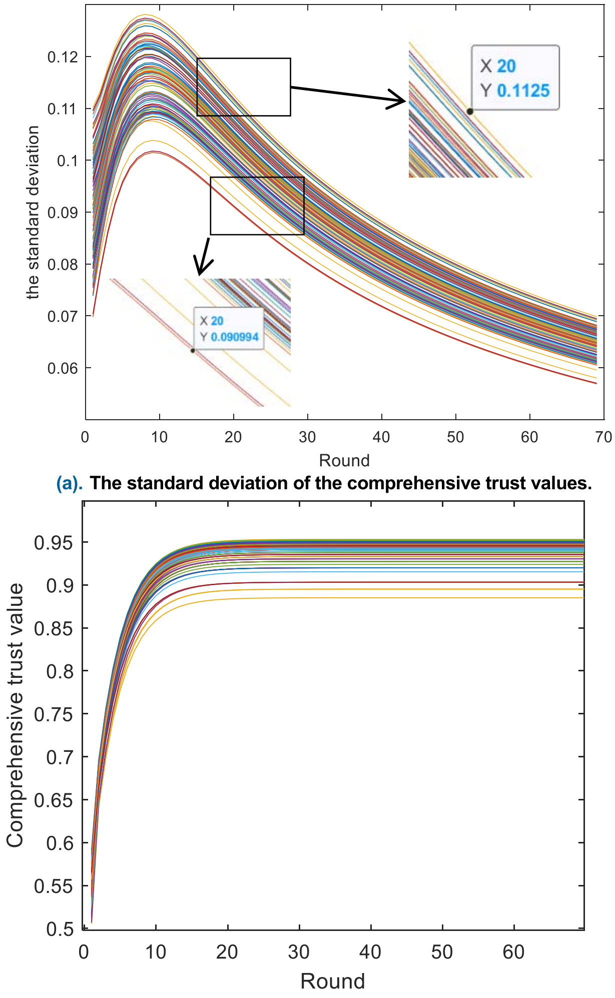
### A. STANDARD DEVIATION THRESHOLD

Figure 7(a) shows the standard deviation of the comprehensive trust values of the 100 nodes in the network as the network runs. Figure 7(b) shows the integrated trust value variation curve for all normal nodes and the convergence after the 20th round. From Figure 7(a), it can be seen that at 20 rounds, the maximum standard deviation is 0.1125, and the minimum standard deviation is 0.090994. So in this paper, the threshold $\chi$ is set at 0.08.

### B. RESIST COMMON ROUTING ATTACKS

#### 1) PACKETS LOST BY MALICIOUS NODES (UNDER BLACK HOLE ATTACKS)

Figure 8 shows that with the increase of malicious nodes, the number of packets discarded by malicious nodes in the network gradually increases. The black hole attack is a malicious node tricking the nodes in the network to establish routing connections with it, resulting in the packets to be forwarded being discarded maliciously. As the malicious nodes discard all the received packets. Therefore, according to Equation (3), when TAGA evaluates the trust value of the malicious nodes, the current trust value will decrease rapidly. Thereby, compared with IASR and EOSR, the trust value of the malicious node quickly drops below the trust threshold. It speeds up the speed of identifying malicious nodes and reduces the number of malicious dropped packets as much as possible. As can be seen in Figure 8, the number of packet loss for TAGA is reduced by an average of 16.96% and 25.48% compared to IASR and EOSR, respectively.



(a). The standard deviation of the comprehensive trust values.

(b). Comprehensive trust value of nodes.

