Hindawi
Security and Communication Networks
Volume 2019, Article ID 2545129, 12 pages
https://doi.org/10.1155/2019/2545129

Research Article

A Novel Trust Model Based on Node Recovery Technique for WSN

Ping Qi,[1,2] **Fucheng Wang**,[1] **and Shu Hong**[1]

[1]*Department of Mathematics and Computer Science, Tongling University, Tongling 244061, China*
[2]*The Institute of Server Computing, Tongling University, Tongling 244061, China*

Correspondence should be addressed to Shu Hong; shuhong_7812@163.com

Received 15 April 2019; Revised 16 July 2019; Accepted 14 August 2019; Published 3 September 2019

Academic Editor: Petros Nicopolitidis

With the rapid development of sensor technology and wireless network technology, wireless sensor network (WSN) has been widely applied in many resource-constrained environments and application scenarios. As there are a large number of sensor nodes in WSN, node failures are inevitable and have a significant impact on task execution. In this paper, considering the vulnerability, unreliability, and dynamic characteristics of sensor nodes, node failures are classified into two categories including unrecoverable failures and recoverable failures. Then, the traditional description of the interaction results is extended to the trinomial distribution. According to the Bayesian cognitive model, the global trust degree is aggregated by both direct and indirect interaction records, and a novel trust model based on node recovery technique for WSNs is proposed to reduce the probability of failure for task execution. Simulation results show that compared with existing trust models, our proposed TMBNRT (trust model based on node recovery technique) algorithm can effectively meet the security and the reliability requirements of WSN.

1. Introduction

Wireless sensor network (WSN) often consists of thousands of spatially dispersed and dedicated sensors which are small-sized and resource-constrained [1]. With the rapid development of sensor technology and wireless network technology, WSN becomes one of the most useful technologies which has been widely applied in many resource-constrained environments and application scenarios such as military, medical care, agriculture, and environmental monitoring [2]. However, sensor nodes are vulnerable to be captured or attacked due to the open and unattended environment. Once a sensor node is captured and if this adversary sensor node is regarded as a normal one, it could destroy the whole network. Therefore, the security problem of the wireless sensor network is vital which needs to be addressed to guarantee proper operation [3].

In the past few years, many research works have been produced on wireless network security. The main security measures include identity authentication, encryption algorithm, data integrity, and intrusion detection. Although these security mechanisms can protect sensor nodes from external invasion, they have little effect on resisting internal attacks with the information of authorized nodes [4]. Furthermore, due to the requirement of high memory usage and power consumption in processing and communication, security mechanisms mentioned above are not suitable for capability-constraint and energy-limited sensor node [5]. Therefore, how to provide an effective and lightweight security mechanism becomes a major challenge.

Since the concept of trust management was first put forward by Blaze et al. [6], it has been widely used as an additional means for security [6]. To the best of our knowledge, the research on how to apply the trust mechanism to resource distribution can be traced back to resource allocation in a grid resource scheduling system by Dogan and Ozguner [7]. They define the "trust" in grid environment as the evaluation of the service provided by the target node, including direct trust (observation of the target node's historic behaviors) and indirect trust (recommendation from other nodes). The trust evaluation algorithm calculates each associated node' trust degree based on its historic behaviors. Since then, most researches focus on solving the security issues with the aid of different trust evaluation methods. For instance, Wang et al. [8] propose a Bayesian cognitive model in Cloud; they use both direct trust value

and indirect trust value to calculate the node's integrated trust. Che et al. [9] present the LTMBE algorithm in WSN. LTMBE algorithm uses direct trust value alone when the confidence coefficient is above the certain threshold in order to meet the resource limitations of tiny sensor nodes in WSNs.

Although some progress has been achieved in this area, obvious limitations still remain as the classification of node failures is not detailed enough. The traditional trust model simply describes the interaction results between two sensor nodes by binomial events (successful interaction and failure interaction). However, sensor nodes are limited with its power and range of transmission, and node failures may occur due to insufficient battery power, environmental factors, data transmission failure, and so on. Therefore, node failures are caused not only by internal attacks, but also by hardware and communication problems resulting in interaction failure. The difference is that the latter can be partially recovered by the node recovery technique while the former cannot.

For example, let x, y, and z be three sensor nodes in WSN, and their interaction results are described by binomial events. Let the interactions between x and y, and x and z be independent, and the number of interactions between them be both 100 times, in which the failure interactions are both 10 times. Obviously, the trust degree of y and z calculated by the traditional trust model is the same. However, when we consider the recoverability of some failures, the interaction failures in WSN should be classified into two categories: unrecoverable failure and recoverable failure. Therefore, as in the example above, assume that the recoverable failures between x and y, x and z are 1 time and 9 times, respectively, and sensor node z is supposed to be more reliable. The main reason is that the interaction failures between x and z are more likely to be recovered by the node recovery technique, and hence, the task completion rate can be improved.

In fact, as one of the failure tolerance techniques, node recovery technique is used to recover the failure of the most used sensor nodes in WSN and has been proven to be a very promising technique with its demonstrated capability [10, 11]. Unfortunately, the existing trust management for WSN is designed without paying much attention to the recoverability of sensor nodes and the node recovery technique. Specifically, binomial events cannot sufficiently describe the interaction results or distinguish conventional failures from malicious attack. In this paper, from the security and reliability points of view, we propose and implement a novel trust model based on the node recovery technique in WSN by extending the traditional binomial description to the trinomial distribution for the interaction results (viz., successful interaction, unrecoverable failures, and recoverable failures) which is more accurate for describing the interaction results between different sensor nodes.

The remainder of this paper is organized as follows: Section 2 presents the related work. Section 3 proposes the novel algorithm and model. Section 4 demonstrates the experimental results. Finally, Section 5 concludes this paper.

2. Related Work

In this section, we summarize some noteworthy trust management methods. Trust management methods are required to optimize the selection of trustworthy sensor nodes. According to different theoretical frameworks, various trust models have been proposed as follows: D-S evidence trust model [12], entropy trust model [13], fuzzy logic trust model [14, 15], Bayesian methodology-based trust model [9, 16, 17], Bayesian cognitive trust model [8], hierarchical trust model [18], and hybrid trust model [19]. By reviewing the trust models mentioned above, we can draw the following conclusions: (1) In order to meet the resource limitations, the trust models for WSN should be lightweight; it is unrealistic to implement a complex trust evaluation on tiny sensor nodes; (2) the evaluation of sensor nodes' trust behavior aroused wide attention of researchers in recent years. The trust degree of a sensor node is always decided by its interaction history and other sensor nodes' recommendation.

Trust is the key influence factor of relationships in social networks. Individuals in social networks and sensor nodes in WSN share great similarities [20]. Therefore, it is quite natural to evaluate each node's reliability based on both direct and recommendation relationship. Bayesian theory and Beta reputation system are widely used because of their accuracy of describing interaction behavior and low computational complexity. Ganeriwal et al. [21] propose a reputation-based framework for sensor networks. In this model, the interactions between two nodes are described by Beta function, and sensor nodes' trust values are evaluated by mathematical expectation. Zeng et al. [22, 23] apply the Bayesian cognitive model to resource scheduling in Cloud and MANETs, respectively. A trust dynamic level scheduling algorithm is proposed to quantify the trustworthiness and decrease the failure. Feng et al. [16] present a credible Bayesian-based trust management (BTMS) in WSN. By integrating the sliding time window and punishment mechanism, a modified Bayesian equation is used to calculate and adjust trust value, which makes it more lightweight and accurate.

In the above studies, the trust value is defined as the evaluation of the target node's ability to provide service as shown by its historical interactions with other nodes. The historical interactions include successful interactions and failure interactions, described by binomial events. However, none of these algorithms take the recoverability of failure interactions and node recovery technique into account.

