Research Article

A Credible Bayesian-Based Trust Management Scheme for Wireless Sensor Networks

Renjian Feng, Xiaona Han, Qiang Liu, and Ning Yu

1 School of Instrumentation Science and Opto-Electronics Engineering, Beihang University, Beijing 100191, China
2 Beijing Institute of System and Engineering, Beijing 100101, China

Correspondence should be addressed to Renjian Feng; rjfeng@buaa.edu.cn

Received 31 December 2014; Revised 2 June 2015; Accepted 7 June 2015

Academic Editor: Hongxin Hu

Copyright © 2015 Renjian Feng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of wireless sensor networks (WSNs), the security of WSNs is an important issue in this field because of their vulnerabilities to attacks. Designing a reasonable trust management scheme that can evaluate the trust relationships among sensor nodes accurately is a challenging but meaningful task. In this paper, a credible Bayesian-based trust management scheme (BTMS) is proposed. The overall trust value is aggregated by both direct and indirect trust information. The former is calculated by a modified Bayesian equation and updated by a sliding window. The latter is computed by recommendations from a third party. Moreover, the indirect trust computation is invoked conditionally according to the uncertainty of direct trust calculated via Entropy Theory and malicious feedbacks are excluded. Meanwhile, different recommendations are appropriately weighted in light of the trust levels of recommenders. Simulations are conducted and the results show that, compared with existing approaches, the proposed trust model performs better in defeating attacks.

1. Introduction

Wireless sensor networks (WSNs) consist of thousands of nodes which are small-sized, capable of sensing, and resource-constrained [1]. In the past few years, WSNs have proven to be one of the most useful technologies and attracted more and more research attention. With the capability of acquiring, processing, and transmitting data, WSNs have been employed in many application scenarios such as environment monitoring, battlefield detection, industry security surveillance, and medical care. However, sensor nodes are vulnerable to attacks due to the unattended and harsh deployment environment. In many applications, sensor nodes are always at risk of being compromised by adversaries who may capture nodes' cryptographic keys [2]. As a result, the adversary nodes may be taken as normal ones in the network, which makes it possible for the adversaries to intercept, delete, or insert information. Once a node is compromised, the availability and integrity of the whole network can be destroyed. In addition, attacks may occur at any time and are difficult to be predicted. Thus, network security is a vital issue needed to be addressed to guarantee correct operation of the whole network.

In the field of network security, asymmetry cryptographic protection is widely used to deal with external attacks in Internet, Peer-to-Peer, and Ad Hoc networks. Unfortunately, due to the complexity and huge computing memory, encryption algorithms are not suitable for processing capability constrained and energy limited sensor nodes. Furthermore, security mechanisms based on encryption can only solve external security problems and are incapable of coping with internal attacks. The special characteristics of nodes make WSNs different from other networks. Nodes may refuse to cooperate with service requester on the purpose of saving resource and this kind of nodes is called selfish nodes. Although they do not attack the network actively, a number of selfish nodes can cause serious consequences in the network. Nonetheless, the existing cryptographic protection is unable to identify the risk caused by authenticated selfish nodes.
Therefore, an effective scheme needs to be established to settle those problems.

Trust management can be used as a natural way to achieve the privacy and security of entities in WSNs [3]. Trust relationship is one of the most complex concepts in WSNs because of its uncertainty, asymmetry, and context independence [4]. Inherited from [5], trust is defined as the confidence of a node that another node will perform as expected. A virtuous trust relationship enables a node to predict and evaluate the interaction security to prevent itself from being attacked before the interactions actually take place. If a node is always willing to forward the received data packets, it will be considered trustworthy and holds a good reputation, where, as a consequence, its requests will be more possible to be satisfied. The basic idea of trust management scheme is to quantify trust to describe the trustworthiness, reliability, or competence of individual nodes [6]. To the best of our knowledge, lots of state-of-the-art models [3, 7–21] have been proposed in this field. Undoubtedly, the present achievements have greatly promoted related research. Nevertheless, some limitations still remain to be solved. For instance, many studies pay little attention to recommendation selection or distribute weights to trust decision factors subjectively, both of which lead to inaccurate trust evaluation. Hence, it is necessary to come up with a reasonable and credible trust management scheme.