**FIGURE 7. Determination of threshold.**

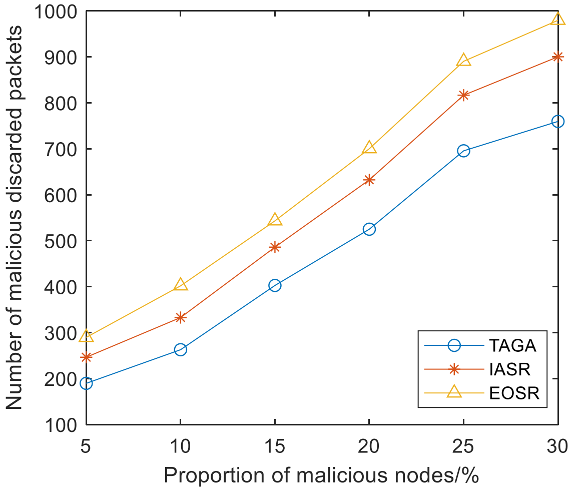


**FIGURE 8. Packets lost by malicious nodes (under black hole attacks).**

#### 2) PACKETS LOST BY MALICIOUS NODES (UNDER HELLO FLOOD ATTACKS)

Figure 9 shows that as the number of malicious nodes launching hello flood attacks increases, the number of

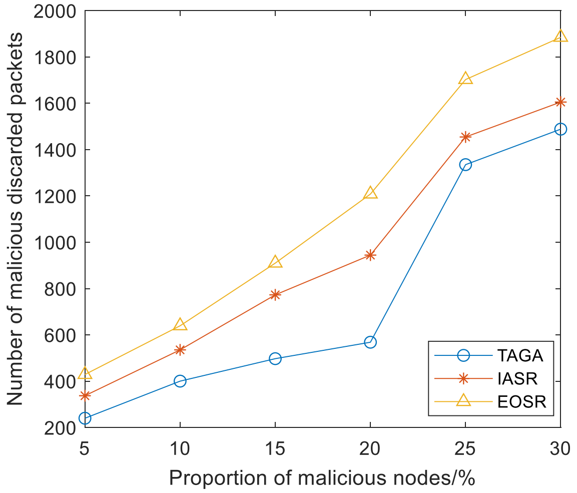


**FIGURE 9.** Packets lost by malicious nodes (under hello flood attacks).

discarded packets by malicious nodes in the network gradually increases. The hello flood attack means that a malicious node broadcasts hello packets using a signal with enough energy to make more distant nodes mistake it as its direct neighbor. As a result, a large number of packets are rejected by the malicious node. TAGA based on equation (2) can make the current trust value decrease quickly for malicious nodes. As can be seen in Figure 9, the number of discarded packets in TAGA is much lower than that of IASR and EOSR. The number of lost packets in TAGA is reduced by 19.83% and 33.13% on average compared with IASR and EOSR, respectively.

### 3) PACKETS LOST BY MALICIOUS NODES (UNDER SELECTIVE FORWARDING ATTACKS)

Figure 10(a) shows that as the number of malicious nodes launching selective forwarding attacks increases, the number of discarded packets by malicious nodes in the network gradually increases. The selective forwarding attack means that the malicious node forwards or discards important packets with a certain probability, and the black hole attack is formed when the packets are discarded with a 100% probability. In this paper, the malicious node discards the received packets with a probability of 70%. From the combination of Figure 10(a) and Figure 8, it is clear that the malicious nodes that initiate the selective forwarding attacks discard more packets than the malicious nodes that initiate the black hole attacks. This is because the malicious nodes both send certain packets and discard some packets, making it more difficult to identify the malicious nodes. As can be seen in Figure 10(a), the number of lost packets for TAGA is reduced by 10.11% and 21.54% on average compared with IASR and EOSR, respectively.

Figure 10(b) shows that the number of discarded packets by malicious nodes in the network gradually decreases as the drop probability increases when there are 30% malicious nodes in the network. This experiment was conducted to verify the impact of the difference in the discard probability on the network. As can be seen from Figure 10(b), the smaller the probability of malicious drops, the greater the damage to the network and the greater the difficulty in identifying malicious nodes.

