Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 652051, 12 pages
http://dx.doi.org/10.1155/2013/652051

Research Article

A Credible Routing Based on a Novel Trust Mechanism in Ad Hoc Networks

Renjian Feng, Shenyun Che, Xiao Wang, and Ning Yu

School of Instrumentation Science and Opto-Electronics Engineering, Beihang University, Beijing 100191, China

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

Received 8 January 2013; Accepted 20 March 2013

Academic Editor: Adel Soudani

Many existing routing protocols in Mobile Ad hoc Networks (MANETs) focus on finding paths in dynamic networks without considering security. In this paper, we propose a trust model which evaluates neighbours' direct trust by factors of encounter time, mobility, and successful cooperation frequency. The revised D-S evidence theory is used to combine multiple recommended pieces of evidence and obtain the recommended trust value. Then based on the novel trust mechanism, we propose a trusted routing protocol named TDS-AODV protocol by extending the AODV protocol. In this protocol, a node makes a routing decision according to the trust values of its neighbour nodes. Finally, two routes are built: the main route with highest route trust value in the candidate routes and a backup route. Simulation results reveal that TDS-AODV can eliminate malicious nodes effectively when building the route; furthermore, it also achieves better performance than TAODV and AODV in terms of throughput, packet delivery ratio, and average end to end delay.

1. Introduction

The past decade has witnessed tremendous research efforts devoted to Mobile Ad hoc Networks (MANETs). MANETs are temporary autonomous systems with the special characteristics of dynamic network topology, limited computational abilities, and continuously changing scale. Due to its flexibility, a MANET is attractive for applications, such as disaster relief, military service, and robot networks [1]. However, this flexibility also causes security problems. Routing security is one of the challenging issues in current research.

Traditional MANET routing protocols, such as destination-sequenced distance vector routing (DSDV) [2], dynamic source routing (DSR) [3], and ad hoc on-demand distance vector routing (AODV) [4], assume that all nodes in the network work in a benevolent manner and no predefined trust exists between communication partners. However, the fact is that malicious behavior among nodes exists; for example, selfish nodes deny relaying the packets of other nodes, and malicious nodes perform impersonation, fabrication, or modification attacks against the network traffic [5]. Hence, it is necessary to incorporate security mechanisms into MANET routing protocols to mitigate the impairment from malicious nodes. However, the security mechanism basing on the traditional cryptosystem is used to resist external attacks, but it cannot effectively solve the internal attacks by malicious nodes [6]. Therefore, the trust mechanism which is considered to be an effective measure to solve those questions has recently been studied.

In our trust mechanism, the successful cooperation frequency factor is considered in direct trust evaluation to guarantee the security of network. It is calculated according to its accumulated observations using the Bayesian inference which adopts Beta distribution. Unlike most trust mechanisms [7–12] that focus on trust evaluation without considering performance of the network, we take other two factors (factors of encounter time and mobility) into account. A good network performance can help save nodes' limited resources and prolong the network lifetime, which is very important in MANET. The network topology in MANET is dynamic; hence, the next hop of a node may not be its next hop the next moment. To create a relatively stable network topology as much as possible, we propose two factors, nodes' average encounter time and mobility, when calculating nodes' direct trust value. The two factors make the trust mechanism more suitable for resource-restricted MANET. D-S evidence

theory, which was first introduced by statistician of Dempster [13] and extended by Shafer [14], is used to calculate direct trust value, integrate indirect evidence, and obtain the overall trust value. We choose D-S evidence here because it does well in dealing with the uncertainty of trust value.

Based on the novel trust mechanism, we put forward a trusted routing protocol, by extending the AODV protocol in MANETs, TDS-AODV for short. In this protocol, a node evaluates its neighbours' trust value according to the trust model and selects reliable nodes as its next-hop nodes. A source can establish multiple reliable paths to a destination in one route discovery process. We consider the number of hops as well as the trust value of paths to the destination. A destination will respond with three shortest paths as candidates and the path trust will be calculated during the process of Route Reply (*RREP*) Message delivery. The one with maximum path trust will be selected as the forwarding route and the second reliable one will be regarded as the backup route. We perform some simulations to compare the performance of TAODV, AODV, and TDS-AODV on Matlab platform. Simulation results show that our method is practical to detect malicious nodes and outperform TAODV and AODV in throughput, packet delivery ratio, and average end to end delay.

The rest of this paper is organized as follows. Section 2 summarizes the related work on trust evaluation and trust-based routing protocols. Section 3 presents the novel trust evaluation mechanism. Section 4 describes TDS-AODV in detail. Section 5 provides the simulation studies. Finally, we conclude this paper in Section 6.

2. Related Works

Researchers are becoming more and more interested in integrating trust into a MANET and have proposed numerous works. In this section, we first focus our attention on trust evaluation models in MANETs and then discuss the trust based routing protocols in MANETs.

2.1. Trust Evaluation. Peng et al. [7] assessed the subjective trust of nodes through the Bayesian method, but they were not able to detect dishonest recommendations. Zouridaki et al. [8] chose to determine the node trustworthiness with respect to reliable packet forwarding by combining first-hand trust and second-hand trust information. However, the trust calculation in unsupervised ad hoc environment involved complex aspects such as availability and mobility. besides packet forwarding. Omar et al. [9] sought to establish a fully distributed trust model based on trust graphs and threshold cryptography.

At present, most of the trust evaluation literatures ignore the uncertainty of trust value. To deal with this problem, some researchers [10–12] resort to D-S evidence theory. D-S evidence theory has the capacity of expressing directly for "uncertain," which makes it suitable to calculate the trust value in MANETs. Xie et al. [10] proposed a trust model for MANETs based on D-S evidence theory. The model can be a good solution for the combination of pieces of evidence, but it failed in addressing the issues concerning conflicting recommendation pieces of evidence. In this paper, we adopt the revised D-S combination rule which includes a consistent intensity to calculate nodes' trust value.

2.2. Trust-Based Routing Protocols. Wang and Wu [15] introduced the trust metric which depended on network traffic statistics to evaluate the trust and then loaded the trust model on the previously proposed distance-based location-aided routing (LAR). The algorithm utilized direct trust and recommendation trust to prevent malicious nodes from joining the forwarding. Li et al. [16] built a simple trust model to evaluate neighbours' behaviours forwarding packets and proposed a trust-based reactive multipath routing protocol extending from AODV. Peng et al. [17] incorporated a new dynamic trust mechanism which was based on multiple constraints and collaborative filtering into the extending DSR. Narula et al. [18] selected soft encryption systems and implemented them in conjunction with a trust-based reputation system and a multipath routing to provide a secure routing scheme. The implementation of this trust-based approach using DSR was then discussed. Sirotheau and Sousa [19] proposed an evaluation mechanism that aimed to mitigate routing misbehavior and other network failures. Four attributes of the routes were considered: level of activity, trust, mobility, and number of hops.

When transmitting a packet to a given destination, a node may have two routes: one is short but incredible while the other is long but credible. One of our main aims is to design a rational strategy which involves both hop counts and trust values in making decisions. The detailed implementation of our scheme is a secure extension of the AODV. Because of its ability to cope with network dynamic changes and repair broken links in routes, AODV is one of the promising protocols for deployment in a MANET.

3. Trust Model Based on D-S Evidence Theory

Trust model essentially performs trust derivation, computation, and application [20]. Trust applications including trust-based route discovery and route selection will be discussed in the next section.