Node recovery is a technique used to recover the failure of used sensor nodes in WSN [24]. Many studies and applications which are related to energy saving and reliability evaluation are based on the node recovery technique. Abbasi et al. [25] propose a least-disruptive topology repair (LeDiR) algorithm. LeDiR algorithm autonomously repositions a subset of the sensor nodes to restore connectivity with minimal topology changes. Rafiei et al. [26] devise a node relocation algorithm to recover large-scale coverage holes. The effects of the number of participating sensor nodes and movement iterations are examined in Voronoi-based node

relocation algorithms. Shih et al. [27] present a genetic algorithm- (GA-) based fault node recovery policy to reduce the replacements of sensor nodes (main influence factors include the number of nodes moved and the total distance moved) and the rate of data loss.

Through the review of these related works, we can conclude that the main idea of the fault recovery algorithm is to resume the failed nodes or reposition some of the healthy nodes to maintain connectivity, or to reduce the rate of energy consumption by optimizing the scheduling policy. However, the research on how to apply the node recovery technique to the trust model is very limited.

3. A Novel Trust Model Based on the Node Recovery Technique

A social network is a social structure composed of a set of social actors such as individuals or organizations. The basic idea of the trust relationship model of sociology is that the individual in a social network reflects the characteristics of its behavior when it cooperates with the others, and "trust" is the evaluation of certain entities' reliable behaviors [28]. The Bayesian cognitive model for probabilistic inference provides a general approach to understand how problems of induction can be solved in principle. It is a rapidly growing approach for cognitive science and has been extended and applied widely. Obviously, the relationship between sensor nodes in WSN fits well with the trust relationship model of sociology.

The cognitive trust model for WSN is shown in Figure 1. In this trust model, the current node calculates the trust degree of its neighbor nodes and chooses the cooperation sensor node by finding the highest trust value for each interaction. According to the trust model discussed above, the trust value of the sensor node reflects the reliability of the service it supplies. Let node i and node j be two sensor nodes in WSN, when there are n times interactions between two nodes, define trust degree as the probability of successful interaction at $n+1$ times.

There are two different kinds of trust degree: direct trust degree and indirect trust degree. As shown in Figure 1(a), there are direct interactions between node i and node j. Then, we can calculate the direct probability of successful interactions, which is called direct trust degree, denoted by θ_{dt}. However, when the sensor node has no interaction or very few interactions with the target node, the number of total interactions is not enough to perform trust evaluation because of its low confidence coefficient. As shown in Figure 1(b), the intermediate nodes are required to collect the indirect interactions between two nodes. Therefore, we can obtain an indirect probability of successful interaction, which is called indirect trust degree, denoted by θ_{rt}. Then, the global trust degree (denoted by θ) is aggregated by direct trust degree and indirect trust degree according to the combination function defined as follows:

$$\theta = f\left(\lambda \cdot \theta_{\mathrm{dt}} + (1-\lambda)\cdot \theta_{\mathrm{rt}}\right) \le \lambda \cdot f\left(\theta_{\mathrm{dt}}\right) + (1-\lambda)\cdot f\left(\theta_{\mathrm{rt}}\right), \quad \lambda \in (0,1), \tag{1}$$

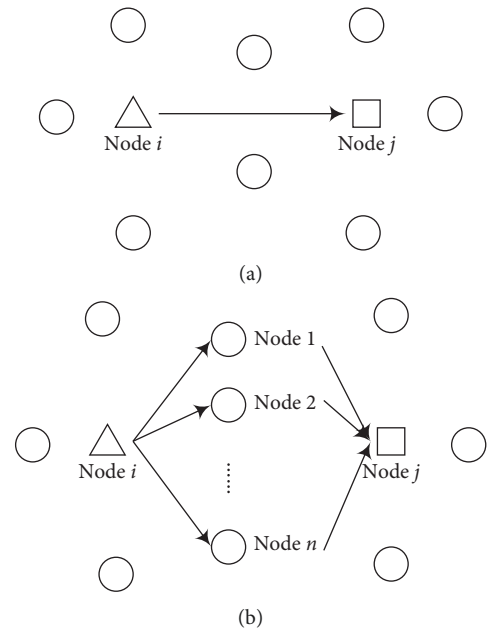

Figure 1: The cognitive trust model: (a) direct interaction and (b) indirect interaction.

where $f(\cdot)$ is a combination function, which satisfies the property of convex function. For instance, we can use linear function as combination function: $\theta = \lambda \cdot \theta_{\mathrm{dt}} + (1-\lambda)\cdot \theta_{\mathrm{rt}}$. The influence factor λ is used to reveal the different influences of direct trust degree and indirect trust degree. $\lambda > 0.5$ means the influence of the direct trust degree is greater than the indirect trust degree.

We make the following assumptions to describe WSN: (1) All sensor nodes are deployed in a two-dimensional space. The sensor nodes have identical primary energy, communication range, computing capability, communication capability, and storage capability; (2) the sensor node can only communicate with its neighbor nodes, and neighbor nodes are defined as the nodes within the communication range. During the preparation phase, the wireless sensor network is formed by broadcast communication, and then the list of neighbor nodes and network topology are established; (3) the wireless sensor network is modeled by an undirected weighted graph $G = (V, E)$, where V is a set of nodes, and E is a set of edges. After each deployment, the sensor node is neither added nor removed; (4) when a request arrives at the sensor node, the sensor node responds to it immediately; and (5) the failures in different sensor nodes are independent.

3.1. Time Decay Factor. In our trust model, a sensor node continuously monitors its neighbor nodes' behavior, and then the node uses these interaction results to evaluate the trust value of other nodes. The objective of the Watchdog mechanism [21] is extended to detect the presence of invalid data resulting from faulty nodes and failure recovery mechanism.

Suppose n times interactions occur between node i and node j, there are α times unrecoverable failure interactions, β times recoverable failure interactions, and γ times successful interactions. Then we use the distribution function to predict the probability in the future. However, the trust value of sensor nodes is also affected by time. It is intuitive that recent interactions have more influence on trust decision. The impact of interactions decreases constantly as the time passes on. When the time increases to a certain threshold, these ancient interactions lose their reference value and should be discarded. Therefore, considering the dynamic of nodes' interaction and the energy consumption of the calculation process, we investigate the interaction between two nodes for a period of time. The concept of time segment is used to reflect the change of the sensor node's trust value with time, which can be a minute or an hour. As shown in Figure 2, the latest effective history records are composed of several recent time segments.

To simplify the calculation, we define "time segment" as 50 seconds in this paper. For the k-th time segment in the latest effective history records, let the number of unrecoverable failure interactions be α_k, the number of recoverable failure interactions be β_k, and the number of successful interactions be γ_k. Decay factor μ is used to dynamically set weights of the corresponding time segment. The history records with decay factor can be defined as follows:

$$\begin{cases} \alpha_k^{new} = \alpha_k * \mu^{m-k}, \\ \beta_k^{new} = \beta_k * \mu^{m-k}, \\ \gamma_k^{new} = \gamma_k * \mu^{m-k}, \\ 0 \le \mu \le 1, \end{cases} \tag{2}$$

where m is the maximum number of time segments and μ is the decay factor and $\mu = 1$ indicates the impact of time factor is ignored.

3.2. Direct Trust Degree. As mentioned above, hardware failures and communication failures can partly be recovered by the node recovery technique. Let the probability of unrecoverable and recoverable failure interaction of the k-th time segment be q_k^{UF} and q_k^{RF}, then the probability of successful interaction is $\theta_k = 1 - q_k^{UF} - q_k^{RF}$.

Plenty of research on failure rule shows that the failure of electronic equipment has strong temporal locality and spatial locality [29]. When the time between failures is regarded as a random process, this random process can be considered to obey *Weibull* distribution (*scale*, *shape*), where $scale > 0$ is the scale parameter and $shape > 0$ is the shape parameter.