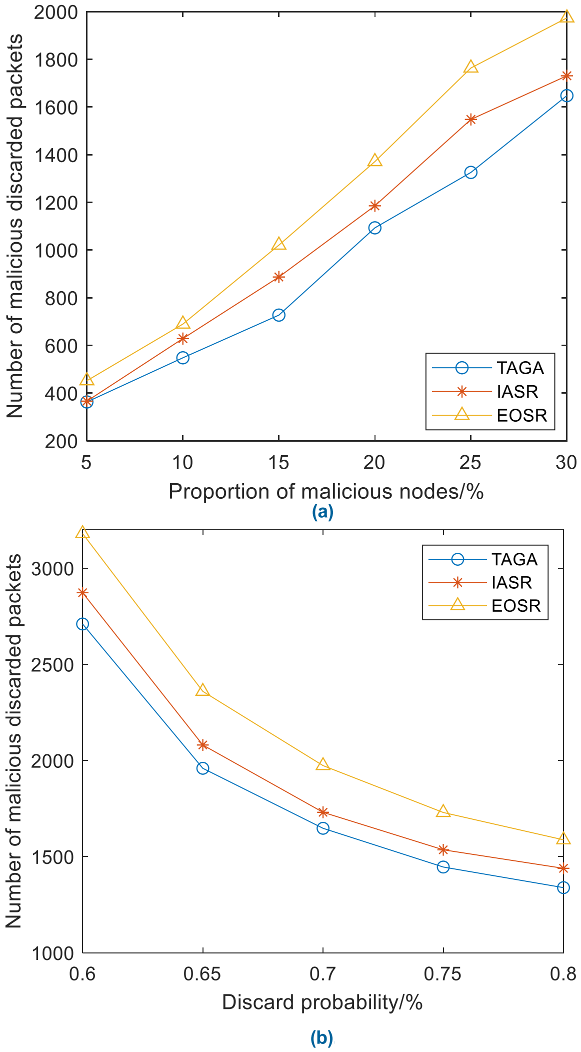


**FIGURE 10.** (a) Packets lost by malicious nodes (under selective forwarding attacks). (b) Number of discarded packets with different discard probabilities.

### 4) PACKETS LOST BY MALICIOUS NODES (UNDER SINKHOLE ATTACKS)

Figure 11 shows that as the number of malicious nodes launching sinkhole attacks increases, the number of packets discarded by malicious nodes in the network gradually increases. The sinkhole attack means that the malicious node creates a black hole centered on itself that attracts all nodes in a specific area to send packets to it. It can be seen from Figure 11 that the number of lost packets for TAGA decrease on average by 20.83% and 36.2% compared to IASR and EOSR, respectively.

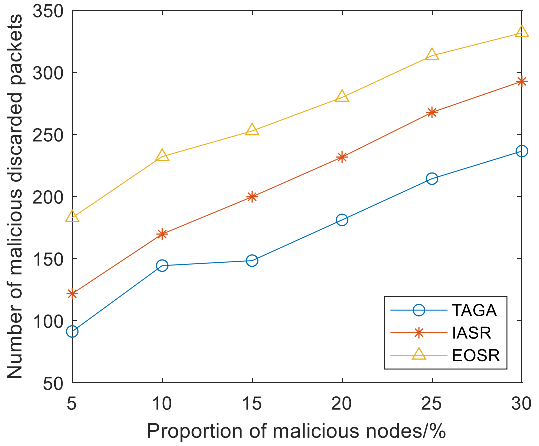


**FIGURE 11.** Packets lost by malicious nodes (under sinkhole attacks).

### C. RESIST SPECIAL TRUST ATTACKS

#### 1) PACKETS LOST BY MALICIOUS NODES (UNDER ON-OFF ATTACKS)

Figure 12 shows that with the increase of malicious nodes launching on-off attacks, the number of packets discarded by malicious nodes in the network gradually increases. The on-off attacks mean that malicious nodes periodically perform both good and malicious behaviors, which brings great challenges to the trust evaluation mechanisms. As can be seen from Figure 12, on-off attacks are more destructive compared to the above attacks. Since TAGA uses the penalty mechanism, TAGA is able to identify malicious nodes quickly and minimize the damage caused by malicious nodes to the network. The number of lost packets in TAGA is reduced by 26.72% and 35.45% on average compared to IASR and EOSR, respectively.