3.1. D-S Evidence Theory. D-S evidence theory is based on the identification frame Ω set comprised by basic propositions which are both exclusive and exhaustive. 2^{Ω} is the power set of Ω, that is, the set of all the possible propositions based on Ω. Here we define Ω as $\{T, -T\}$, where T and $-T$ represent two trust states, namely, credible and incredible. 2^{Ω} is $\{\emptyset, \{T\}, \{-T\}, \{T, -T\}\}$, in which $\emptyset$, $\{T\}$,$\{-T\}$, and $\{T, -T\}$ represent the empty set, the propositions of nodes' "Trust", "Distrust", and "Uncertain", respectively. There are definitions of basic reliability function m on 2^{Ω}: $2^{\Omega} \rightarrow [0, 1]$, Belief Bel: $2^{\Omega} \rightarrow [0, 1]$ and Plausibility Pl: $2^{\Omega} \rightarrow [0, 1]$, satisfying the following equations:

$$m(\emptyset) = 0,$$

$$\sum_{A \subseteq \Omega} m(A) = 1, \quad A \neq \emptyset,$$

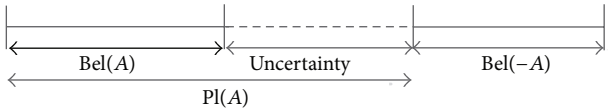

Figure 1: Belief (Bel) and Plausibility (Pl).

$$\begin{aligned} \mathrm{Bel}(A) &= \sum_{B \subseteq A} m(B), \quad \forall A \subseteq \Omega, \\ \mathrm{Pl}(A) &= 1 - \mathrm{Bel}\left(\overline{A}\right), \quad \forall A \subseteq \Omega, \end{aligned} \tag{1}$$

where A is named focal element, $m(A) > 0$ is the basic confidence level of A, representing how much the evidence supports A to happen.

The difference between Belief and Plausibility is referred to as Belief Interval. It is represented by the range of maximum uncertainty. The relationship of Belief and Plausibility is shown in Figure 1.

3.2. Trust Factors. The definition of "Trust" in this paper refers to the confidence that node i has on node j about the ability to forward packets successfully. Nodes tend to select the neighbour that has higher trust value as the intermediate node. In general, the trust between nodes only has some connection with malicious behaviors; however, we should consider more factors that depend on the interactions between neighbour nodes in a MANET due to its flexibility.

3.2.1. Factor of Average Encounter Time $ACF_{i,j}$. The concept of average encounter time does well in quantifying node's encounter history record. Encounter means that two nodes enter each other's wireless transmission range. The larger the $\mathrm{ACF}_{i,j}$ is, the more possibly node i chooses node j as the next hop. The $\mathrm{ACF}_{i,j}$ during period T is calculated by the following equation

$$\mathrm{ACF}_{i,j} = \frac{\sum_{t=0}^{t=T} \delta_{i,j}}{T}. \tag{2}$$

If two nodes enter each other's wireless transmission range $\delta_{i,j} = 1$, else $\delta_{i,j} = 0$. For example, in Figure 2, node i and node j encounter three times during period T; the $\mathrm{ACF}_{i,j}$ is:

$$\mathrm{ACF}_{i,j} = \frac{\sum_{t=0}^{t=T} \delta_{i,j}}{T} = \frac{T_2 - T_1 + T_4 - T_3 + T_6 - T_5}{T}. \tag{3}$$

3.2.2. Factor of Mobility $MOL_{i,j}$. The topology of MANET is dynamic due to the node movement; hence, in order to establish a more stable routing, it is necessary to take the node mobility into account when a node selects its cooperative nodes. The factor $\mathrm{MOL}_{i,j}$ is constructed as

$$\begin{aligned} \mathrm{MOL}_{i,j} &= \left| \frac{d_{i,j}(t+T) - d_{i,j}(t)}{d_{i,j}(t+T) + d_{i,j}(t)} \right|, \\ d_{i,j}(t) &= \sqrt{\left(x_i(t) - x_j(t)\right)^2 + \left(y_i(t) - y_j(t)\right)^2}, \end{aligned} \tag{4}$$

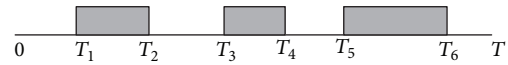

Figure 2: Average encounter time.

where $d_{i,j}$ denotes the distance between node i and node j at time t, $(x_i(t), y_i(t))$ and $(x_j(t), y_j(t))$ are the coordinates of node i and node j at time t, respectively.

3.2.3. Factor of Successful Cooperation Frequency $SCF_{i,j}$. Node i has a detection mechanism to obtain its interaction results $\mathrm{record}_{i,j} = (\alpha_{i,j}, \beta_{i,j})$ with node j. $\alpha_{i,j}$ and $\beta_{i,j}$, respectively, denote the number of successful cooperation and unsuccessful cooperation about node j observed by node i. Suppose $\mathrm{SCF}_{i,j}$ can be easily expressed by beta distribution, that is, $\mathrm{SCF}_{i,j} \sim \mathrm{Beta}(\alpha_{i,j}, \beta_{i,j})$. The factor $\mathrm{SCF}_{i,j}$ is constructed as

$$\mathrm{SCF}_{i,j} = \frac{\alpha_{i,j}}{\alpha_{i,j} + \beta_{i,j}}. \tag{5}$$

Once node j behaves badly, $\beta_{i,j}$ will increase and $\mathrm{SCF}_{i,j}$ will decrease, which leads to the decrease of the possibility that node i chooses node j as the next hop.

3.3. Direct Trust. Subject node i monitors the behaviors of object node j in one cycle and acquires the current trust value $\mathrm{CDT}_{i,j} = (m_{i,j}^C(\{T\}), m_{i,j}^C(\{T, -T\}), m_{i,j}^C(\{-T\}))$ based on the following expression:

$$\begin{aligned} m_{i,j}^C(\{T\}) &= \frac{\left(\omega_1 * \mathrm{ACF}_{i,j} + \omega_2 * \left(1 - \mathrm{MOL}_{i,j}\right) + \omega_3 * \mathrm{SCF}_{i,j}\right)}{\left(\sum_{k=1}^{3} \omega_k\right)}, \\ m_{i,j}^C(\{-T\}) &= \frac{\left(\omega_1 * \left(1 - \mathrm{ACF}_{i,j}\right) + \omega_2 * \mathrm{MOL}_{i,j} + \omega_3 * \left(1 - \mathrm{SCF}_{i,j}\right)\right)}{\left(\sum_{k=1}^{3} \omega_k\right)}, \\ m_{i,j}^C(\{T, -T\}) &= 1 - m_{i,j}^C(\{T\}) - m_{i,j}^C(\{-T\}), \end{aligned} \tag{6}$$

where $0 < \omega_k < 1$, $k = 1, 2, 3$, ω_k are determined by specific application environment, usually $\omega_3 > \omega_1$, $\omega_3 > \omega_2$ as security is more important.

Furthermore, the direct trust value is recalculated in accordance with history records. Assuming the direct trust value of latest cycle is $\mathrm{HDT}_{i,j}$, the update of direct trust value is calculated as follows:

$$\mathrm{DT}_{i,j} = \gamma \times \mathrm{HDT}_{i,j} + (1 - \gamma) \times \mathrm{CDT}_{i,j}, \tag{7}$$

where $\mathrm{DT}_{i,j}$ is the direct trust value of subject node i on object node j in current cycle, parameter γ is the adaptive time factor

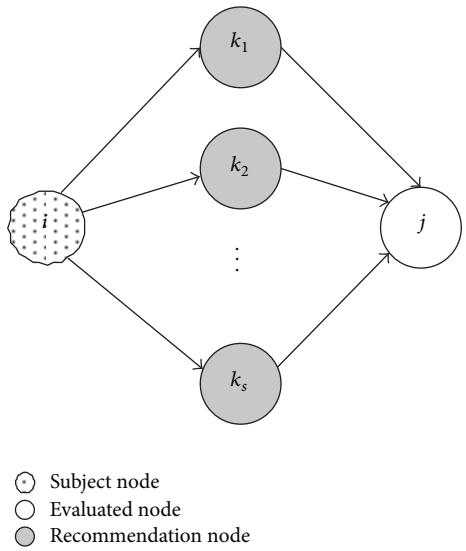

Figure 3: Recommendation relationship between subject node i and object node j.