Aimed at the former problems, we put forward a Bayesian-based trust management scheme, short for BTMS. This paper refines our previous work [22]. In this proposal, we use both direct and indirect trust to obtain the overall trust and improve the algorithm to make it more adaptive and credible. Firstly, with the interaction records, direct trust values are computed based on Bayesian Theory. Secondly, indirect trust is calculated only when the direct trust is highly uncertain and the Entropy Theory is used to evaluate the uncertainty. The recommendations received from highly reputed nodes are given more weights. By doing this, the defect of usual practice in which weights are assigned subjectively is overcome. Finally, a self-confidence factor is introduced when establishing the overall trust.

The remainder of this paper is organized as follows. Section 2 provides related works about trust management. In Section 3, BTMS is introduced in detail and Section 4 shows the simulation results and security analysis. Last, we conclude the paper in Section 5.

2. Related Works

Up to the present, typical trust management methods include Bayesian trust model, Entropy trust model, Fuzzy logic trust model, D-S Evidence trust model, and Game Theory trust model [23]. Bayesian methodology, in which trust calculation is based on nodes’ history behavior records, enriches the comprehension of trust in a probabilistic view [24]. Entropy is used to evaluate the randomness in a signal or event [11]. Thus, with uncertainty in it, trust can be expressed with entropy. Instead of giving a precise value, Fuzzy logic model estimates trust with an approximate set with the usage of natural language. D-S Evidence Theory can extend to subjective logic trust model and the judgment of event uncertainty degree is denoted by subjective logic. Game Theory model tries to detect and exclude selfish nodes by means of game playing.

Bayesian Theory is a widely used tool in trust evaluation. Ganeriwal et al. [12] came up with a classical reputation-based framework for sensor networks (RFSN) where nodes use reputation to evaluate others’ trustworthiness. In this model, a beta reputation system of sensor networks (BRSN) was presented. Also, researchers reached a conclusion that the distribution of reputation fitted beta function well and they used mathematical expectation as nodes’ trust value. However, only good reputation of a node is allowed to be propagated, which leads to the consequence that trust information about malicious or misbehavior nodes cannot be transmitted in time.

Mejia et al. [13] proposed a completely distributed trust model based on a noncooperative game that utilized a bacterial-like algorithm to let the nodes quickly learn the appropriate cooperation behavior. However, Game Theory is a suggestion for how game players ought to behave instead of a predictive tool. Therefore, Game Theory is not so appropriate to resolve trust problems in WSNs. Tian and Yang [14] presented $R^2$ Trust, which was a robust reputation and risk evaluation based trust management model. The model evaluated trust values in the view of both reputation and risk with direct interactions and peers’ referrals. However, it is essential to study how dynamic behaviors impact the trust value. In [15], Bao et al. came up with a hierarchical dynamic trust management protocol for cluster-based wireless sensor networks, which considered two aspects of trust degree: social trust and QoS trust, respectively. Intimacy and honesty were taken into account to measure social trust whereas energy and unselfishness were chosen to measure QoS trust. However, the update interval may impact the protocol performance. Anita et al. [16] presented a trust prediction model for routing based on Fuzzy Theory. In this model, only recommendations from a certain subset of neighbors with maximum number of interactions were adopted. Based on the historical behavior, fluctuations in trust value within a period of time, and recommendation inconsistency, the model could predict the future behaviors of neighbors. But the application of Fuzzy Theory may yield information loss and inaccurate result. Aivaloglou and Gritzalis [17] proposed a hybrid trust and reputation management which combined aspects from certificate-based and behavior-based approaches. This model exploited predeployment knowledge on network topology and information flows to support the needs of nodes with highly diverse network roles. However, it is difficult to get predeployment information. Combining fuzzy sets and grey theory, Wu et al. [18] presented an incentive trust model to evaluate each node’s reliability based on both direct and indirect relationship. Only those held higher trust values were allowed to forward data packets. Besides, a time slice scheme was introduced to ensure enough time for a node to enjoy its service. However, the computation is complex for nodes with limited processing ability. Zhang et al. [19] proposed a multiple-level trust management framework called ML-TRUST. In this framework, three levels of trust, namely,
subjective belief, objective reputation, and recommendation trust, were introduced to establish trustworthy relationships among nodes for their cooperation. However, this model does not include trust sharing and renewal mechanism.