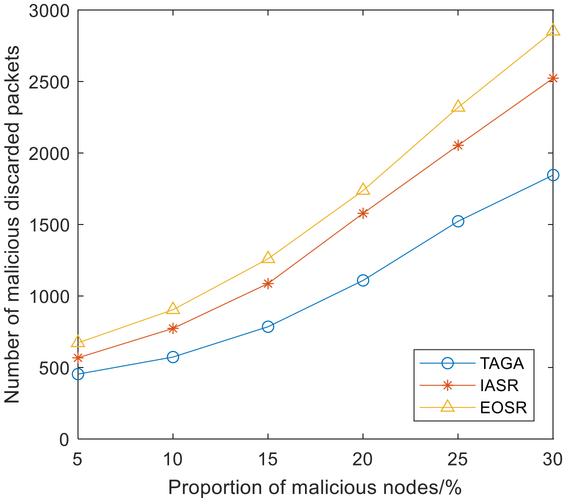


**FIGURE 12.** Packets lost by malicious nodes (under on-off attacks).

#### 2) MALICIOUS RECOMMENDATION (BAD-MOUTH ATTACKS)

Figure 13 shows that with the increase of malicious nodes launching bad-mouth attacks, the number of false trust values spread by malicious nodes in the network gradually increases. The bad-mouth attack is when the evaluator evaluates the target nodes, the malicious node provides false trust value to the evaluator, which causes the evaluator to incorrectly evaluate the trust value of the target nodes, making the trust value of the target nodes decrease. When IASR computes the indirect values, it uses the weighting coefficient $\alpha_m$, which can effectively avoid the influence of malicious recommendations. EOSR uses the deviation degree of indirect trust to filter each false recommendation. However, the trust models of both schemes cannot make the trust value of the adverse nodes that launch the bad-mouth attacks lower. So in both schemes, malicious nodes will continue to spread false recommendations. While the adaptive penalty factor in TAGA can effectively reduce the trust of the adverse nodes, and the credibility constructed by using the median of direct trust can effectively filter the false recommendation values.

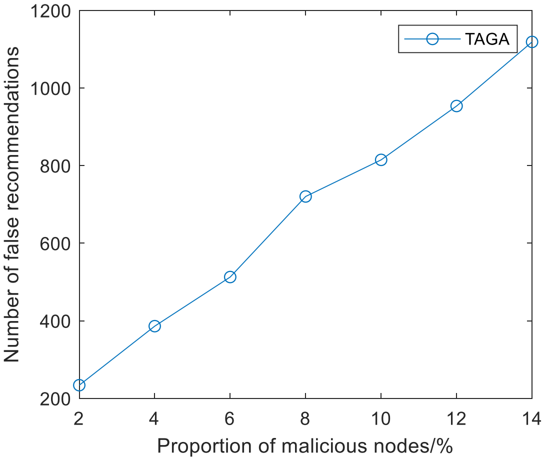


**FIGURE 13.** Malicious recommendation (bad-mouth attacks).

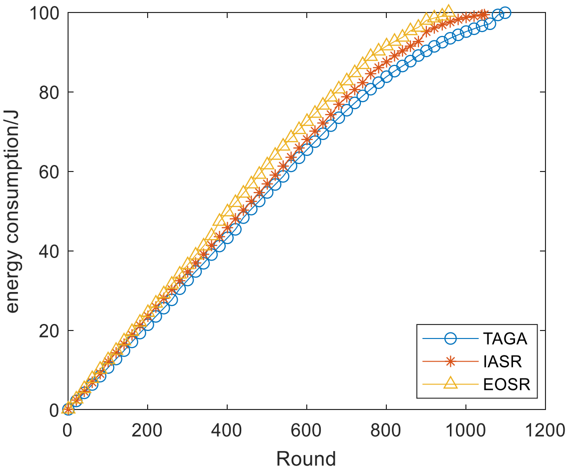


**FIGURE 14.** Energy consumption of network.

### D. ENERGY CONSUMPTION OF NETWORK

Figure 14 shows that the energy consumption in the network increases with the number of rounds. First, TAGA uses