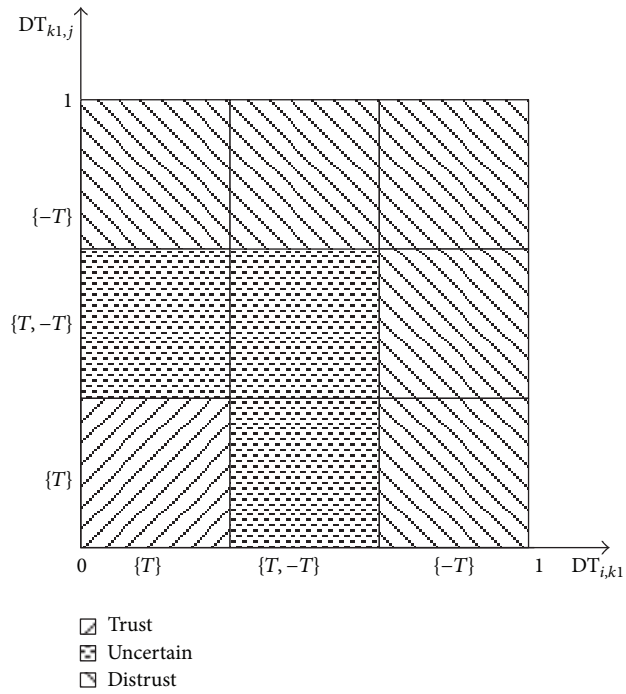

Figure 4: The process of trust transitivity.

used to weigh history experience against current information. To keep γ preferably dynamic characteristic, it is satisfied as

$$\gamma = \begin{cases} \gamma_s, & m_{i,j}^H(\{T\}) \geq m_{i,j}^C(\{T\}), \\ \gamma_l, & m_{i,j}^H(\{T\}) < m_{i,j}^C(\{T\}), \end{cases} \tag{8}$$

where $0 < \gamma_s < \gamma_l < 1$, the parameter $m_{i,j}^C(\{T\})$ and $m_{i,j}^H(\{T\})$ represent the trust components of $\mathrm{CDT}_{i,j}$ and $\mathrm{HDT}_{i,j}$, respectively.

3.4. Recommendation Trust Evaluation

3.4.1. Trust Transitivity. Suppose the recommended trust value of node i on node j can be obtained through s different paths, and the number of recommendation paths s depends on nodes' distribution and communication radius. In order to avoid trust recycle recursion and decrease network communication payload, the recommendation values are confined to direct trust value of the common neighbours owned by both node i and node j. As shown in Figure 3, node i can get the trust recommendation of node j from $k_1, k_2, k_3, \ldots, k_s$.

$\mathrm{RT}_{i,j}^1$ denotes the recommended trust value of node i on node j through recommendation path pt1 = $\{k1\}$. The vector forms of $\mathrm{RT}_{i,j}^1$, DT_{i,k_1}, $\mathrm{DT}_{k_1,j}$ are as follows:

$$\begin{aligned} \mathrm{RT}_{i,j}^1 &= \left(m_{i,j}^1(\{T\}), m_{i,j}^1(\{T,-T\}), m_{i,j}^1(\{-T\})\right), \\ \mathrm{DT}_{i,k_1} &= \left(m_{i,k_1}^D(\{T\}), m_{i,k_1}^D(\{T,-T\}), m_{i,k_1}^D(\{-T\})\right), \\ \mathrm{DT}_{k_1,j} &= \left(m_{k_1,j}^D(\{T\}), m_{k_1,j}^D(\{T,-T\}), m_{k_1,j}^D(\{-T\})\right). \end{aligned} \tag{9}$$

Let us set $\Theta = \{\{T\}, \{T,-T\}, \{-T\}\}$, A, E and $F \subseteq \Theta$. Then, the $\mathrm{RT}_{i,j}^1$ is calculated as follows:

$$m_{i,j}^1(A) = \begin{cases} m_{i,k_1}^D(A) \times m_{k_1,j}^D(A), & A = \{T\}, \\ \sum\limits_{E=A \text{ or } F=A} m_{i,k_1}^D(E) \times m_{k_1,j}^D(F), & A = \{-T\}, \\ 1 - m_{i,j}^1(\{T\}) - m_{i,j}^1(\{-T\}), & A = \{T,-T\}. \end{cases} \tag{10}$$

Using the symbol $\otimes$ to denote this operation, we get

$$\mathrm{RT}_{i,j}^1 = \mathrm{DT}_{i,k_1} \otimes \mathrm{DT}_{k_1,j}. \tag{11}$$

To vividly show the process of trust transitivity, we resort to Figure 4. It is obvious to see that as long as one of DT_{i,k_1} and $\mathrm{DT}_{k_1,j}$ is distrust, then $\mathrm{RT}_{i,j}^1$ is distrust.

Extending the above transitivity to multihop, we can get recommended trust through complex recommendation paths with many middle nodes

$$\mathrm{RT}_{i,j}^1 = \mathrm{DT}_{i,\bullet} \otimes \cdots \otimes \mathrm{DT}_{\bullet,j}, \tag{12}$$

where the symbol $\bullet$ indicates anonymous nodes in recommendation path.

3.4.2. Dynamic Aggregation of Recommended Trust. On the basis of trust transitivity, node i obtains recommended trust values on node j through s recommendation paths, namely,

$$\begin{aligned} \mathrm{RT}_{i,j}^{1} &= \left(m_{i,j}^{1}(\{T\}), m_{i,j}^{1}(\{T,-T\}), m_{i,j}^{1}(\{-T\})\right) \\ \mathrm{RT}_{i,j}^{2} &= \left(m_{i,j}^{2}(\{T\}), m_{i,j}^{2}(\{T,-T\}), m_{i,j}^{2}(\{-T\})\right) \\ &\vdots \\ \mathrm{RT}_{i,j}^{s} &= \left(m_{i,j}^{s}(\{T\}), m_{i,j}^{s}(\{T,-T\}), m_{i,j}^{s}(\{-T\})\right). \end{aligned} \tag{13}$$

Then, node i would aggregate these pieces of evidence to get a consensus on node j. Due to the existence of malicious nodes that may offer false recommendation, we introduce the revised D-S combination rule which adopts a consistent intensity to adjust weights of recommended trust values. The integration process is described in detail as follows.

Firstly, we compute the corresponding average weight denoted as I_u. The consistent intensity between $\mathrm{RT}_{i,j}^{u}$ and $\mathrm{RT}_{i,j}^{v}$ is defined as follows [21]:

$$I_{u,v} = 1 - \sqrt{\frac{1}{2}\left(\left\|\vec{m}_{i,j}^{v}\right\|^2 + \left\|\vec{m}_{i,j}^{u}\right\|^2 - 2\left\langle \vec{m}_{i,j}^{v}, \vec{m}_{i,j}^{u}\right\rangle\right)}, \quad v = 1,2,\ldots,s;\ u = 1,2,\ldots,s, \tag{14}$$

where $\|\vec{m}_{i,j}^{v}\|^2 = \langle \vec{m}_{i,j}^{v}, \vec{m}_{i,j}^{v}\rangle$, $\|\vec{m}_{i,j}^{u}\|^2 = \langle \vec{m}_{i,j}^{u}, \vec{m}_{i,j}^{u}\rangle$, $\langle \vec{m}_{i,j}^{v}, \vec{m}_{i,j}^{u}\rangle$ is the inner product of $\vec{m}_{i,j}^{v}$ and $\vec{m}_{i,j}^{u}$.