The probability density function $pdf(t; scale, shape)$ and cumulative distribution function $cdf(t; scale, shape)$ can be calculated as follows:

$$\begin{cases} pdf(t; scale, shape) = \dfrac{shape}{scale}\left(\dfrac{t}{scale}\right)^{shape-1} e^{-(t/scale)^{shape}}, \\ cdf(t; scale, shape) = 1 - e^{-(t/scale)^{shape}}. \end{cases} \tag{3}$$

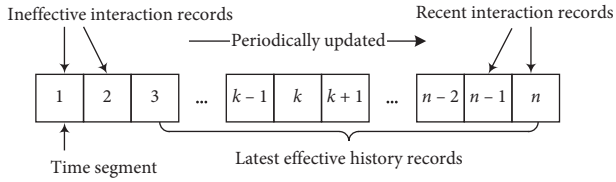

Figure 2: Time segment.

Then, calculate the failure rate $h(t; scale, shape)$ based on *Weibull* distribution function:

$$h(t; scale, shape) = \frac{pdf(t)}{1 - cdf(t)} = \frac{shape}{scale}\left(\frac{t}{scale}\right)^{shape-1}. \tag{4}$$

The probability of recoverable failure interaction of the k-th time segment is given by

$$q_k^{RF} = \frac{shape}{scale}\left(\frac{t}{scale}\right)^{shape-1}, \quad t \in k\text{−th time segment}. \tag{5}$$

Let i and j be two nodes in WSN, the direct interactions between two nodes are described by trinomial distribution (successful/unrecoverable failure/recoverable failure). When the number of time segments is m, suppose $\theta = (\theta_1, \theta_2, \ldots, \theta_m)$ be random variables, and its prior distribution is uniformly distributed. The prior joint probability density function can be calculated as follows:

$$f(\theta_1, \ldots, \theta_m) = \begin{cases} \dfrac{1}{V_m}, & (\theta_1, \ldots, \theta_m) \in G_m, \\ 0, & (\theta_1, \ldots, \theta_m) \notin G_m, \end{cases} \tag{6}$$

where G_m is a point set in m dimension Euclidean space and V_m is a Lebesgue measure of G_m. In order to estimate the expectation value of the m-th time segment, the posteriori joint probability density function and the posteriori marginal probability density function should be calculated.

Suppose sample set $X = \{X_{11}, X_{12}, X_{13}; \ldots; X_{m1}, X_{m2}, X_{m3}\}$, where X_{k1}, X_{k2}, X_{k3} represent the number of unrecoverable failure, recoverable failure, and successful interaction of the k-th time segment separately. Let x_0 be a sample observation of sample set X, $x_0 = \{\alpha_1, \beta_1, \gamma_1; \ldots; \alpha_m, \beta_m, \gamma_m\}$.

Suppose $f(\theta \mid x_0)$ is the posteriori joint probability density function of θ, $p(\theta_m \mid x_0)$ is the posteriori marginal probability density function of θ_m, and $\widehat{\theta}_m$ is the Bayesian estimation of θ_m. According to the Bayesian theory and formula (6), the posteriori joint probability density function $f(\theta \mid x_0)$ can be calculated as follows:

$$f(\theta \mid x_0) = \frac{P\{X = x_0 \mid \theta\}}{\int_{G_m} P\{X = x_0 \mid \theta\} d\theta}. \tag{7}$$

On the basis of the above analysis and evaluation, the following formula can be derived:

$$P\{X = x_0 \mid \theta\} = \prod_{k=1}^{m} P\{X_{k1} = \alpha_k, X_{k2} = \beta_k, X_{k3} = \gamma_k \mid \theta\}$$
$$= \prod_{t=1}^{m} \frac{(\alpha_k + \beta_k + \gamma_k)}{\alpha_k! \cdot \beta_k! \cdot \gamma_k!} \left(q_k^{\mathrm{UF}}\right)^{\alpha_k} \left(q_k^{\mathrm{RF}}\right)^{\beta_k} (\theta_k)^{\gamma_k}. \tag{8}$$

According to formula (5), the probability of recoverable failure interaction of the k-th time segment can be calculated by failure rate, and the probability of unrecoverable failure interaction can be represented as a calculated expression:

$$q_k^{\mathrm{UF}} = 1 - q_k^{\mathrm{RF}} - \theta_k = 1 - \frac{shape}{scale}\left(\frac{t}{scale}\right)^{shape-1} - \theta_k. \tag{9}$$

According to formulas (7) and (8), the posteriori joint probability density function can be calculated as follows:

$$f(\theta \mid x_0) = \frac{\prod_{t=1}^{m} \left(1 - q_k^{\mathrm{RF}} - \theta_k\right)^{\alpha_k} \theta_i^{\gamma_k}}{\int_{G_m} \prod_{t=1}^{m} \left(1 - q_k^{\mathrm{RF}} - \theta_k\right)^{\alpha_k} \theta_k^{\gamma_k} d\theta}, \quad \theta \in G_m. \tag{10}$$

Identity relation (11) is introduced to compute the denominator in formula (10):

$$\int_0^y b(x \mid a, u, v)dx = \int_0^y \frac{x^{u-1}(1-a-x)^{v-1}}{B(u,v)} dx$$
$$= \sum_{i=u}^{u+v-1} \binom{u+v-1}{i} y^i (1-a-y)^{u+v-1-i}, \tag{11}$$

where u and v are positive integers, $B(u, v)$ is the Beta function, and $0 \le a < 1$. Then, the denominator in formula (10) can be calculated as follows:

$$\int_{G_m} \prod_{t=1}^{m} \left(1 - q_k^{\mathrm{RF}} - \theta_k\right)^{\alpha_k} \theta_k^{\gamma_k} d\theta$$
$$= \int_0^{1-q_k^{\mathrm{RF}}} d\theta_m \int_0^{\theta_m} d\theta_{m-1} \cdots \int_0^{\theta_3} d\theta_2 \int_0^{\theta_2} \prod_{i=1}^{m} \left(1 - q_k^{\mathrm{RF}} - \theta_k\right)^{\alpha_k} \theta_k^{\gamma_k} d\theta_1. \tag{12}$$

Substituting formulas (11) and (12) into formula (10), the posteriori joint probability density function is calculated by the following formula:

$$f(\theta \mid x_0) = \frac{\prod_{t=1}^{m} \left(1 - q_k^{\mathrm{RF}} - \theta_k\right)^{\alpha_k} \theta_k^{\gamma_k}}{\left(1 - q_k^{\mathrm{RF}}\right)^{g_m} \sum_{h_1}^{g_1} \sum_{h_2}^{g_2} \cdots \sum_{h_{m-1}}^{g_{m-1}} W(h_1, h_2, \ldots, h_{m-1})}, \quad \theta \in G_m. \tag{13}$$

In (13), $W(h_1, h_2, \ldots, h_{m-1})$ is an intermediate result and can be calculated as follows:

$$\begin{cases} W(h_1, h_2, \ldots, h_{m-1}) = \prod_{j=1}^{m-1} d_j \\ \quad \cdot B(s_m + h_{m-1}, g_m - s_m - h_{m-1} + 1); \\ d_j = \binom{g_j}{h_j} B(s_j + h_{j-1}, g_j - s_j - h_{j-1} + 1), & j = 1, 2, \ldots, m; \\ h_0 = 0,\ h_{k-1} = s_{k-1} + h_{k-2}, & k = 1, 2, \ldots, m; \\ g_k = \sum_{j=1}^{k} (\alpha_k + \gamma_k) + k, & k = 1, 2, \ldots, m; \\ s_k = \gamma_k + 1, & k = 1, 2, \ldots, m. \end{cases} \tag{14}$$

The posteriori marginal probability density function of θ_m is calculated by the following formula:

$$p(\theta_m \mid x_0) = \int_0^{\theta_m} d\theta_{m-1} \cdots \int_0^{\theta_3} d\theta_2 \int_0^{\theta_2} f(\theta \mid x_0) d\theta_1$$
$$= \frac{\theta_m^{\,s_m + h_{m-1} - 1} \left(1 - q_m^{\mathrm{RF}} - \theta_m\right)^{g_m - s_m - h_{m-1}}}{B(s_m + h_{m-1}, g_m - s_m - h_{m-1} - 1)}. \tag{15}$$