LEACH as the basic framework to improve the CH election threshold based on the trust value and residual energy of nodes, so that nodes with high security and high energy as CHs. Second, TAGA uses AGA to select secure routes for CHs. In constructing the fitness function of the AGA, the optimal path selection is controlled using the comprehensive trust value within each hop and energy consumption. IASR jointly controls the path selection based on the energy consumption from the node to the BS, the remaining energy of the relay node and the trust value. Therefore, the relay node which is far away may be selected. While EOSR may select the node with less hops but less residual energy as a relay node based on the path evaluation formula. Therefore the energy cost of TAGA is lower than that of IASR and EOSR. Compared with IASR and EOSR, the energy efficiency of TAGA has increased by 4.97% and 14.84%.

### E. LARGE-SCALE NETWORK

Table 4 shows the simulation results that 1000 nodes randomly distributed in a 1000m*1000m area, where 30% of the nodes are captured as malicious nodes. IASR and EOSR do not identify malicious nodes since the malicious nodes that launch bad-mouth attacks only spread false trust values and do not participate in discarding packets. Performance generated by TAGA defense black hole attacks, hello flood attacks, selective forwarding attacks, sinkhole attacks, and on-off attacks is improved by 28.27%, 24.15%, 24.95%, 24.86% and 26.29% when compared with IASR, by 31.08%, 26.47%, 28.03%, 27.58% and 28.08% when compared with EOSR.

**TABLE 4.** Simulation results under large-scale network.

| attacks | TAGA | IASR | EOSR |
|---|---|---|---|
| Black hole attacks | 4823 | 6724 | 6998 |
| Hello flood attacks | 5414 | 7138 | 7363 |
| Selective forwarding attacks | 8648 | 11523 | 12016 |
| Sinkhole attacks | 1321 | 1758 | 1824 |
| On-off attacks | 11244 | 15254 | 15633 |
| Bad-mouth attacks | 5463 | -- | -- |

### F. SUMMARY TABLE

After the above simulations for different scenarios, we summarize the proposed TAGA scheme in Table 5. The total discards in Table 5 shows that the nodes that initiate the on-off attacks drop the most packets and the sinkhole attacks drop the least packets. Both black hole attacks and sinkhole attacks discard all received packets; selective forwarding attacks and on-off attacks discard some packets. This shows that it is easier to defend malicious nodes with a large number of dropped packets and harder to defend malicious nodes with partially dropped packets. It can be seen from the average discards in Table 5 that the average number of packets discarded by malicious nodes in the scenario of 1000m*1000m is less than 100m*100m. This is because the number of packets dropped by each malicious node in each evaluation cycle increases significantly leading to the rapid identification of malicious nodes. When TAGA faces black hole attacks, Hello flood attacks, selective forwarding attacks, sinkhole attacks, and on-off attacks in the 1000m*1000m scenario, the average number of packet loss is 36.48%, 63.6%, 47.49%, 44.09% and 39.09% lower than that in the 100m*100m scenario, respectively.

**TABLE 5.** Summary tables.

| TAGA | Total discards | | Average discards | |
|---|---|---|---|---|
| Scenario | 1000m*<br>1000m | 100m*<br>100m | 1000m*<br>1000m | 100m*<br>100m |
| Black hole attacks | 4823 | 759 | 16.07 | 25.3 |
| Hello flood attacks | 5414 | 1488 | 18.05 | 49.6 |
| Selective forwarding attacks | 8648 | 1647 | 28.83 | 54.9 |
| Sinkhole attacks | 1321 | 236 | 4.4 | 7.87 |
| On-off attacks | 11244 | 1846 | 37.48 | 61.53 |

## VII. CONCLUSION

In this paper, an energy-aware and trust-based routing protocol for wireless sensor networks using adaptive genetic algorithm called TAGA is presented to resist common routing attacks and special trust attacks so as to resist multiple attacks, improve the speed of identifying the attackers, and select secure and energy efficient routes. In TAGA, adaptive penalty factors, volatilization factors and filtering mechanisms are designed to construct the comprehensive trust values, which are used to evaluate the security performance of the nodes. Then an improved adaptive genetic algorithm with a novel CH election threshold is applied to select the secure and high-energy nodes as CHs. Finally, an adaptive genetic algorithm is adopted to find the optimal path for each CH. The simulation results verify the effectiveness of TAGA, and indicate that it can effectively decrease the impact of the malicious nodes, reduce the number of lost packets, and improve the network energy utilization.