The difference between two pieces of recommended trust evidence increases with the reduction of consistent intensity. The lower the consistent intensity is, the more probably false trust recommendation may occur.

Furthermore, the matrix of consistent intensity composed of all the recommended trust values is defined as follows:

$$I_{s\times s} = \begin{bmatrix} 1 & I_{1,2} & \cdots & I_{1,s} \\ I_{2,1} & 1 & \cdots & I_{2,s} \\ \vdots & \vdots & \ddots & \vdots \\ I_{s,1} & I_{s,2} & \cdots & 1 \end{bmatrix}. \tag{15}$$

Through summation in row and normalization, the totally consistent intensity of recommended trust $\mathrm{RT}_{i,j}^{u}$, which is equal to the average weight I_u, is computed by

$$I_u = \frac{\sum_{v=1,\ v\neq u}^{s} I_{u,v}}{\mathrm{Max}\left(\sum_{v=1,\ v\neq w}^{s} I_{w,v}\right)_{1\le w\le s}}. \tag{16}$$

Then, the basic reliability function m of every recommended trust evidence is amended by I_u as follows:

$$\begin{aligned} m_{i,j}^{u\prime}(\{T\}) &= I_u \times m_{i,j}^{u}(\{T\}), \\ m_{i,j}^{u\prime}(\{-T\}) &= I_u \times m_{i,j}^{u}(\{-T\}), \\ m_{i,j}^{u\prime}(\{T,-T\}) &= 1 - m_{i,j}^{u\prime}(\{T\}) - m_{i,j}^{u\prime}(\{-T\}), \\ & u = 1,2,\ldots,s. \end{aligned} \tag{17}$$

Next, we apply the amended basic trust reliability function m to D-S combination rule. Assume that Bel_1 and Bel_2 are two trust degree functions that are on the same identification frame Ω; their basic reliability degree functions are m_1 and m_2. And m, the basic trust reliability function of Bel, can be expressed as follows:

$$\begin{aligned} m(A) &= m_1(A) \oplus m_2(A) = \frac{\sum_{X\cap Y=A} m_1(X)\times m_2(Y)}{1-K}, \quad A\neq\emptyset,\ A\subseteq\Omega, \\ m(\emptyset) &= 0, \\ K &= \sum_{X\cap Y=\emptyset} m_1(X)\times m_2(Y), \end{aligned} \tag{18}$$

where $\oplus$ is called "Direct Sum," representing the combinatorial operation between pieces of evidence.

Extending to s independent pieces of evidence which belongs to the same identification frame Ω, we can get

$$m(A) = \left(\left(m_1(A)\oplus m_2(A)\right)\oplus\cdots\right)\oplus m_s(A), \quad m(\emptyset)=0, \quad A\neq\emptyset,\ A\subseteq\Omega. \tag{19}$$

At last, the consistent recommended trust $\mathrm{RT}_{i,j}'$ is obtained.

3.5. Overall Trust Value Synthesis. Through the observation and recommendation from neighbour nodes, subject node i computes $\mathrm{DT}_{i,j}(t)$ and $\mathrm{RT}_{i,j}(t)$. The D-S evidence theory can combine conflicting and uncertain information to make a correct decision and accelerate converge rate of trust calculation. Consequently, it is used to synthesize $\mathrm{DT}_{i,j}(t)$ and $\mathrm{RT}_{i,j}(t)$ for the overall trust value $\mathrm{OT}_{i,j}(t)$:

$$\begin{aligned} m_{i,j}^{O}(A) &= m_{i,j}^{D}(A)\oplus m_{i,j}^{R}(A) \\ &= \frac{1}{N}\sum_{E\cap F=A} m_{i,j}^{D}(E)\times m_{i,j}^{R}(F), \quad A\subseteq\theta, \\ N &= \sum_{E\cap F\neq\emptyset} m_{i,j}^{D}(E)\times m_{i,j}^{R}(F) > 0. \end{aligned} \tag{20}$$

Algorithm 1 shows the process that subject node i judges whether node j is "Trust", "Distrust" or "Uncertain". The threshold values η and ξ are determined by specific application environment; here, we define $\eta = 0.4$ and $\xi = 0.1$. If the trust component is the biggest and the uncertain component is smaller than η, node i regards node j as "Trust." If the distrust component is the biggest and the uncertain component is smaller than η, node i regards node j as "Distrust." Otherwise, node i regards node j as "Uncertain".

4. Trust-Based Routing Protocol

In this section, we extend the AODV protocol to which can establish trusted route with minimum hops and maximum

(1) **if** $m^o_{i,j}(\{T, -T\}) > \eta$ **then**
(2) node i regard node j as "Uncertain";
(3) **else if** $m^o_{i,j}(\{T\}) - m^o_{i,j}(\{-T\}) > \xi$ && $m^o_{i,j}(\{T\}) > m^o_{i,j}(\{T, -T\})$ **then**
(4) node i regard node j as "Trust";
(5) **else if** $m^o_{i,j}(\{-T\}) - m^o_{i,j}(\{T\}) > \xi$ && $m^o_{i,j}(\{-T\}) > m^o_{i,j}(\{T, -T\})$ **then**
(6) node i regard node j as "Distrust";
(7) **else**
(8) node i regard node j as "Uncertain";
(9) **end if**

Algorithm 1: Process of judging node j's style.

path trust based on trust mechanism denoted by TDS-AODV. The differences between AODV and TDS-AODV are listed as follows.

(1) We append the model of trust computation and fields including $\mathrm{ACF}_{i,j}$, $\mathrm{MOL}_{i,j}$, $\mathrm{SCF}_{i,j}$, and $\mathrm{OT}_{i,j}$ in the neighbour table of each node.
(2) Every node maintains a local black list.
(3) We append T_{route} field in the route reply message and T_{route} denotes the accumulated route trust.
(4) We set backup route to avoid initiating the route discovery frequently.

4.1. Route Discovery. During the process of route discovery, when node i chooses another node j to forward a packet, node i may suffer some attacks from node j, such as black hole attack. Thus, it is important to choose a reliable next hop node. The process of judging whether node j can be the next hop of node i is as follows.

Step 1. Node i checks whether it has the trust value of node j ($\mathrm{OT}_{i,j}$); if it has, turn to Step 5, else turn to Step 2.

Step 2. Node i computes $D_{i,j}$ according to (6)–(8) and broadcasts a *Recommendation_Query* message to the common neighbours denoted as node k.

Step 3. After receiving the *Recommendation_Query* message, node k sends $D_{k,j}$ to node i if $m_{k,j}(\{T, -T\}) < \eta$.

Step 4. Node i calculates $\mathrm{RT}_{i,j}$ based on (13)–(18) and $\mathrm{OT}_{i,j}$ based on (19).

Step 5. Whether node j is reliable can be estimated using Algorithm 1. If node j is trusted, node i will update $\mathrm{OT}_{i,j}$ and regards node j as its credible next hop node, else node i will not choose node j to transmit packets and move node j into its local black list as a malicious node.

Once a node is in a black list, it will neither receive packets from its neighbour nor have its packets forwarded. That is, a malicious node in a black list is excluded by its neighbours. When a node exists in the black lists of all its neighbours, it will be excluded from the local network.