3. BTMS Algorithm

In this paper, we define trust as the probability that node A holds on node B to perform as expected [5]. The overall trust is established by direct and indirect trust according to interaction history. Firstly, both direct trust and indirect trust value are calculated on the basis of Bayesian Theory and the indirect trust computation will be invoked only when the uncertainty of direct trust, which is calculated according to the Entropy Theory, is high. Secondly, the trust level of recommenders is applied to assign weights to different recommendations. Finally, the overall trust value is aggregated by introducing self-confidence factor. In this model, nodes communicate via a shared bidirectional wireless channel and operate in the promiscuous mode.

3.1. Direct Trust Evaluation

3.1.1. The Entropy Theory. Entropy Theory is a concept in the realm of thermodynamics, statistics, and information theory. It is the measurement of uncertainty or information quantity in a random signal or event [20]. The entropy of a random variable X is defined as

\[
H(X) = -\sum_{i=1}^{n} P(x_i) \log P(x_i). \tag{1}
\]

Here, \(P(X)\) is the probability density function of variable X. If X is binary, (1) can be written as

\[
H(X) = -P(x_i) \log P(x_i) - [1 - P(x_i)] \log [1 - P(x_i)]. \tag{2}
\]

According to (1), the coordinate plot of entropy is given in Figure 1. As shown, the entropy function is a strictly convex function in its definition domain \([0, 1]\]. Taking \(P(x_i)\) as the probability with which an event \(x_i\) will happen, the entropy function is symmetry using \(P(x_i) = 0.5\) as symmetry axis and \(0 \leq H(X) \leq 1\). When \(P(x_i) = 0.5\), \(H(X)\) reaches the maximum, which means that whether the event will occur is completely uncertain under this scenario. On the contrary, when \(P(x_i) = 1\) (or \(P(x_i) = 0\)), \(H(X) = 0\), which means that the event will (or will not) take place without uncertainty.

3.1.2. Direct Trust Value Calculation. The direct trust, denoted by \(DT_{ij}\), is derived from the direct observations of node \(i\) on node \(j\). Here, the evaluating node \(i\) and evaluated node \(j\) are denoted by \(N_i\) and \(N_j\), respectively.

Due to its flexibility and simplicity as well as strong foundations on the theory of statistics, beta distribution is employed as the prior distribution of interactions among nodes. Assuming that the way of future interaction is the same as that of the previous, the desired value of \(DT_{ij}\) can be represented by the mathematical expectation of beta distribution as follows:

\[
DT_{ij} = \frac{\alpha_{ij} + 1}{\alpha_{ij} + \beta_{ij} + 2}. \tag{3}
\]

Here, we define \((\alpha_{ij}, \beta_{ij})\) as the interaction record between \(N_i\) and \(N_j\), where \(\alpha_{ij}\) and \(\beta_{ij}\) are the number of successful and unsuccessful interactions, respectively. As is well known, trust is dynamic and related to risk. Hence, we modify (3) as follows:

\[
DT_{ij} = \frac{\alpha_{ij} + 1}{\alpha_{ij} + \beta_{ij} + 2} \left( 1 - \frac{\beta_{ij}}{W} \right). \tag{4}
\]

In (4), \(W\) is the total number of effect interaction records and the expression \((1 - \beta_{ij}/W)\) is called punishment factor. With the increase of \(\beta_{ij}/W\), \((1 - \beta_{ij}/W)\) decreases correspondingly. We introduce the punishment factor which shows strict punishment to nodes’ misbehaviors for three reasons. First, a malicious node can cooperate with others to accumulate good reputation and then choose not to cooperate to launch attacks. If a highly reputed node launches attacks or suffers network failures such as link failure or running out of energy, its direct trust value will decline rapidly because of the effect of punishment factor. Second, risk exists during interaction. So, it is of necessity to illustrate interaction security level. As the expression \((\beta_{ij}/W)\) can partly signify the risk of interaction, the punishment factor indicates the security level when \(N_i\) interacts with \(N_j\). Third, as the direct trust value is a ratio in our previous work [22], short-term and long-term experiences of interactions may lead to the same trust level. For example, with 5 successful interactions and 1 unsuccessful interaction, or 59 successful interactions and 19 unsuccessful interactions, the direct trust value is 0.75 according to [22]. But in this proposal, if \(W\) is 150, the direct trust values are 0.745 and 0.655, respectively. So, the punishment factor indicates nodes’ action stability and punishes those who misbehave.