Let the loss function be quadratic, the direct trust degree is calculated by the Bayesian estimation of θ_m as follows:

$$\theta_{\mathrm{dt}} = \widehat{\theta}_m = E(\theta_m \mid x_0) = \int_0^{1-q_m^{\mathrm{RF}}} \theta_m p(\theta_m \mid x_0) d\theta_m$$
$$= \frac{\left(1 - q_m^{\mathrm{RF}}\right) \sum_{h_1}^{g_1} \sum_{h_2}^{g_2} \cdots \sum_{h_{m-1}}^{g_{m-1}} W(h_1, h_2, \ldots, h_{m-1}) \cdot \left(s_m + h_{m-1}/g_m + 1\right)}{\sum_{h_1}^{g_1} \sum_{h_2}^{g_2} \cdots \sum_{h_{m-1}}^{g_{m-1}} W(h_1, h_2, \ldots, h_{m-1})}. \tag{16}$$

When the recoverable failure and unrecoverable failure are simply regarded as interaction failure, the above formulas (15) and (16) are consistent with Benoit probabilistic model.

In order to save nodes' memory, we use recent interaction records instead of all interaction records to evaluate the probability of successful service provision of the

target node. Then, to reduce the energy consumption, only the direct interaction records are used to calculate the ability of reliable service.

However, the accuracy of the direct trust degree is related to the number of total interactions. When there are few interactions or even no interactions between two nodes, it is not suitable to exclusively use the direct interaction records for evaluating the trustworthiness. The method of increasing the number of interaction records is to collect the indirect interactions by asking its neighbor nodes. Nonetheless, searching means more energy consumption. In order to obtain a trade-off between accuracy and energy consumption, the searching will be stopped when the number of samples is sufficient. Considering the calculability of the probability of recoverable failure interactions, we approximately evaluate the confidence interval of the direct trust degree by Binomial proportion confidence interval.

Let $(\widehat{\theta}_{dt} - \delta, \widehat{\theta}_{dt} + \delta)$ be the confidence interval, where δ is the error level. According to the linear confidence interval theory [28], confidence level γ can be obtained as follows:

$$\begin{aligned}\gamma &= P\left(\widehat{\theta}_{\mathrm{dt}} - \delta < \widehat{\theta}_{\mathrm{dt}} < \widehat{\theta}_{dt} + \delta\right) \\ &= \frac{\Gamma\left(\sum_{k=1}^{m}(\alpha_k + \beta_k)\right)\Gamma\left(\sum_{k=1}^{m}\gamma_k\right)}{\Gamma\left[\sum_{k=1}^{m}(\alpha_k + \beta_k) + \sum_{k=1}^{m}\gamma_k\right]} \\ &\cdot \int_{\widehat{\theta}_{\mathrm{dt}}-\delta}^{\widehat{\theta}_{\mathrm{dt}}+\delta} \theta^{\sum_{k=1}^{m}(\alpha_k+\beta_k)-1}(1-\theta)^{\sum_{k=1}^{m}\gamma_k-1}d\theta.\end{aligned} \tag{17}$$

The confidence level and accuracy of interval estimation are two trade-off factors, which cannot be increased together. Therefore, let γ_0 be a threshold of confidence level. Then we improve the accuracy by increasing the number of samples. When the accuracy is at an acceptable level, the direct trust degree can be evaluated. If not, the indirect interaction records are used to obtain an indirect probability of successful cooperation between two nodes. The relationship between n_0, δ, and γ_0 can be expressed by formula (18), where n is the number of interaction records:

$$n \geq -\frac{1}{2\delta^2}\ln\left(\frac{1-\gamma_0}{2}\right). \tag{18}$$

3.3. Indirect Trust Degree. When the direct interaction records are not enough to perform trust evaluation, intermediate nodes are used to provide indirect interaction records. In fact, the habit of human cognition shows that people always trust those with high credibility. Therefore, the evaluation of trustworthiness of intermediate nodes is necessary. Then we search the indirect interaction records according to the trust degree of these nodes.

Suppose there are three sensor nodes: node i, node j, and node x, the interactions between i and x, and j and x are independent with the same gamma distribution. Let the numbers of unrecoverable failure interactions, recoverable failure interactions, and successful interactions of the k-th time segment between i and x be α_{k1}, β_{k1}, and γ_{k1}, respectively, let the numbers of unrecoverable failure interactions, recoverable failure interactions, and successful interactions of the k-th time segment between j and x be α_{k2}, β_{k2}, and γ_{k2}, respectively. The interaction records with decay factor can be calculated by formula (2).

Suppose a sample observation $x_0' = \{\alpha_{k1} + \alpha_{k2}, \beta_{k1} + \beta_{k2}, \gamma_{k1} + \gamma_{k2}; \ldots; \alpha_{m1} + \alpha_{m2}, \beta_{m1} + \beta_{m2}, \gamma_{m1} + \gamma_{m1}\}$. Let the loss function be quadratic, the indirect trust degree is calculated by the Bayesian estimation of θ_m' as follows:

$$\begin{cases} s_k = \gamma_{k1} + \gamma_{k2} + 1, & k = 1, 2, \ldots, m, \\ g_k = \sum_{j=1}^{k}(\alpha_k + \gamma_k) + k, & k = 1, 2, \ldots, m, \\ h_0 = 0,\ h_{k-1} = s_{k-1} + h_{k-2}, & k = 1, 2, \ldots, m, \\ d_j = \begin{pmatrix} g_j \\ h_j \end{pmatrix} B\left(s_j + h_{j-1}, g_j - s_j - h_{j-1} + 1\right), & j = 1, 2, \ldots, m-1, \\ W(h_1, h_2, \ldots, h_{m-1}) = \prod_{j=1}^{m-1} d_j \cdot B\left(s_m + h_{m-1}, g_m - s_m - h_{m-1} + 1\right), & \\ \theta_{\mathrm{rt}} = \widehat{\theta}_m' = E(\theta_m' \mid x_0') = \int_0^{1-q_m^{\mathrm{RF}}} \theta_m' p(\theta_m' \mid x_0') d\theta_m', & \\ \quad = \frac{(1-q_m^{\mathrm{RF}})\sum_{h_1}^{g_1}\sum_{h_2}^{g_2}\cdots\sum_{h_{m-1}}^{g_{m-1}} W(h_1, h_2, \ldots, h_{m-1}) \cdot (s_m + h_{m-1}/g_m + 1)}{\sum_{h_1}^{g_1}\sum_{h_2}^{g_2}\cdots\sum_{h_{m-1}}^{g_{m-1}} W(h_1, h_2, \ldots, h_{m-1})}. & \end{cases} \tag{19}$$

3.4. Global Trust Degree. Direct trust degree and indirect trust degree can be aggregated into a global trust degree by entropy theory. The basic concept of entropy theory is to measure the average rate at which information is provided by a stochastic source of data. The entropy of a random variable X is defined by

$$H(X) = -\sum_{i=1}^{n} P(x_i)\log_b P(x_i). \tag{20}$$

According to formula (20), the entropy of direct trust degree and indirect trust degree can be obtained by the following formula:

$$\begin{cases} H(\theta_{dt}) = -\theta_{dt}\log_2\theta_{dt} - (1-\theta_{dt})\log_2(1-\theta_{dt}), \\ H(\theta_{rt}) = -\theta_{rt}\log_2\theta_{rt} - (1-\theta_{rt})\log_2(1-\theta_{rt}). \end{cases} \tag{21}$$

The global trust degree can be calculated by formula (22) according to the influence factor λ in formula (23):

$$\begin{cases} \theta = \lambda\cdot\theta_{dt} + (1-\lambda)\cdot\theta_{rt}, & \text{else}, \\ \theta = \theta_{dt}, & \gamma \geq \gamma_0, \end{cases} \tag{22}$$

$$\lambda = \frac{1 - (H(\theta_{dt})/\log_2\theta_{dt})}{(1 - (H(\theta_{dt})/\log_2\theta_{dt})) + (1 - (H(\theta_{rt})/\log_2\theta_{rt}))}. \tag{23}$$

3.5. Node Recovery Technique. According to the node recovery technique in WSN, the task execution process is divided into the execution procedure and the recovery procedure. During the execution procedure, the tasks are continually scheduled onto sensor nodes and then transmitted to the next sensor node. Table 1 shows several causes for common failures and the corresponding recovery methods for wireless sensor networks. During the recovery procedure, different types of recoverable failures are recovered by several recovery policies, such as retry, node repositioning, and local remeshing.