Sending packets by the trusted route will decrease the probability of malicious attacks and improve the survivability of MANETs. We evaluate the trustworthiness of a route by the trust value of nodes along the route, denoted by T_{route} [16]

$$T_{\mathrm{route}} = \prod m_{i,k}(\{T\}), \quad n_i \in \mathrm{route}, \quad n_k \in \mathrm{route}, \quad n_i \rightarrow n_k, \quad n_k \neq n_d, \tag{21}$$

where n_i and n_k are any two adjacent nodes among the route; n_d is the destination node in the route; $n_i \rightarrow n_k$ means that n_k is the next hop node of n_i; $n_k \neq n_d$ means that the destination node n_d should not forward the packets for itself and $m_{i,d}(\{T\})$ is not used to calculate the path trust to node n_d.

As shown in Figure 5, the trust value of path *P(A, B, C, D)* is equal to 0.68 (i.e. $T_{A,B,C,D} = m_{A,B}(\{T\}) \times m_{B,C}(\{T\}) = 0.85 \times 0.8 = 0.68$). Figure 6 shows an example of a multiple path. Among the three paths from A to H, path *P(A, F, G, H)* is the most credible path.

In our trusted routing mechanism, the route discovery includes three processes: (i) Route Request (*RREQ*) Message Delivery; (ii) Route Reply (*RREP*) Message Delivery; and (iii) route selection.

4.1.1. RREQ Delivery. An *RREQ* packet contains the following fields: ⟨*SourceAddr*, *SourceSequenceNo*, *BroadcastID*, *DestAddr*, *DestSequenceNo*, *HopCounter*⟩.

When the source node S needs to send data to the destination node D, it first checks whether there is a feasible path found between S and D. If so, S sends the data to D; otherwise, S will broadcast a *RREQ* to start a route discovery.

When any reliable intermediate node K whose authentication process was discussed before receives a *RREQ* packet from a neighbour J, it deals with the request according to the following steps.

Step 1. It checks whether one copy of the same *RREQ* has been received according to the *BroadcastID*. If so and the later copy has greater *HopCounter*, the *RREQ* will be discarded and the procedure ends; otherwise, go to Step 2.

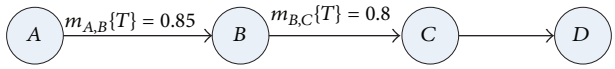

Figure 5: Path trust computation of a single path.

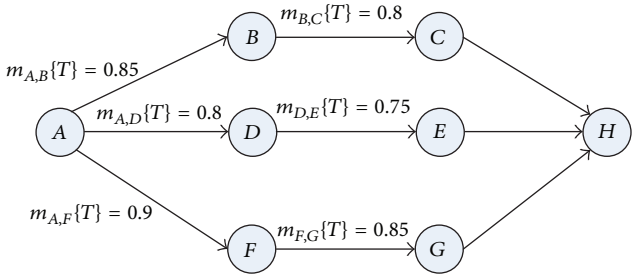

Figure 6: Path trust computation of a multiple path.

Step 2. If node J is not the source, node K creates a reverse route to S using the previous hop (node J) of the *RREQ* as the next hop.

Step 3. K checks whether there is a valid route to the destination. If so and the *DestSequenceNo* of the route is greater than that in the *RREQ*, K unicasts a Route Replay (*RREP*) message to S via J through the reverse route; otherwise, go to Step 4.

Step 4. K increases *HopCounter* by one and propagates the *RREQ* to all its neighbours.

The pseudocode of the *RREQ* is shown in Algorithm 2.

4.1.2. RREP Delivery. An *RREP* packet contains the following information: ⟨*SourceAddr*, *SourceSequenceNo*, *DestAddr*, *DestSequenceNo*, *HopCounter*, *LifeTime*, *PathTrust*⟩. When D receives the *RREQ* packet, it deals with the request according to the following steps.

Step 1. If it is the first time for D to receive a *RREQ* packet, then D sets a timer window t_D and records the route of *RREQ* in its cache and go to Step 6, otherwise go to Step 2.

Step 2. If t_D expires, it discards the follow-up *RREQ* packets, otherwise go to Step 3.

Step 3. If there are less than three routes in the cache of D, then add the new route in its cache and go to Step 6, otherwise go to Step 4.

Step 4. D compares the hop count of the new route with that of the route which owns the maximum hop count in its cache (denoted as route X). If the former is more than or equal to the latter, D discards the new *RREQ*, otherwise turn to Step 5.

Step 5. D uses the new route to substitute route X and then turns to Step 6.

Step 6. D sets T_{route} and then unicasts the *RREP* packets with T_{route} to the intermediate node.

After receiving a *RREP* packet, the intermediate node computes T_{route} according to (21) and updates the field of T_{route} then it forwards the *RREP* packet with T_{route}. The pseudo code of *RREP* delivery algorithm is shown in Algorithm 3.

4.1.3. Route Selection. When S receives the *RREP* packet, if the timer window t_D does not expire, it needs to update the T_{route} field of this message according to (21). Otherwise, S discards follow-up *RREP* packets and picks the one with largest T_{route} as its main route. The route with second largest T_{route} is regarded as backup route which aims at avoiding initiating the route discovery frequently. The pseudo code of route selection algorithm is shown in Algorithm 4.

4.2. Route Maintenance. After each successful route discovery takes place, S can deliver its data to D through a route. However, the route may break at any time instant due to the mobility of nodes or attacks. In order to maintain a stable and secure network connection, route maintenance is necessary to ensure the system survivability. AODV protocol designed two types of route maintenance mode one is a local repair mechanism and the other is that S reestablishes the route. Detailed process is discussed as follows.

Once the route is found, each node along the route periodically sends *HELLO* messages to its neighbour node for link failure detection. Link failure occurs when the neighbour node does not reply to the *HELLO* messages after a period of time. When a node N detects a link failure, it first sends a Route Error (*RERR*) message to S. S checks whether there is a backup route; if a backup route is found, S replaces the failure route with the backup and sends a *FoundBackup* message to N. Otherwise S sends a *NonBackup* message to N and then N starts a local repair mechanism. N broadcasts a *RREQ* message to find an alternative route between N and D. If no route is found, the system resorts back to another mechanism of sending a *RERR* message upstream to S, starting a new route discovery.

In TDS-AODV, besides link failure, if $T_{\text{route}} < T_{\text{thr}}$, S will also perform route maintenance which works as follows. During the transmission, if S finds the trust of a route has decreased, it sends a route check message along the route to check the route status and sets a timeout period to wait for the route check message from D. When S receives the reply, it will update the T_{route} and judge whether T_{route} is larger than T_{thr}. If $T_{\text{route}} < T_{\text{thr}}$, S resorts to the backup route and updates the path trust of the backup route (denoted as T_{rb}). If $T_{\text{rb}} > T_{\text{thr}}$, S discards the main route and uses the backup route to send packets. Otherwise, a new route discovery is triggered.

5. Simulation Studies

To evaluate the performance of TDS-AODV, we use the simulation tool MATLAB. In our simulation, fifty nodes at first are randomly placed in a specific field (100 m ×

(1) **To source node**:
(2) **if** there is a feasible path found between S and D **then**
(3) S sends data to D;
(4) **else**
(5) broadcasts the *RREQ* to start a route discovery;
(6) **end if**
(7) **To a reliable intermediate node**:
(8) checks whether one copy of the same *RREQ* has been received;
(9) **if** so and the later copy has greater *HopCounter* **then**
(10) discards *RREQ* and the procedure ends;
(11) **else**
(12) creates a reverse route to S using the previous hop of the *RREQ* as the next hop;
checks whether there is a valid route to the destination;
(13) **if** so and the *DestSequenceNo* of the route is greater than that in the *RREQ* **then**
(14) unicasts a Route Replay (*RREP*) message to S via J through the reverse route;
(15) **else**
(16) increases *HopCounter* by one;
propagates the *RREQ* to all its neighbours;
(17) **end if**
(18) **To destination node**:
(19) calls the process of route reply;
(20) **end if**

Algorithm 2: The *RREQ* delivery algorithm.