3.1.3. Updating of Direct Trust. Trust value is obtained based upon history interaction records and changes dynamically with time. It is intuitive that old experience has less contribution and new experience has more influence on trust decision. In this paper, trust is updated by a sliding window \(W\). Only interaction records within the sliding window are
valid; that is to say, the maximum number of effective interaction records is \( W \). The length of sliding window \( W \), which reflects evaluator’s emphasis on history information, is preset according to application environment.

As Figure 2 depicts, each sliding window keeps a record of the latest \( W \) interactions and is divided into \( q \) slots which are marked \( q, q - 1, q - 2, \ldots, 2, 1 \) from right to left. A social phenomenon is that while it takes long-term consistent cooperative behavior to gain good reputation, only a few misbehaviors can ruin it, which implies that reputation is easy to lose and hard to get. Inspired by this, we introduce an adaptive forgetting factor \( \theta_l \) to set weights to different slots:

\[
\theta_l = 1 - D_l^{ij}, \quad l = 1, 2, \ldots, q.
\]

In (5), \( D_l^{ij} \) is the direct trust value at the end of \( l \)th slot. Equation (5) demonstrates that bad behavior is remembered for a longer time.

For slot \( l \), where \( l = 1, 2, \ldots, q - 1, q \), \( N_j \) monitors \( N_i \)'s behavior and counts the number of successful interactions \( \alpha_{ij}(l) \) and unsuccessful interactions \( \beta_{ij}(l) \) at the end of this slot. Thus, the update of trust is equivalent to updating the value of the two parameters \( \alpha_{ij} \) and \( \beta_{ij} \) in the following method:

\[
\alpha_{ij}^{\text{new}} = \sum_{l=1}^{q} \alpha_{ij}(l) \cdot \theta_l^{q-l}, \quad l = 1, 2, \ldots, q - 1, q.
\]

\[
\beta_{ij}^{\text{new}} = \sum_{l=1}^{q} \beta_{ij}(l) \cdot \theta_l^{q-l}, \quad l = 1, 2, \ldots, q - 1, q.
\]

The updated direct trust value can be represented as

\[
DT_{ij} = \frac{\alpha_{ij}^{\text{new}} + 1}{\alpha_{ij}^{\text{new}} + \beta_{ij}^{\text{new}} + 2} \left( 1 - \frac{\beta_{ij}^{\text{new}}}{W} \right).
\]

The sliding window is a FIFO queue and simulates the decaying process of interaction history. The targeted updating scheme has the following advantages. For one thing, the window abandons old records and incorporates new experiences as time elapses, which improves the adaptability and saves nodes’ memory. For another, this updating method clearly reflects the character that trust relationship decays over time and enhances the accuracy of trust quantification. Moreover, the updating method is also a kind of redemption mechanism. On account of environment change, link mistake, battery draining, or some other system faults, a good entity may behave abnormally. Those ones may be taken as malicious or even expelled from network. As the sliding window moves forward, the proposed update model can remove the effect yielded by accident.

3.1.4. Judgment of Direct Trust. The traditional way, where both direct and indirect trust are considered when computing the overall trust value, costs more energy and makes the load of resource-constrained sensor nodes heavy. In [22], we use confidence level to decide whether the direct trust is credible enough. However, the accuracy of interval estimation and confidence level are two tradeoff factors, which means they cannot be increased together. In an interaction, whether a node will act as expected or not is indeterminate. Inspired by human experience, if this kind of uncertainty goes down to a certain degree, there is no more need to consider the recommendation from others. In other words, direct observation is sufficient to run the trust evaluation. Hence, suppose \( H(DT_{ij}) \) is the entropy of direct trust and \( h \) is the threshold of uncertainty; if \( h \leq H(DT_{ij}) \leq 1 \), which means that the uncertainty of direct trust is high and more information is needed, then the indirect trust computation is invoked; otherwise, the overall trust \( OT_{ij} \) holds about \( N_j \) is simply set to direct trust value; that is, \( OT_{ij} = DT_{ij} \).