To describe the recoverability of hardware failures, a random variable X_i is defined as follows:

$$X_i = \begin{cases} 0, & \text{if the failure on sensor node } i \text{ is recoverable}, \\ 1, & \text{if the failure on sensor node } i \text{ is unrecoverable}. \end{cases} \tag{24}$$

The probability that a hardware failure on sensor node i is recoverable is denoted by x_i with $P\{X_i = 0\} = x_i$, $P\{X_i = 1\} = 1 - x_i$. Figure 3 shows an example of a node execution process with the node retry-recovery technique. The first failure occurs at t_1 and is recovered at t_2; the second failure occurs at t_3 and is recovered at t_4, and so on [30]. The sensor node will keep executing until the task is successfully completed or the task is terminated by an unrecoverable failure. In this example, all of the failures are recoverable failures, and the task is completed successfully at last.

In the meantime, with the increase of failure nodes and recovery procedures, a large number of sensor nodes are retried or repositioned. It has a great influence on the availability of the sensor nodes and results in a heavy burden on WSN. Obviously, with x_i increased, the overhead of each recovery procedure is increased as well. To prevent unreasonably large mean overhead, a constraint can be placed on the number of recovery procedures performed.

The maximum number of recovery procedures allowed to be performed during the execution of a task on node i is denoted by N_i ($N_i \geq 1$). When the recoverable failures may be continually occurred on a sensor node, once the number of recoverable failures exceeds the deadline, the fault recovery procedure will not be executed. Then the recoverable failure is converted to the unrecoverable failure. Therefore, it should be noted that not all recoverable failures can be recovered due to time constraints.

4. Simulation Results and Analysis

In order to analyze the proposed trust model, we further developed a simulation platform based on OMNeT++ (Objective Modular Network Testbed in C++) [31]. OMNeT++ provides a component-based C++ simulation library and framework primarily for building network simulators, and it can effectively demonstrate and predict the performance change of WSN.

4.1. Experimental Setting. Without losing generality, CSMA/CA protocol and LEACH protocol are adopted for MAC layer and routing layer, respectively. Each cluster has 2~8 sensor nodes, and the number of neighbor nodes is set between 2 and 8. Other experimental parameters are set as follows: Sensory data is using the sample datasets provided by Berkeley lab [32]. Sensor nodes are randomly assigned to a 1000 m × 1000 m square area. The bandwidth of sensor nodes is defined to be uniformly distributed between 10 kbps and 20 kbps. Energy consumption for transmitter and receiver is 50 nJ/bit. The initial energy and survival energy of sensor nodes are set to be 2000 mJ and 50 mJ, respectively. The task type is related to the value of CCR (communication-to-computation ratio). By using a range of CCR values, different types of applications can be accommodated. The computation-intensive applications can be modeled by assuming CCR = 0.1. In contrast, the communication-intensive applications can be modeled by assuming CCR = 10. In our simulation experiment, CCR is set to 1.

In order to satisfy experimental requirements, two types of data have been added: failure data from malicious nodes and failure data from normal nodes. For the former, two kinds of noncooperative nodes are set as 10% and 20% of the total nodes. When they execute tasks, their failure rates are set as 80% and 50%, respectively. For the latter, we investigate the system log and analyze the research results in related studies [33]. Node failure has strong characteristic of temporal locality (for example, 74.5% of node failures occurred in 20% of time segments) and spatial locality (80% of node failures occurred on 14.5% of sensor nodes). Therefore, the timespan of the normal node between failures is set to follow the *Weibull* distribution, and the shape parameter is

Table 1: Causes and recovery methods of several common wireless sensor networks.

Failure type	Failure cause	Recovery methods
Service failure	Node failure	Node repositioning and local remeshing
Incorrect manipulation	Input error	Retry and node repositioning
Timeliness	Node crash	Retry and node repositioning
Unsatisfactory	Connection failure	Retry, node repositioning, and local remeshing

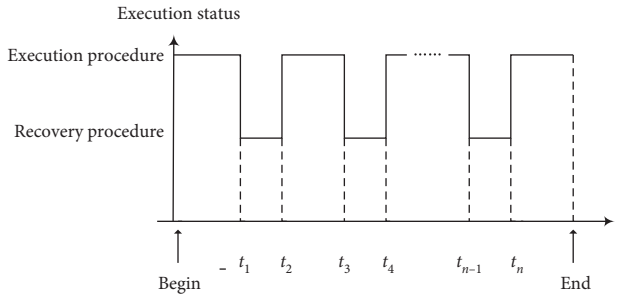

Figure 3: An example of execution procedure and recovery procedure with the node recovery technique.

set as 0.15. *zipf* distribution is used to simulate the non-uniform distribution in the space. *zipf* parameter is set as 0.15 which means 15% of sensor nodes are liable to failures. δ and γ_0 are set as 0.1 and 0.95, respectively, in formula (18).

4.2. Effect of Parameters on the Trust Model. In this experiment, we discuss the effect of parameters on the proposed trust model and mainly focus on the effect of influence factor λ and decay factor μ in formulas (22) and (2). Each data point is the average of the data obtained in 20 experiments.

4.2.1. Influence Factor. In order to discuss the effect of the influence factor, sensor node can obtain the interaction history of other normal nodes by searching the whole network in this subsection. The initial direct trust value of node i is set to 0.5, and then we reevaluate the global trust degree by using the increase recommendation records of intermediate nodes. λ is set to 1.0, 0.5, and self-adaptive influence factor, respectively. Figure 4 shows the results.

In Figure 4, when $\lambda = 1.0$, the intermediate nodes have no effect on the trust value. Therefore, the trust value of node i is always equal to 0.5. When $\lambda = 0.5$, the trust value of node i increases quickly, which reflects the effect of intermediate nodes and indirect trust evaluation. When λ is calculated by entropy theory, self-adaptive influence factor is used to measure the weights of direct trust degree and indirect trust degree. TMBNRT algorithm does not search the whole network to obtain indirect interaction records, and no more intermediate nodes are required. Hence, the trust degree of TMBNRT algorithm can quickly reach a stable level.

4.2.2. Decay Factor. In order to discuss the effect of decay factor and the dynamic properties of trust, we divide time into 20 sequences. In the first 10 sequences, node i cooperates well with other nodes to obtain positive evaluation.

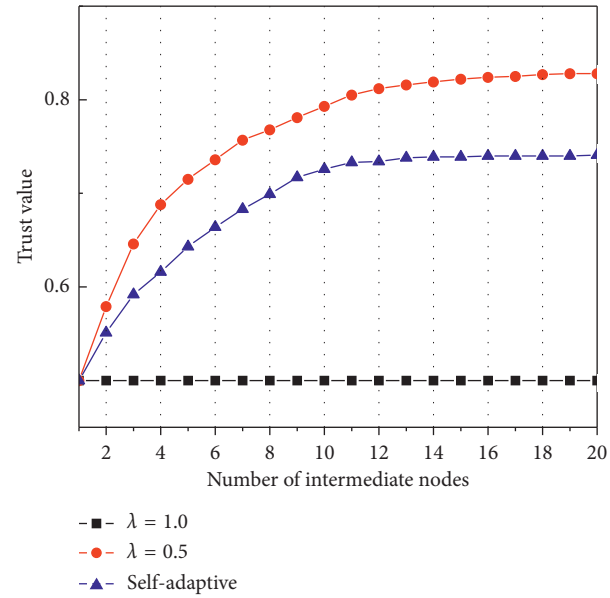

Figure 4: The effect of the influence factor to the trust value.