(1) **To destination node**:
(2) sets $T_{\text{route}} = 1$;
(3) **if** received the first *RREQ* packet **then**
(4) sets a timer window t_D;
increases the destination sequence number by 1;
records the route of *RREQ* in its cache;
sends the *RREP* with T_{route} along the path to the intermediate node;
(5) **else if** t_D expires **then**
(6) discards the follow-up *RREQ* packets;
(7) **else if** there are less than three routes in its cache **then**
(8) adds the new route in the cache;
sends the *RREP* with T_{route} along the path to the intermediate node;
(9) **else if** the hop count of the new route is more than or equal to that of route X **then**
(10) discards the new *RREQ*;
(11) **else**
(12) uses the new route to substitute route X;
sends the *RREP* with T_{route} along the path to the intermediate node;
(13) **end if**
(14) **To a reliable intermediate node**:
(15) updates T_{route} according to (21);
forwards the *RREP*;
(16) **To source node**:
(17) updates T_{route} according to (21);
calls route selection;

Algorithm 3: The *RREP* delivery algorithm.

(1) when source node receives the *RREP*, checks the t_s;
(2) **if** t_s does not expire **then**
(3) updates the T_{route}
(4) **else**
(5) discards the follow-up *RREP*;
selects the route with the largest as its main route;
picks the route with second largest T_{route} as its backup routes;
(6) **end if**

Algorithm 4: The route selection algorithm.

100 m) and move to another random position with a speed chosen between 0 to 30 m/s. The malicious nodes randomly drop data packets based on their trust value. The simulation parameters are listed in Table 1.

5.1. Performance Metrics. To measure the performance of our proposed TDS-AODV, we identify three metrics: (i) throughput: the number of packets transmitted per unit time from the source node to the destination node; (ii) packet delivery ratio: the ratio of the number of packets received to the total number of packets; and (iii) average end to end delay: the average delay between the sending of the packets by the source node and its receipt at the destination node.

The network topology of TDS-AODV was compared with that of TAODV [22] and AODV in this paper. We also carried out three simulations in terms of the maximum node speed and the proportion of malicious nodes to compare the above three performances of two protocol.

5.2. Simulation Results and Analysis. Figures 7 and 8 are the network topology of TDS-AODV and AODV with 20% malicious nodes. It is obvious to see that our method can avoid malicious nodes becoming the next hop effectively while in AODV malicious nodes can be selected as the next hop. The reason is that TDS-AODV takes nodes' trust value into account.

Figure 9 shows the average routing hop of TDS-AODV and AODV with different numbers of malicious nodes. when the number of malicious nodes accounts for a certain proportion of the number of total nodes, the average route hop of TDS-AODV is a little higher than that of AODV, because nodes would rather choose a relative longer path than choose malicious nodes as the next hop nodes in TDS-AODV. Although the path of TDS-AODV may be a little longer, the performance of TDS-AODV is still better than that of AODV as it eliminates malicious nodes out of the routing paths, which will be proven by the following simulation experiments.

Figures 10 and 11 depict the throughput of TDS-AODV, TAODV, and AODV. The routing throughput of TDS-AODV is averagely 29.60% lower than that of AODV and 21.27% lower than that of TAODV in Figure 10. This is because that our method can detect malicious nodes effectively and thus prevent the channel congestion. The throughput changes little

Table 1: Simulation parameters.

Parameters	Value
Simulation time	100 s
Number of nodes	50
Source node	Node 1
Destination node	Node 50
Area size	100 m × 100 m
Transmission radius	25 m
Max speed	0–30 m/s
Number of malicious node	0–20

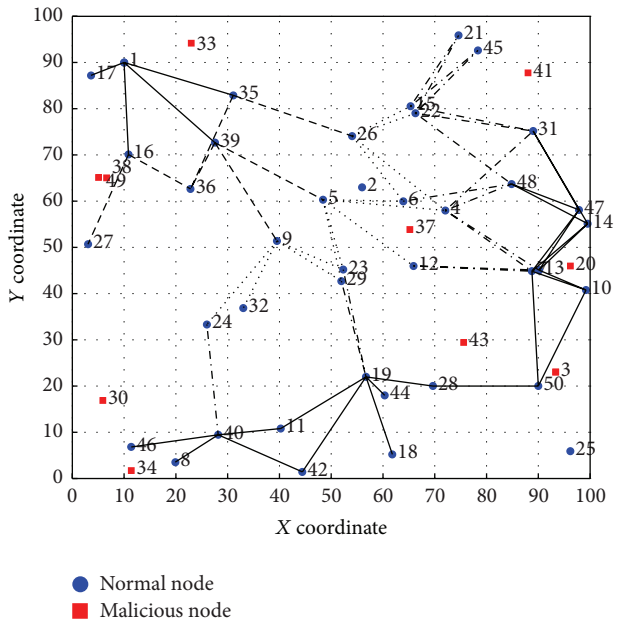

Figure 7: Network topology of TDS-AODV.

at different maximum speed which indicates our method has excellent dynamic. As shown in Figure 11, the throughput rises slowly with the increase in the number of malicious nodes. Besides, TDS-AODV rises more slowly than TAODV and AODV as it prevents the malicious nodes from becoming the next hop and affects less by malicious nodes.

The packet delivery ratio of TDS-AODV, TAODV, and AODV is shown in Figures 12 and 13. It can be observed

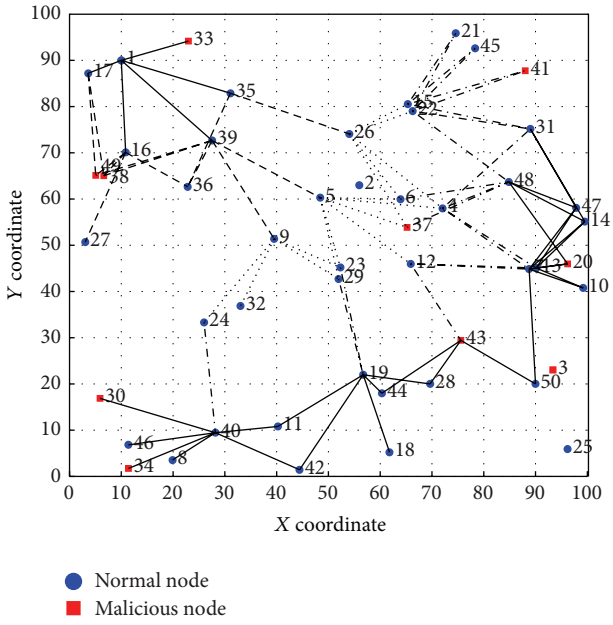

Figure 8: Network topology of AODV.

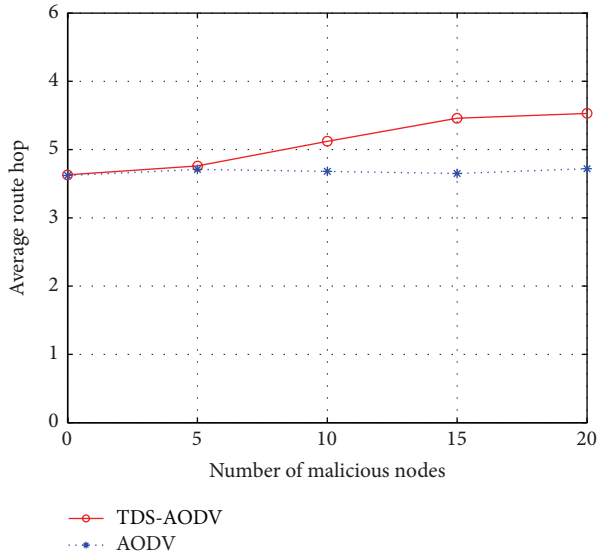

Figure 9: Everage route hop of TDS-AODV and AODV with different numbers of malicious nodes.