By this means, the energy consumption is reduced and the calculation process agrees more with human cognition.

3.2. Indirect Trust Evaluation

3.2.1. Recommendation Trust Calculation. When a node is considered “uncertain,” the recommendation trust is needed. The evaluating node \( i \) acquires the recommendation about node \( j \) through their common neighbor nodes \( k \), symbolized as \( N_k \). \( N_i \) has prior reputation information \( (\alpha_{ik}, \beta_{ik}) \) about \( N_k \) already. \( N_j \) broadcasts a query command to its neighbors and the common neighbors shared by \( N_i \) and \( N_j \) send back their interaction records with \( N_j \) in response. This process is shown in Figure 3. The red and blue dashed circles represent communication range of \( N_i \) and \( N_j \), respectively.

Suppose the recommendation provided by \( N_k \) is \( R_{kj}^k \). Given \( (\alpha_{ik}, \beta_{ik}) \) and \( (\alpha_{kj}, \beta_{kj}) \), the recommendation trust value is computed in the following way, where \( (R_{ij}^k, R_{ij}^k) \) is the recommendation interaction record:

\[
R_{ij}^k = \frac{2 \cdot \alpha_{ik} \cdot \alpha_{kj}}{(\beta_{ik} + 2) \cdot (\alpha_{kj} + \beta_{kj} + 2) + 2 \cdot \alpha_{ik}};
\]

\[
R_{ij}^k = \frac{2 \cdot \alpha_{ik} \cdot \beta_{kj}}{(\beta_{ik} + 2) \cdot (\alpha_{kj} + \beta_{kj} + 2) + 2 \cdot \alpha_{ik}};
\]

\[
R_{ij}^k = E \left[ \beta \left( R_{ij}^k + 1, R_{ij}^k + 1 \right) \right] = \frac{R_{ij}^k + 1}{R_{ij}^k + R_{ij}^k + 2}.
\]

3.2.2. Judgment of Recommender. Due to the existence of malicious nodes, not all recommendations are reliable and dependable feedbacks result in false outcome. In this work, we judge the credibility of recommenders to ensure accuracy.
when calculating indirect trust. Only recommendations from trustworthy entities are accepted. The trust degree of \( N_i \) towards recommender \( k \), notated by \( T_{ik} \), is derived from the following equation:

\[
T_{ik} = \frac{\alpha_{ik} + 1}{\alpha_{ik} + \beta_{ik} + 2}.
\] (9)

Suppose there are \( n \) recommenders and their trust values held by \( N_i \) are notated as \( T_{i1}, T_{i2}, \ldots, T_{in} \). If \( T_{ik} \geq \gamma \) (\( k = 1, 2, \ldots, n \)), where \( \gamma \) is a preset threshold and \( 0 \leq \gamma \leq 1 \), the recommendation from \( N_k \) is accepted. Otherwise, it will be totally neglected.

3.2.3. Weight Assignment. In the targeted trust management scheme, the weights of each recommendation need to be computed. In some existing works, weights are set in subjective ways which lack adaptability and cannot reflect the proportion of each recommendation correctly. Intuitively, \( N_i \) should give more weight to recommendations from recommenders with high reputation. Hence, we allocate weights based on the trust degree of recommenders to avoid individual preference. The following approach is taken to calculate the weight of \( R_{ij}^k \):

\[
\omega_k = \frac{T_{ik}}{\sum_{k=1}^{m} T_{ik}}, \quad k = 1, 2, \ldots, m.
\] (10)

Here, \( \omega_k \) is the weight of \( R_{ij}^k \) and \( m \) is the number of accepted recommendations. Apparently, \( 0 \leq \omega_k \leq 1 \) and \( \sum_{k=1}^{m} \omega_k = 1 \). Finally, the indirect trust value, denoted by \( \Gamma_{ij} \), is obtained:

\[
\Gamma_{ij} = \sum_{k=1}^{m} \omega_k \cdot R_{ij}^k.
\] (11)

3.3. Overall Trust Aggregation. The overall trust value \( \Omega_{ij} \), which node \( i \) holds about node \( j \), is established via the following formula:

\[
\begin{align*}
\Omega_{ij} &= \begin{cases} 
DT_{ij} & 0 \leq H\left( DT_{ij} \right) < h \\
\lambda \cdot DT_{ij} + (1 - \lambda) \cdot \IT_{ij} & h \leq H\left( DT_{ij} \right) \leq 1.
\end{cases}
\end{align*}
\] (12)