In the next 10 sequences, node i behaves badly and obtains negative evaluation. In this experiment, μ is set to 1.0, 0.5, and 0.0, respectively. Figure 5 shows the results.

As shown in Figure 5, when $\mu = 0.0$, only the latest time segment is considered. The trust degree of node i reaches a stable value very quickly. However, each failure will lead to a significant decline in the trust value. The change of trust degree has too much volatility. When $\mu = 1.0$, the decay factor is not considered. The whole history records are aggregated and the trust value drops slowly when node i behaves badly. Obviously, it is considerably easier for a malicious node to launch an on-off attack. When $\mu = 0.5$, the trust value keeps up with the node's current status much better. This indicates that the decay factor can objectively and effectively reflect the behavior of the sensor nodes.

4.3. Evaluation of Node Recovery Technique. In this experiment, we compare the recovery overhead under different recovery parameters. Principal parameters of recovery overhead include the total distance moved (namely, the total distance that nodes collectively travel during the recovery procedure) and the total energy consumed (namely, the total energy consumption during the recovery procedure and execution procedure) [34]. Experimental settings are listed as follows: Decay factor is set to 0.8, and the number of sensor nodes is set to 200. The results are shown in Figure 6.

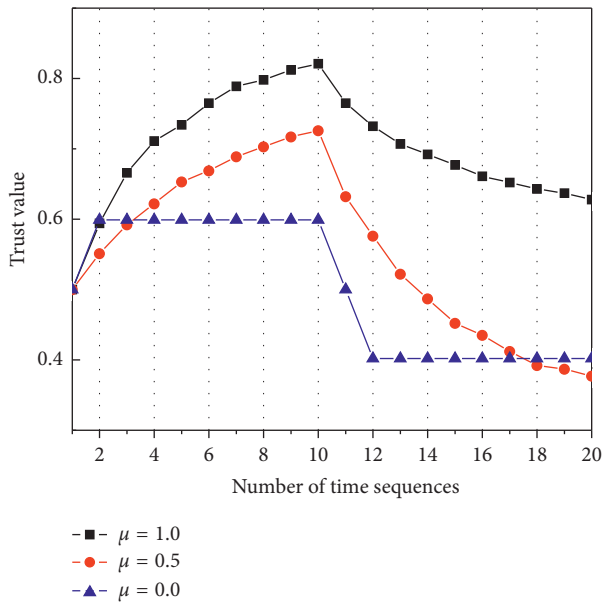

Figure 5: The effect of the decay factor to the trust value.

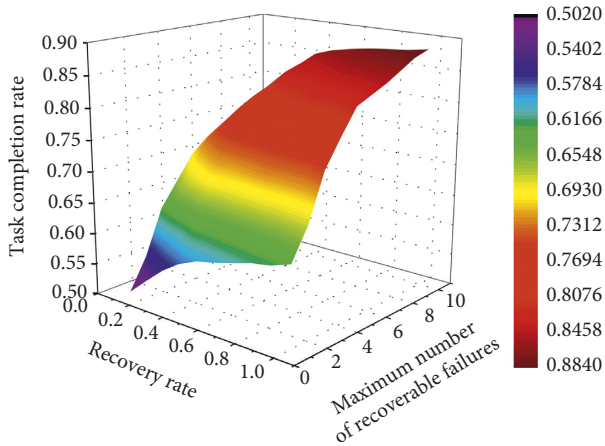

Figure 6: Comparison of task completion rate with respect to the increase in x_i and N_i.

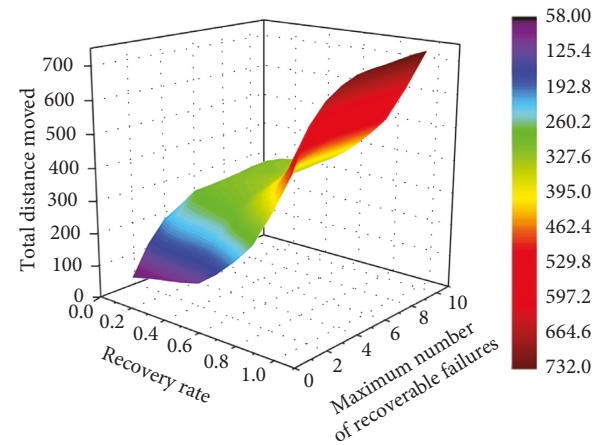

Figure 7: Comparison of total distance with respect to the increase in x_i and N_i.

As shown in Figure 6, with the increase of x_i and N_i, the task completion rate is growing as well. This indicates that the trust model considering node recovery technique can effectively improve the task completion rate. In the meantime, N_i has more influence on the task completion rate based on the observation. When x_i remains unchanged, the task completion rate has increased by 45.63% on average with N_i increased from 1 to 10. On the contrary, the task completion rate has increased only 24.94% with x_i increased from 0.1 to 1.0.

In addition, the growth rate of task completion rate with different parameters has also changed. When $N_i \leq 5$, the task completion rate grows fast. However, it increased relatively slower when $5 \leq N_k \leq 10$. This is because the majority of recoverable failures can be recovered within five recovery procedures.

The higher value of x_i and N_i can obviously improve the reliability of WSN. However, with the increase of the recovery rate and recovery procedures, the cost has increased as well. Therefore, we try to investigate the relation of x_i, N_i, and the total distance moved. The results are shown in Figure 7.

As shown in Figure 7, when $0 < x_i < 0.6$, the growth rate of the total distance moved is relatively lower. However, when $x_i > 0.6$, the growth rate increases rapidly. We can see that the total distance is extremely large when $X_i = 1$. The main reason is that the recovery scheme tries to reposition some of the healthy nodes (such as its child node) to maintain the connectivity. Therefore, these nonlinear processes will result in noticeably increased cost of each recovery procedure. As the next step, we discuss the relation between the total distance and N_i. We have arrived at a conclusion opposed to x_i. When $N_i \leq 5$, the growth rate of the total distance is relatively higher. When N_i is close to 10, the growth rate decreased. The main reason is that most of the recoverable failures are recovered with fewer than 5 to 6 recovery procedures, and this conclusion is consistent with the experimental result shown in Figure 6.

4.4. Comparison of Trust Value. In this experiment, the sensor nodes are divided into two groups: malicious node and normal node. For each group, we compare the proposed TMBNRT algorithm with traditional RFSN algorithm [21] and BTMS algorithm [16] in trust value. Experimental settings are listed as follows: Decay factor is set to 0.8, X_i is set to 0.6, and N_i is set to 4. The number of sensor nodes is set to 200. The results are shown in Figure 8.

As shown in Figure 8(a), the normal nodes and the malicious nodes can be distinguished effectively and efficiently. This illustrates that the proposed TMBNRT algorithm can identify malicious nodes effectively to avoid malicious attacks. In Figure 8(b), the difference between the malicious node and the normal node is relatively small. In Figure 8(c), the trust value of two kinds of sensor nodes are very close, and it can be seen that the trust value in Figure 8(b) and Figure 8(c) is relatively low.