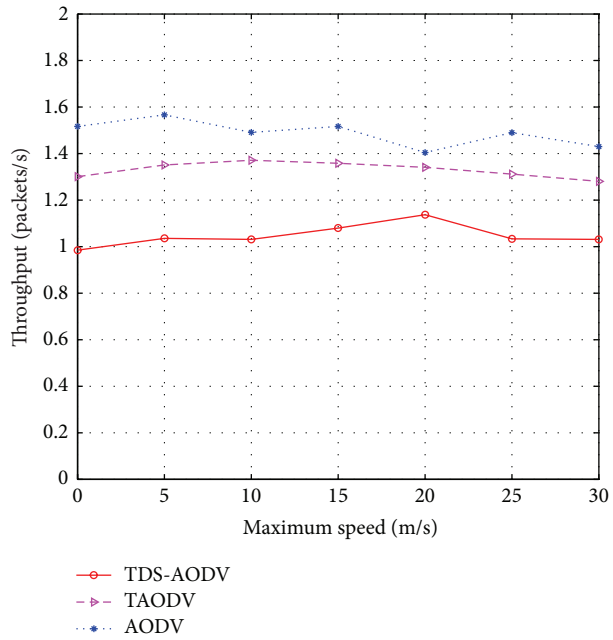

Figure 10: Performance of network throughput at different maximum speed.

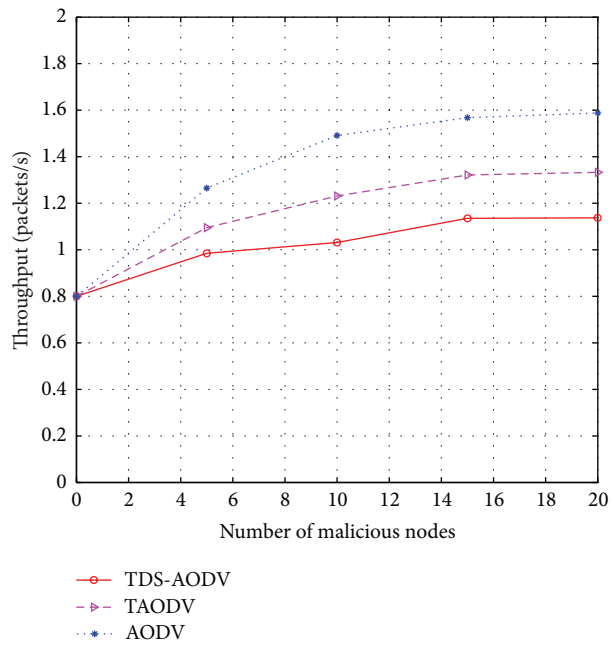

Figure 11: Performance of network throughput with different number of malicious nodes.

that TDS-AODV outperforms TAODV and AODV in the packet delivery ratio because of the fact that in TDS-AODV intermediate nodes make routing selection considering hop count and trust value. It shows the packet delivery ratio of TDS-AODV is averagely 46.24% higher than that of AODV and 17.18% higher than that of TAODV in Figure 12. Figure 13 indicates that TDS-AODV has better fault tolerance as its packet delivery ratio declines slowly with the increase in the number of malicious nodes.

We give the average end to end delay comparisons of TDS-AODV, TAODV, and AODV in Figures 14 and 15. As shown in Figure 14, the average end-to-end delay of three schemes rises very slowly with the increase in the maximum speed. However, the average delay of AODV is 18.73% higher than that of TDS-AODV and the average delay of TAODV is 7.74% higher than that of TDS-AODV in Figure 14 due to the

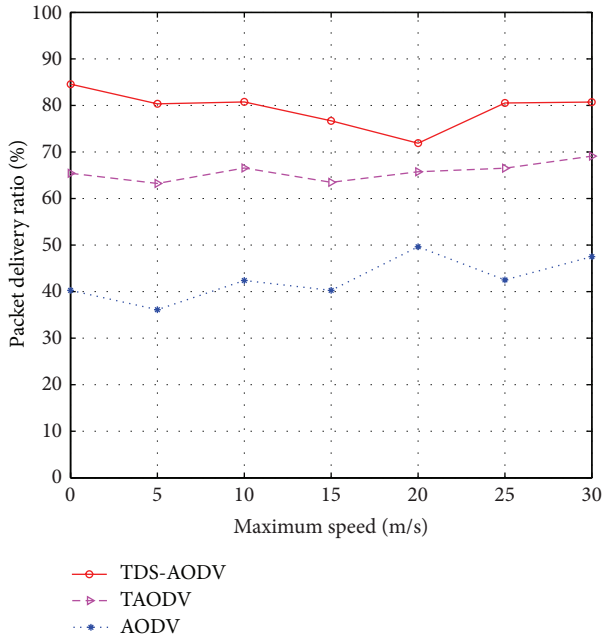

Figure 12: Performance of packet delivery ratio at different maximum speed.

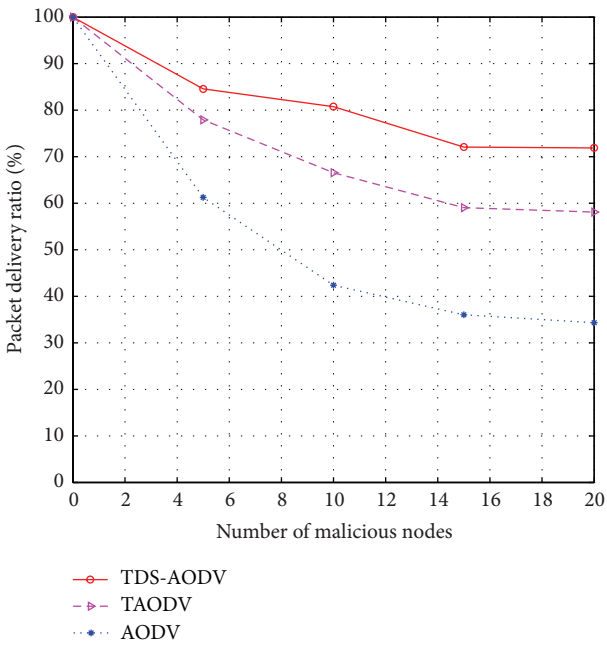

Figure 13: Performance of packet delivery ratio with different number of malicious nodes.

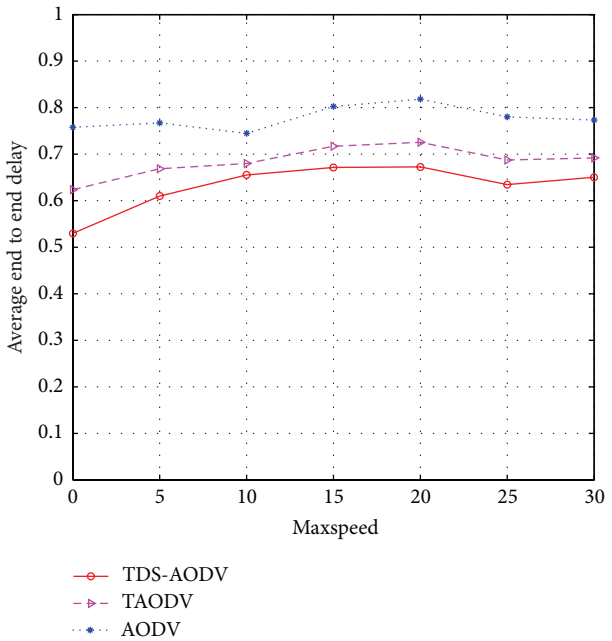

Figure 14: Performance of average delay at different maximum speed.