In (12), \( \lambda \) is referred to as self-confidence factor and calculated as follows:

\[
\lambda = 1 - \rho^t, \quad \rho \in (0, 1).
\] (13)

In (13), \( t \) represents the number of direct interactions between \( N_i \) and \( N_j \), \( \rho \) takes value in the range \((0, 1)\) and varies according to different application environment. Obviously, the weight of \( DT_{ij} \) increases with the rise of direct interaction records \( t \), which indicates that \( N_i \) is more willing to believe its own direct judgment. Importing self-confidence factor \( \lambda \) not only defends the false recommendation from malicious entity but is consistent with human interaction habit.

4. Simulations

In this section, performance and security assessments are conducted on Matlab platform. We compare BTMS model with our previous proposal, LTMBE in [22], and existing trust management schemes, typically RFSN. The results demonstrate that BTMS has a powerful capability of trust estimation and attack resistance.

4.1. Trust Evaluation for Normal Nodes. In this section, we assess the overall trust of normal nodes. It is assumed that a normal node always chooses to cooperate. As depicted in Figure 4, the overall trust increases with the increase
of successful interactions. Because a normal node always cooperates, the overall trust value approaches 1.

4.2. Rationality of Importing Punishment Factor. The punishment factor is exploited to show strict punishment to nodes’ misbehavior. In this subsection, we analyze the rationality of importing punishment factor. Here, we set the maximum valid historical record $W$ to be 150. As Figure 5 shows, the increase of number of unsuccessful interactions results in sharp decrease of direct trust value because misbehaviors are punished. $W$ varies according to application environment. The smaller $W$ is, the more attention the latest records receive and the more severe the punishment is.

As interaction goes on, nodes may get the same direct trust value in different phases in LTMBE. In Table 1 and Figure 6, an example is given to demonstrate the function of differentiating nodes’ action stability of punishment factor.

In Figure 6, the direct trust of target nodes obtained by LTMBE remains almost the same, whereas that obtained by BTMS decreases with the effect of punishment because nodes with more unsuccessful interaction records are punished. Consequently, their direct trust values are lower. In this way, BTMS can differentiate the action stability of target nodes that maintain the same direct trust value and punish those who behave abnormally.

4.3. On-Off Attack. Adversaries can launch on-off attack whereby nodes behave well or badly alternatively trying to remain undetected while causing damage [21]. To defend against this kind of attack, we adopt adaptive forgetting factor to weight corresponding records. Suppose that an attacker behaves well in the first 30 interactions to build up good reputation but behaves badly in the next 30 rounds. After that, it behaves well continuously. Here, we set the fixed forgetting factor to be 0.7.

We can see intuitively from Figure 7 that, in RFSN, albeit the trust value drops fast when the malicious node launches on-off attack, it can regain trust simply by cooperating a few times. Although the trust value in LTMBE tracks the node’s current status, its effectiveness is constrained. Conversely, in BTMS, the trust value keeps up with the node’s current status much better than LTMBE. More importantly, a node can recover its trust value but this recovery needs more good actions and much longer time. We can come to a conclusion that BTMS outperforms RFSN and LTMBE in inhibiting on-off attack.

4.4. Bad Mouthing Attack and Ballot Stuffing Attack. Reputation-based systems are highly susceptible to false recommendation which is originated from dishonest recommender. Specifically, false recommendation can be classified into bad mouthing attack and ballot stuffing attack. In bad mouthing attack, malicious nodes collude to provide unfairly negative ratings about normal nodes [23].

Rather, in ballot stuffing attack, also called false praising attack, the compromised nodes propagate unfairly positive information about malicious nodes [2]. To defeat bad mouthing attack and ballot stuffing attack, we introduce recommendation selection scheme, where recommendations from untrustworthy nodes will be rejected, and self-confidence factor when synthesizing trust values.
In the first place, the damage caused by poorly reputed malicious recommenders will be removed by our recommendation selection scheme. As Figure 8 shows, the trust value of a malicious node decreases fast and becomes very small due to improper performance. Once its trust degree falls below the threshold $\gamma$, recommendation from this node will be completely disregarded.