## REFERENCES

[1] W. Fang, W. Zhang, W. Yang, Z. Li, W. Gao, and Y. Yang, "Trust management-based and energy efficient hierarchical routing protocol in wireless sensor networks," *Digit. Commun. Netw.*, vol. 7, no. 4, pp. 470–478, Nov. 2021, doi: 10.1016/j.dcan.2021.03.005.

[2] A. Rachedi and A. Hasnaoui, "Advanced quality of services with security integration in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 15, no. 6, pp. 1106–1116, Apr. 2015.

[3] G. D. Devanagavi, N. Nalini, and R. C. Biradar, "Secured routing in wireless sensor networks using fault-free and trusted nodes," *Int. J. Commun. Syst.*, vol. 29, no. 1, pp. 170–193, Jan. 2016.

[4] K. Thangaramya, K. Kulothungan, S. I. Gandhi, and M. Selvi, "Intelligent fuzzy rule-based approach with outlier detection for secured routing in WSN," *Soft Comput.*, vol. 24, no. 21, pp. 16483–16497, Apr. 2020.

[5] L. Zhang, N. Yin, X. Fu, Q. Lin, and R. Wang, "A multi-attribute pheromone ant secure routing algorithm based on reputation value for sensor networks," *Sensors*, vol. 17, no. 3, p. 541, Mar. 2017.

[6] V. Gomathy, N. Padhy, D. Samanta, M. Sivaram, V. Jain, and I. S. Amiri, "Malicious node detection using heterogeneous cluster based secure routing protocol (HCBS) in wireless adhoc sensor networks," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 11, pp. 4995–5001, Feb. 2020.

[7] G. Thahniyath and M. Jayaprasad, ‘‘Secure and load balanced routing model for wireless sensor networks,’’ *J. King Saud Univ.-Comput. Inf. Sci.*, Oct. 2020, doi: 10.1016/j.jksuci.2020.10.012.

[8] K. Hamouid, S. Othmen, and A. Barkat, ‘‘LSTR: Lightweight and secure tree-based routing for wireless sensor networks,’’ *Wireless Pers. Commun.*, vol. 112, no. 3, pp. 1479–1501, Jan. 2020.

[9] M. Rathee, S. Kumar, A. H. Gandomi, K. Dilip, B. Balusamy, and R. Patan, ‘‘Ant colony optimization based quality of service aware energy balancing secure routing algorithm for wireless sensor networks,’’ *IEEE Trans. Eng. Manag.*, vol. 68, no. 1, pp. 170–182, Feb. 2021.

[10] M. Mathapati, T. S. Kumaran, A. Muruganandham, and M. Mathivanan, ‘‘Secure routing scheme with multi-dimensional trust evaluation for wireless sensor network,’’ *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 6, pp. 6047–6055, Jun. 2021.

[11] W. Fang, W. Zhang, W. Chen, Y. Liu, and C. Tang, ‘‘TMSRS: Trust management-based secure routing scheme in industrial wireless sensor network with fog computing,’’ *Wireless Netw.*, vol. 26, no. 5, pp. 3169–3182, Sep. 2020.

[12] L. Wei, Y. Qing, and Y. Nan, ‘‘A trust-based secure routing algorithm for wireless sensor networks,’’ in *Proc. 34th Chin. Control Conf. (CCC)*, Jul. 2015, pp. 7726–7729, doi: 10.1109/ChiCC.2015.7260866.