The main reason is that BTMS and RFSN do not take recoverable failures into account. Both algorithms assume that interaction failures are caused by malicious attack. Their

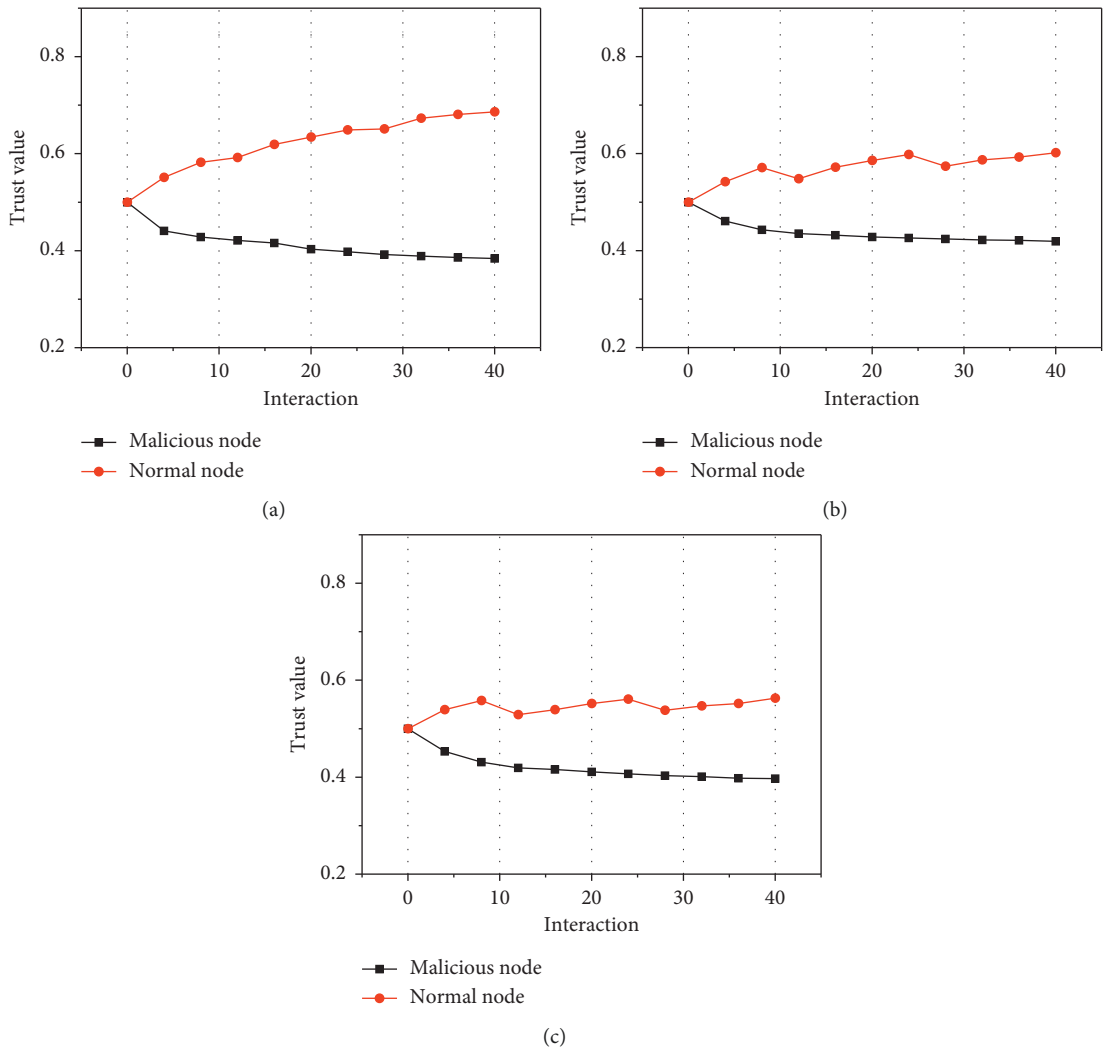

Figure 8: Comparison of the trust value of malicious nodes and normal nodes under a varying number of interactions: (a) TMBNRT, (b) RFSN, and (c) BTMS.

interaction results are described by binomial events. Once these recoverable failures occur within a normal node, binomial distribution-based trust model treats it as a malicious attack and this will result in a fast drop in the trust value of normal nodes. Therefore, BTMS and RFSN cannot effectively identify malicious nodes and normal nodes according to interactive behavior and trust value.

More importantly, the punishment factor in BTMS is used to show punishment to nodes' misbehavior, and the node needs much longer time to recover its trust value. Obviously, it is very difficult for normal nodes to recover its reputation once the failure appears. However, due to the unattended and harsh deployment environment, hardware failures and communication failures of the sensor nodes are inevitable. As a result, there is only a little difference in the trust value between normal nodes and malicious nodes in BTMS.

4.5. Comparison of Algorithm Performance. In this simulation experiment, with the same configuration, we compare

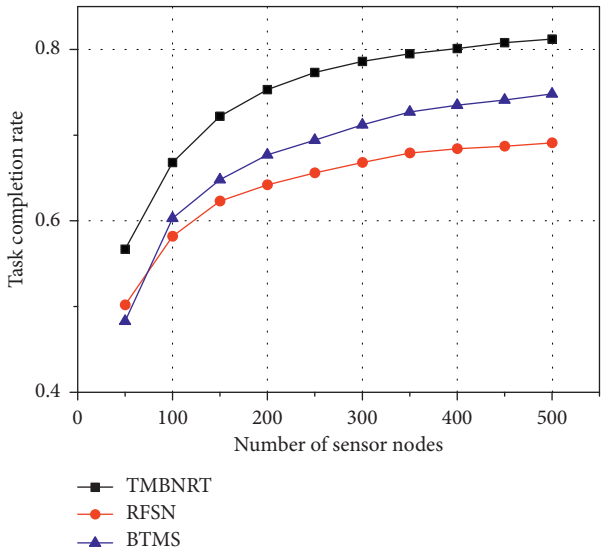

Figure 9: Comparison of task completion rate of TMBNRT with RFSN and BTMS under a varying number of nodes.

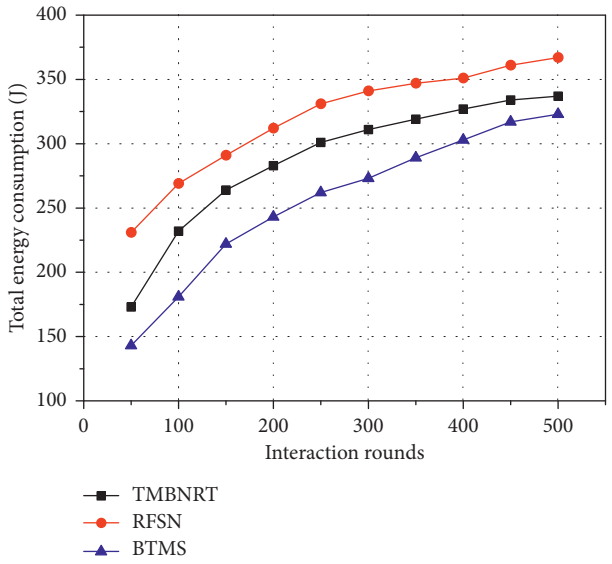

Figure 10: Comparison of total energy consumption of TMBNRT with RFSN and BTMS under varying rounds.

the proposed TMBNRT algorithm with RFSN algorithm and BTMS algorithm in task completion rate. The number of nodes is generated randomly from 50 to 500. The experimental results are shown in Figure 9.

In Figure 9, with the number of sensor nodes increasing, the task completion rates of three algorithms are increased as well. In the meantime, the task completion rate of TMBNRT is much higher than RFSN's and BTMS's. This indicates that the proposed trust model can assure the successful execution of tasks.

Then, in order to investigate the energy consumption of three algorithms, we compare TMBNRT algorithm with RFSN algorithm and BTMS algorithm in total energy consumption under varying interaction rounds. The number of sensor nodes is set to 200. The experimental results are shown in Figure 10.

As shown in Figure 10, the total energy consumption of TMBNRT is more than BTMS but less than RFSN. It is obvious that RFSN has the highest energy consumption among the three algorithms after 500 interaction rounds. The main reason is that RFSN needs to search the whole network to obtain the interaction records, while TMBNRT and BTMS only need to calculate the indirect trust degree when $\gamma < \gamma_0$.