To further show the effectiveness of recommendation selection scheme, we compare our model with RFSN and LTMBE under bad mouthing attack and ballot stuffing attack when the direct interaction between the evaluating node and the evaluated node is 150. Here, we set $\gamma = 0.3$ and the experiment results are shown in Figure 9.

As RFSN only propagates good reputation, it performs well when a malicious node launches bad mouthing attack, but RFSN cannot deal with ballot stuffing attack well and establishes relative high trust value for the evaluated node. LTMBE also suffers from the two kinds of attacks even though it assigns low weights to false recommendations, which cannot completely remove the negative impact. In addition, both RFSN and LTMBE do not include independent judgment of recommender. On the contrary, the advantage of adopting recommendation selection scheme is obvious. Under both attacks, no matter what the proportion of untrustworthy recommenders is, BTMS establishes almost the same overall trust value as that when there are no malicious recommenders. So, BTMS is resilient and functions well when poorly reputed recommenders launch bad mouthing attack and ballot stuffing attack.

Although we disregard recommendations from attackers with bad reputation, highly reputed attackers who behave well aiming at remaining undetected may still exist. This kind of attackers destroys the network more secretly. RFSN eliminates this potential threat by simply prohibiting bad reputation disseminating. However, the attack resistance comes at the cost of system accuracy as no nodes are allowed to share their bad experience with others. The ability of LTMBE to resist these attacks under this situation is limited as the false recommendations are still taken into consideration. BTMS excludes these potential threats via utilizing self-confidence factor. As the direct interaction increases, the evaluating node gives more weight to direct trust which reflects real status of evaluated node. Thereby, the damage resulting from false recommendations provided by highly reputed dishonest recommenders is mitigated. We repeat the experiment when highly reputed recommenders launch bad mouthing attack and ballot stuffing attack. The results are shown in Figure 10. It is obvious that when highly reputed recommender launches bad mouthing attack or ballot stuffing attack, RFSN can only cope with bad mouthing attack well. Trust values obtained by LTMBE are still affected by both of the attacks no matter how many times the evaluating node directly interacts with the evaluated node. As anticipated, the more direct interactions there are, the more the overall trust values get close to true trust values in BTMS.

With the previous analysis, we can conclude that BTMS can withstand bad mouthing attack and ballot stuffing attack effectively, no matter whether the recommenders are trustworthy or not.

5. Conclusions

In this paper, a credible Bayesian-based trust management scheme, BTMS for short, is proposed. First of all, direct trust is built on the basis of Bayesian Theory and updated by an adaptive forgetting factor to enhance flexibility. The punishment factor is introduced to show strict punishment to misbehavior and differentiates nodes holding the same direct trust value. The calculation of indirect trust is invoked conditionally and the recommendation information is selected when computing indirect trust. Moreover, defect
caused by allotting weights subjectively is overcome by distributing weights according to trust level of recommenders. Afterwards, we use self-confidence factor to establish overall trust. Simulation results indicate that, compared with RFSN and LTMBE, BTMS is of great help in defending against on-off attack, bad mouthing attack, and ballot stuffing attack.

Trust mechanism provides security service for upper layer network application, such as secure routing, secure data fusion, access control, and topology control [23]. Take secure routing for example. A routing protocol with trust calculation inevitably consumes more energy than those without any trust mechanism. In our future work, we would like to focus
on the application of trust mechanism and find a tradeoff between security and energy saving.

**Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

**Acknowledgment**

The authors are grateful to the anonymous reviewers for their insightful comments.

**References**

[1] W. T. Zhu, J. Zhou, R. H. Deng, 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] O. Khalid, S. U. Khan, S. A. Madani et al., “Comparative study of trust and reputation systems for wireless sensor networks,” *Security and Communication Networks*, vol. 6, no. 6, pp. 669–688, 2013.

[3] M. K. Denko, T. Sun, and I. Woungang, “Trust management in ubiquitous computing: a Bayesian approach,” *Computer Communications*, vol. 34, no. 3, pp. 398–406, 2011.

[4] X.-Y. Li, F. Zhou, and X.-D. Yang, “Developing dynamic P2P trust model using theory of entropy-based multi-source information fusion,” *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 2, pp. 777–790, 2011.