[13] A. Ahmed, K. A. Bakar, M. I. Channa, and A. W. Khan, ‘‘A secure routing protocol with trust and energy awareness for wireless sensor network,’’ *Mobile Netw. Appl.*, vol. 21, no. 2, pp. 272–285, 2016.

[14] K. A. Awan, I. Din, A. Almogren, M. Guizani, A. Altameem, and S. U. Jadoon, ‘‘RobustTrust—A pro-privacy robust distributed trust management mechanism for Internet of Things,’’ *IEEE Access*, vol. 7, pp. 62095–62106, 2019.

[15] W. Fang, C. Zhu, W. Chen, W. Zhang, and J. J. P. C. Rodrigues, ‘‘BDTMS: Binomial distribution-based trust management scheme for healthcare-oriented wireless sensor network,’’ in *Proc. 4th Int. Wireless Commun. Mobile Comput. Conf.*, Jun. 2018, pp. 382–387, doi: 10.1109/IWCMC.2018.8450403.

[16] R. W. Anwar, A. Zainal, F. Outay, A. Yasar, and S. Iqbal, ‘‘BTEM: Belief based trust evaluation mechanism for wireless sensor networks,’’ *Future Gener. Comput. Syst.*, vol. 96, pp. 605–616, Jul. 2019.

[17] M. Zhang, ‘‘Trust computation model based on improved Bayesian for wireless sensor networks,’’ in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2017, pp. 960–964, doi: 10.1109/ICCT.2017.8359777.

[18] A. B. Feroz Khan and G. Anandharaj, ‘‘A cognitive energy efficient and trusted routing model for the security of wireless sensor networks: CEMT,’’ *Wireless Pers. Commun.*, vol. 119, no. 4, pp. 3149–3159, Apr. 2021.

[19] Y. Hu, Y. Wu, and H. Wang, ‘‘Detection of insider selective forwarding attack based on monitor node and trust mechanism in WSN,’’ *Wireless Sensor Netw.*, vol. 6, no. 11, pp. 237–248, 2014.

[20] D. Qin, S. Yang, S. Jia, Y. Zhang, J. Ma, and Q. Ding, ‘‘Research on trust sensing based secure routing mechanism for wireless sensor network,’’ *IEEE Access*, vol. 5, pp. 9599–9609, 2017.

[21] A. Beheshtiasl and A. Ghaffari, ‘‘Secure and trust-aware routing scheme in wireless sensor networks,’’ *Wireless Pers. Commun.*, vol. 107, no. 4, pp. 1799–1814, Apr. 2019.

[22] X. Yu, F. Li, T. Li, N. Wu, H. Wang, and H. Zhou, ‘‘Trust-based secure directed diffusion routing protocol in WSN,’’ *J. Ambient Intell. Hum. Comput.*, pp. 1–13, Nov. 2020, doi: 10.1007/s12652-020-02638-z.

[23] P. A. Patil, R. S. Deshpande, and P. B. Mane, ‘‘Trust and opportunity based routing framework in wireless sensor network using hybrid optimization algorithm,’’ *Wireless Pers. Commun.*, vol. 115, no. 1, pp. 415–437, Jun. 2020.

[24] T. Kalidoss, L. Rajasekaran, K. Kanagasabai, G. Sannasi, and A. Kannan, ‘‘QoS aware trust based routing algorithm for wireless sensor networks,’’ *Wireless Pers. Commun.*, vol. 110, no. 4, pp. 1637–1658, Feb. 2020.

[25] M. Hajiee, M. Fartash, and N. Osati Eraghi, ‘‘An energy-aware trust and opportunity based routing algorithm in wireless sensor networks using multipath routes technique,’’ *Neural Process. Lett.*, vol. 53, no. 4, pp. 2829–2852, Aug. 2021.

[26] J. Jasper, ‘‘A secure routing scheme to mitigate attack in wireless adhoc sensor network,’’ *Comput. Secur.*, vol. 103, Apr. 2021, Art. no. 102197.