Compared with the BTMS algorithm, the total energy consumption of TMBNRT is increased by 4.33%. However, the proposed algorithm improves the task completion rate by 8.57%. According to the experimental results, we can draw the following conclusion: by increasing some energy consumption, TMBNRT algorithm can efficiently reduce the failure rate of task execution. With the increasing number of interaction rounds, the performance gain is almost twice as much as the increase of energy consumption, which is very practical in WSN.

5. Conclusions

In this paper, by evaluating the trust value of sensor nodes, a trust management scheme based on the node recovery technique is proposed to decrease the probability of task execution failure. First, the traditional binary model for the interaction results is extended to the trinomial distribution. Second, the direct trust degree and the indirect trust degree are calculated using Bayesian theory and updated by the decay factor to enhance the sensitivity. The simulation results show that our proposed TMBNRT algorithm can effectively meet the reliability requirements of WSN and outperformed other representative algorithms.

The trust model in this paper is based on some simplified assumptions and parameters setting. Our future work is to increase the general usability of the proposed trust model, for example, the flexibility in selecting the appropriate node recovery parameters so as to find the best trade-off between the reliability and the energy consumption.

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 study was supported by the Natural Science Fund of Education Department of Anhui Province under grant no. KJ2019A0704 and the National Natural Science Foundation of China under grant no. 61802332.

References

[1] W. T. Zhu, J. Zhou, and F. Bao, "Detecting node replication attacks in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1022–1034, 2012.

[2] J. Cho, A. Swami, and I. Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 2, no. 2, pp. 562–583, 2010.

[3] J. Newsome, E. Shi, D. Song et al., "The sybil attack in sensor networks: analysis & defenses," in *Proceedings of the International Symposium on Information Processing in Sensor Networks*, IEEE, Berkeley, CA, USA, April 2004.

[4] J. Duan, D. Yang, H. Zhu, S. Zhang, and J. Zhao, "TSRF: a trust-aware secure routing framework in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 10, no. 1, article 209436, p. 14, 2014.

[5] H. Kaur and S. Saxena, "A review on node replication attack identification schemes in WSN," in *Proceedings of the International Conference on Computing*, IEEE Computer Society, Delhi, India, June 2017.

[6] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in *Proceedings of the IEEE Symposium on Security & Privacy*, IEEE, Oakland, CA, USA, May 1996.

[7] A. Dogan and F. Ozguner, "Reliable matching and scheduling of precedence constrained tasks in heterogeneous distributed computing," in *Proceedings of the 29th International Conference on Parallel Processing*, pp. 307–314, IEEE Computer Society, Toronto, Canada, 2000.

[8] W. Wang, D. G. Zeng, and J. Yao, "Cloud-DLS: dynamic trusted scheduling for cloud computing," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2321–2329, 2012.

[9] S. Che, X. R. Feng, and X. Wang, "A lightweight trust management based on bayesian and entropy for wireless sensor networks," *Security and Communication Networks*, vol. 8, no. 2, pp. 168–175, 2015.

[10] J. Kaur and K. Bansal, "A review on failure node recovery algorithms in WSN," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 8, pp. 530–534, 2015.

[11] M. Nandi, A. Dewanji, B. Roy et al., "Model selection approach for distributed fault detection in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, no. 5, pp. 444–447, 2014.

[12] Z. Sun, Z. Zhang, C. Xiao, and G. Qu, "D-S evidence theory based trust ant colony routing in WSN," *China Communications*, vol. 15, no. 3, pp. 27–41, 2018.

[13] G. Yan and L. Wenfen, "BeTrust: a dynamic trust model based on bayesian inference and tsallis entropy for medical sensor networks," *Journal of Sensors*, vol. 2014, Article ID 649392, 10 pages, 2014.

[14] V. R. Prabha and P. Latha, "Fuzzy trust protocol for malicious node detection in wireless sensor networks," *Wireless Personal Communications*, vol. 94, no. 4, pp. 2549–2559, 2017.

[15] A. Tajeddine, A. Kayssi, A. Chehab, and H. Artail, "Fuzzy reputation-based trust model," *Applied Soft Computing*, vol. 11, no. 1, pp. 345–355, 2011.

[16] R. Feng, X. Han, Q. Liu et al., "A credible bayesian-based trust management scheme for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, article 678926, p. 9, 2015.

[17] A. Ahmed and A. R. Bhangwar, "WPTE: Weight-Based Probabilistic Trust Evaluation Scheme for WSN," in *Proceedings of the IEEE International Conference on Future Internet of Things & Cloud: Workshops*, IEEE Computer Society, Prague,Czech, August 2017.

[18] W. Ya, N. Z. Meng-Ran, and Z. Jia, "Trust analysis of WSN nodes based on fuzzy theory," *International Journal of Computers and Applications*, vol. 39, no. 8, pp. 1–5, 2017.

[19] E. Aivaloglou and S. Gritzalis, "Hybrid trust and reputation management for sensor networks," *Wireless Networks*, vol. 16, no. 5, pp. 1493–1510, 2010.

[20] X. Wu, J. Huang, and J. L. Shu, "BLTM: Beta and LQI based trust model for wireless sensor networks," *IEEE Access*, vol. 7, pp. 43679–43690, 2019.

[21] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, pp. 267–281, 2008.

[22] W. Wang, G. Zeng, J. Zhang et al., "Dynamic trust evaluation and scheduling framework for cloud computing," *Security & Communication Networks*, vol. 5, no. 3, pp. 311–318, 2012.

[23] W. Wang and G. Zeng, "Bayesian cognitive trust model based self-clustering algorithm for MANETs," *Science China Information Sciences*, vol. 53, no. 3, pp. 494–505, 2010.

[24] M. Yuvaraja and M. Sabrigiriraj, "Fault detection and recovery scheme for routing and lifetime enhancement in WSN," *Wireless Networks*, vol. 23, no. 1, pp. 267–277, 2017.

[25] A. A. Abbasi, M. F. Younis, and U. A. Baroudi, "Recovering from a node failure in wireless sensor-actor networks with minimal topology changes," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 256–271, 2013.

[26] A. Rafiei, M. Abolhasan, D. R. Franklin et al., "Effect of the number of participating nodes on recovery of WSN coverage holes," in *Proceedings of the 27th International Telecommunication Networks and Application Conference (ITNAC 2017)*, IEEE Computer Society, Melbourne, Australia, January 2017.

[27] H. C. Shih, J. H. Ho, B. Y. Liao et al., "Fault node recovery algorithm for a wireless sensor network," *IEEE Sensors Journal*, vol. 13, 2013.

[28] H. Rue, A. Riebler, S. H. Sørbye, J. B. Illian, D. P. Simpson, and F. K. Lindgren, "Bayesian computing with inla: a review," *Annual Review of Statistics and Its Application*, vol. 4, no. 1, pp. 395–421, 2017.

[29] J. Dhananandhini, C. M. Niranjana, K. Rajeswari et al., "Detecting sensor node failure and node scheduling scheme for wsn," *International Journal of Engineering Research and Development*, vol. 9, no. 2, pp. 53–56, 2013.

[30] P. Qi and L. Li, "A fault recovery-based scheduling algorithm for cloud service reliability," *Security and Communication Networks*, vol. 8, no. 5, pp. 703–714, 2015.

[31] H. R. Shaukat and F. Hashim, "MWSN modeling using OMNET++ simulator," in *Proceedings of the International Conference on Intelligent Systems*, Warsaw, Poland, September 2014.

[32] S. Madden, "Intel Berkeley research lab data," 2003, http://berkeley.intel-research.net/labdata.

[33] J. Bahi, M. Haddad, and H. M. Kheddouci, "Efficient distributed lifetime optimization algorithm for sensor networks," *Ad Hoc Networks*, vol. 16, no. 2, pp. 1–12, 2014.

[34] H. Shi, K. Hou, H. Zhou et al., "Energy efficient and fault tolerant multicore wireless sensor network: E^2MWSN," in *Proceedings of the 7th International Conference on Wireless Communications*, pp. 1–4, IEEE, Wuhan, China, September 2011.