[5] P. Trakadas, S. Maniatis, P. Karkazis, T. Zahariadis, H. C. Leligou, and S. Voliotis, “A novel flexible trust management system for heterogeneous wireless sensor networks,” in *Proceedings of the International Symposium on Autonomous Decentralized Systems (ISADS ’09)*, pp. 1–6, IEEE, Athens, Greece, March 2009.

[6] Y. L. Sun, Z. Han, and K. J. R. Liu, “Defense of trust management vulnerabilities in distributed networks,” *IEEE Communications Magazine*, vol. 46, no. 2, pp. 112–119, 2008.

[7] X. Y. Li, F. Zhou, and J. P. Du, “LDTs: a lightweight and dependable trust system for clustered wireless sensor networks,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 924–935, 2013.

[8] B. Qureshi, G. Y. Min, and D. Kouyatso, “A distributed reputation and trust management scheme for mobile peer-to-peer networks,” *Computer Communications*, vol. 35, no. 5, pp. 608–618, 2012.

[9] D. Chen, G. R. Chang, D. W. Sun, J. J. Li, J. Jia, and X. W. Wang, “TRM-to-F: a trust management model based on fuzzy reputation for internet of things,” *Computer Science and Information Systems*, vol. 8, no. 4, pp. 1207–1228, 2011.

[10] Y. L. Yu, K. Q. Li, W. L. Zhou, and P. Li, “Trust mechanisms in wireless sensor networks: attack analysis and countermeasures,” *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.

[11] H. Luo, J. M. Tao, and Y. Sun, “Entropy-based trust management for data collection in wireless sensor networks,” in *Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM ’09)*, pp. 3171–3174, Beijing, China, September 2009.

[12] 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, article 15, 2008.

[13] M. Mejia, N. Peña, J. L. Muñoz, O. Esparza, and M. A. Alzate, “A game theoretic trust model for on-line distributed evolution of cooperation in MANETs,” *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 39–51, 2011.

[14] C. Q. Tian and B. J. Yang, “R2 Trust, a reputation and risk based trust management framework for large-scale, fully decentralized overlay networks,” *Future Generation Computer Systems*, vol. 27, no. 8, pp. 1135–1141, 2011.

[15] F. Y. Bao, I.-R. Chen, M. J. Chang, and J.-H. Cho, “Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection,” *IEEE Transactions on Network and Service Management*, vol. 9, no. 2, pp. 169–183, 2012.

[16] X. Anita, M. A. Bhagaveni, and J. M. L. Manickam, “Fuzzy-based trust prediction model for routing in WSNs,” *The Scientific World Journal*, vol. 2014, Article ID 480202, 11 pages, 2014.

[17] D. Hongjun, J. Zhiping, and D. Xiaona, “An entropy-based trust management model based on beta function for distributed routing in Internet of Things,” *China Communications*, vol. 9, no. 4, pp. 89–98, 2012.

[18] S. Y. Che, R. J. Feng, X. Liang, and X. Wang, “A lightweight trust management model based on Bayesian and Entropy for wireless sensor networks,” in *Proceedings of The International Conference on Embedded Software and Systems (ICESS ’08)*, pp. 27–34, Sichuan, China, July 2008.

[19] P. Dong, J. F. Guan, X. P. Xue, and H. C. Wang, “Attack-resistant trust management model based on beta function for distributed routing in Internet of Things,” *China Communications*, vol. 9, no. 4, pp. 89–98, 2012.

[20] S. Y. Che, R. J. Feng, X. Liang, and X. Wang, “A lightweight trust management model based on Bayesian and Entropy for wireless sensor networks,” *Security and Communication Networks*, vol. 8, no. 2, pp. 168–175, 2015.

[21] G. E. P. Kumar, I. Titus, and S. I. Thekkekara, “A comprehensive overview on application of trust and reputation in wireless sensor network,” in *Proceedings of the International Conference on Modelling Optimization and Computing (ICMOC ’12)*, pp. 2903–2912, April 2012.

[22] M. Momani, S. Challal, and R. Alhmouz, “Bayesian fusion algorithm for inferring trust in wireless sensor networks,” *Journal of Networks*, vol. 5, no. 7, pp. 815–822, 2010.