[27] W. Fang, W. Zhang, W. Chen, J. Liu, Y. Ni, and Y. Yang, ‘‘MSCR: Multidimensional secure clustered routing scheme in hierarchical wireless sensor networks,’’ *EURASIP J. Wireless Commun. Netw.*, vol. 2021, no. 1, pp. 1–20, Jan. 2021.

[28] Q. Zhang, X. Liu, J. Yu, and X. Qi, ‘‘A trust-based dynamic slicing mechanism for wireless sensor networks,’’ *Proc. Comput. Sci.*, vol. 174, pp. 572–577, Oct. 2020.

[29] T. Yang, X. Xiangyang, L. Peng, L. Tonghui, and P. Leina, ‘‘A secure routing of wireless sensor networks based on trust evaluation model,’’ *Proc. Comput. Sci.*, vol. 131, pp. 1156–1163, Oct. 2018.

[30] Q. Shi, L. Qin, Y. Ding, B. Xie, J. Zheng, and L. Song, ‘‘Information-aware secure routing in wireless sensor networks,’’ *Sensors*, vol. 20, no. 1, p. 165, Dec. 2019.

[31] N. Sun and Y. Lu, ‘‘A self-adaptive genetic algorithm with improved mutation mode based on measurement of population diversity,’’ *Neural Comput. Appl.*, vol. 31, no. 5, pp. 1435–1443, May 2019.

[32] J. Xu, L. Pei, and R.-Z. Zhu, ‘‘Application of a genetic algorithm with random crossover and dynamic mutation on the travelling salesman problem,’’ *Proc. Comput. Sci.*, vol. 131, pp. 937–945, May 2018.

[33] X. Guoxin, T. Xin, P. Wei, and R. Tao, ‘‘Clustering analysis based on chaos micro variation adaptive genetic algorithm for radio fuze jamming,’’ in *Proc. 29th Chin. Control Decis. Conf. (CCDC)*, May 2017, pp. 616–620, doi: 10.1109/CCDC.2017.7978287.

[34] H. Hu, Y. Han, M. Yao, and S. Xue, ‘‘Trust based secure and energy efficient routing protocol for wireless sensor networks,’’ *IEEE Access*, early access, Apr. 27, 2021, doi: 10.1109/ACCESS.2021.3075959.

[35] C. Wang, X. Liu, and H. Hu, ‘‘Energy-efficient and load-balanced clustering routing protocol for wireless sensor networks using a chaotic genetic algorithm,’’ *IEEE Access*, vol. 8, pp. 158082–158096, 2020.

[36] S. A. Sert, E. Onur, and A. Yazici, ‘‘Security attacks and countermeasures in surveillance wireless sensor networks,’’ in *Proc. 9th Int. Conf. Appl. Inf. Commun. Technol. (AICT)*, Oct. 2015, pp. 201–205.

[37] S. A. Sert, C. Fung, R. George, and A. Yazici, ‘‘An efficient fuzzy path selection approach to mitigate selective forwarding attacks in wireless sensor networks,’’ in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6.

**YOUJIA HAN** received the B.Eng. degree from the Anhui Wenda University of Information Engineering, Hefei, China, in 2018. He is currently pursuing the M.S. degree with the Changchun University of Technology, Changchun, China. His main research interest includes wireless sensor network security.

**HUANGSHUI HU** received the B.Eng. degree in computer application and the M.S. and Ph.D. degrees from the Changchun University of Science and Technology (now Jilin University), Changchun, China, in 1999, 2005, and 2012, respectively. From 2005 to 2008, he worked as a Research and Development Manager at Changchun Lianxin Technology Company Ltd., Changchun. From 2008 to 2013, he worked as the Technical Director of Jilin Omnidirectional Technology Company Ltd., Changchun. Since 2015, he has been with the Department of Computer Science and Engineering, Changchun University of Technology, Changchun, where he is currently a Professor. His main research interests include topology control in wireless sensor networks and multifunction vehicle bus networks.

**YUXIN GUO** graduated from Dalian Jiaotong University, Dalian, China, in 2020. She is currently pursuing the master's degree with the Changchun University of Technology, Changchun, China. Her main research interest includes routing in wireless sensor networks.