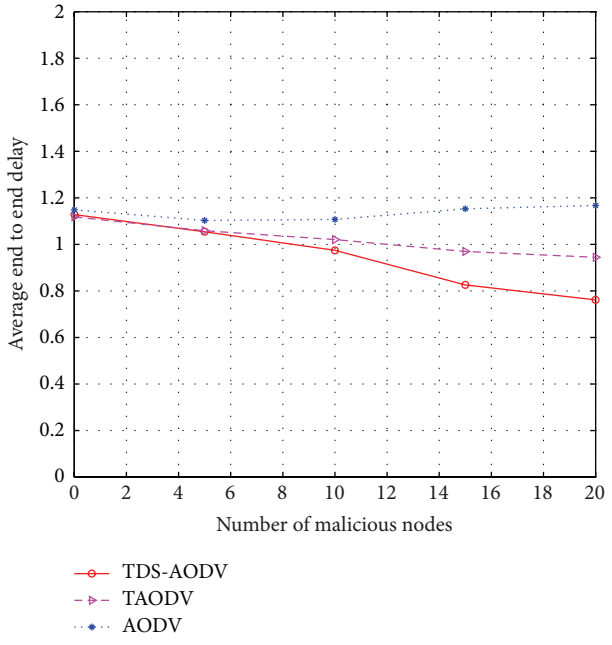

Figure 15: Performance of average delay with different number of malicious nodes.

lack of consideration of dynamic topology. Figure 15 depicts the performance of average delay with different number of malicious nodes. The average end to end delay of TDS-AODV declines faster than AODV. Because when intermediate nodes choose the next hop they will not consider the malicious nodes and thus save the time.

6. Conclusions

In this paper, we propose a novel trust mechanism after investigating on trust models of ad hoc networks and routing in current researches. In this trust mechanism, direct trust value on each neighbour node is calculated by using trust factors of average encounter time, mobility, and successful cooperation frequency, which are defined according to node

behaviors. Meanwhile, the revised D-S evidence theory is used to combine multiple recommended pieces of evidence and obtain the recommended trust value. Then, a trusted routing protocol based on the novel trust mechanism, by extending the AODV protocol is presented. In this protocol, a source establishes a main path and a backup path which are evaluated by two aspects: hop counts and trust values. At last, we validate the correctness and effectiveness of TDS-AODV by comparing its performance with TAODV and AODV on Matlab platform. Simulation results show that TDS-AODV is able to eliminate malicious nodes effectively when building the route and achieves an improvement in throughput, packet delivery ratio, and average end-to-end delay.

In our future work, we will conduct extensively simulation and rigorous analysis to quantify and evaluate the trade-off between the security and the nodes' energy consumption. In addition, a comprehensive performance evaluation will be conducted to compare TDS-AODV with other routing protocols (e.g., DSR).

Acknowledgments

The authors are grateful to the anonymous reviewers for their insightful comments. This work is supported by the National Natural Science Foundation of China under Grants no. 61201317 and no. 61001138.

References

[1] B. Wang, C. H. Huang, L. Y. Li, and W. Z. Yang, "Trustbased trustbased minimum cost opportunistic routing for Ad Hoc networks," *Journal of Systems and Software*, vol. 84, no. 12, pp. 2107–2122, 2011.

[2] C. E. Perkins and B. Highly, "Dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proceedings of the Conference on Communications Architectures, Protocols and Applications (SIGCOMM '94)*, pp. 234–244, 1994.

[3] D. B. Johnson and D. A. Maltz, "Dynamic source routing in Ad Hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996.

[4] C. E. Perkins and E. M. Royer, "Ad-Hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 90–100, New Orleans, LA, USA, February 1999.

[5] A. A. Pirzada, A. Datta, and C. S. McDonald, "Trust-based routing for Ad-Hoc wireless networks," in *Proceedings of the 12th IEEE International Conference on Networks (ICON '04)*, pp. 326–330, November 2004.

[6] H. Xia, Z. P. Jia, X. Li, and F. Zhang, "A subjective trust management model based on AHP for MANETs," in *Proceedings of the International Conference on Network Computing and Information Security (NCIS '11)*, vol. 1, pp. 363–368, Guilin, China, May 2011.

[7] S. C. Peng, W. J. Jia, and G. J. Wang, "Voting-based clustering algorithm with subjective trust and stability in mobile Ad-Hoc networks," in *Proceedings of the 5th International Conference on Embedded and Ubiquitous Computing (EUC '08)*, vol. 2, pp. 3–9, Shanghai, China, December 2008.

[8] C. Zouridaki, B. L. Mark, M. Hejmo, and R. K. Thomas, "E-hermes: a robust cooperative trust establishment scheme for mobile Ad Hoc networks," *Ad Hoc Networks*, vol. 7, no. 6, pp. 1156–1168, 2009.

[9] M. Omar, Y. Challal, and A. Bouabdallah, "Reliable and fully distributed trust model for mobile Ad Hoc networks," *Computers and Security*, vol. 28, no. 3-4, pp. 199–214, 2009.

[10] H. Xie, J. F. Ma, L. Yang, and X. W. Dong, "A trust method of MANETs based on D-S evidence theory," *International Journal of Advancements in Computing Technology*, vol. 4, no. 2, pp. 247–257, 2012.

[11] L. D. Huang, G. Xue, X. L. He, and H. L. Zhuang, "A trust model based on evidence theory for P2P systems," *Applied Mechanics and Materials*, vol. 20, no. 23, pp. 99–104, 2010.

[12] L. M. Jiang, J. Xu, K. Zhang, and H. Zhang, "A new evidential trust model for open distributed systems," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3772–3782, 2012.

[13] A. P. Dempster, "Upper and lower probabilities induced by a multi-valued mapping," *The Annals of Mathematical Statistics*, vol. 38, no. 2, pp. 325–339, 1967.

[14] G. Shafer, *A Mathematical Theory of Evidence*, Princeton University Press, New Jersey, NJ, USA, 1976.

[15] K. Wang and M. Wu, "Improved secure trust-based location-aided routing model for MANETs," *China Communications*, vol. 8, no. 3, pp. 154–162, 2011.

[16] X. Li, Z. P. Jia, P. Zhang, and H. Y. Wang, "Trust-based on-demand multipath routing in mobile Ad Hoc networks," *IET Information Security*, vol. 4, no. 4, pp. 212–232, 2010.

[17] S. C. Peng, W. J. Jia, G. J. Wang, J. Wu, and M. Y. Guo, "Trusted routing based on dynamic trust mechanism in mobile Ad-Hoc networks," *IEICE Transactions on Information and Systems*, vol. E93.D, no. 3, pp. 510–517, 2010.

[18] P. Narula, S. K. Dhurandher, S. Misra, and I. Woungang, "Security in mobile Ad-Hoc networks using soft encryption and trust-based multi-path routing," *Computer Communications*, vol. 31, no. 4, pp. 760–769, 2008.

[19] S. L. F. Sirotheau and R. T. D. Sousa, "Evaluating trust in Ad Hoc network routing by induction of decision trees," *IEEE Latin America Transactions*, vol. 10, no. 1, pp. 1332–1343, 2012.

[20] A. A. Pirzada, C. McDonald, and A. Datta, "Performance comparison of trust-based reactive routing protocols," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 695–710, 2006.

[21] A. L. Jousselme, D. Grenier, and E. Bosse, "A new distance between two bodies of evidence," *Information Fusion*, vol. 2, no. 2, pp. 91–101, 2001.

[22] X. Q. Li, M. R. Lyu, and J. C. Liu, "A trust model based routing protocol for secure Ad Hoc networks," in *Proceedings of the IEEE Aerospace Conference Proceedings*, vol. 2, pp. 1286–1295, March 2004.