Multi-Attribute Decision-Making Based Trusted Multipath Routing in Mobile Ad Hoc Networks

Qi Xue, Qingkuan Dong, Qing Li
The State Key Laboratory of Integrated Services Networks,
Xidian university, Xi'an Shaanxi 710071, China
xueqi2utopia@163.com, qkdong@mail.xidian.edu.cn, yake_y@aliyun.com

***Abstract*-Mobile Ad Hoc networks (MANETs) are a kind of acentric and self-organizing multi-hop wireless network. Topology changes frequently in mobile Ad Hoc networks. Node's energy, bandwidth and computation capability are limited. In addition, the open links are vulnerable to network attacks. Therefore the problems of the security and reliability of routing are very serious. Multipath routing is able to enhance the reliability of routing effectively, but in the existing Multipath routing algorithms, the resource attribute and behavioral characteristics of nodes are out of consideration.**

In this paper, a multiple attribute decision-making based trusted multipath routing algorithm is proposed. In this work, a trust management scheme is presented by introducing fuzzy attribute measure method of F->AHM. Furthermore, a trusted multipath routing algorithm based on AOMDV protocol is designed using the proposed trust management scheme. The simulation results show that the proposed algorithm has better dynamic adaptability and anti-attacks capability. The robustness and security of MANET routing are improved effectively.

Keywords: ***MANET, trusted routing, triangular fuzzy number, F->AHM, T-AOMDV***

I. INTRODUCTION

MANETs are wireless and they do not require any infrastructure to set up. This makes them ideal for military and emergency disaster. But they are prone to instability and vulnerability due to some of its characteristics such as openness, mobility, dynamic topology and restricted power supply. Consequently, their security requirements are very urgent and it is more difficult to design and implement security solutions for MANETs than for wired networks.

Up to now, Many schemes of secure MANET routing have been proposed, such as cryptography-based secure routing [1-2] and trust mechanism based secure routing [3-6]. The cryptography based secure routing schemes mainly use encryption and authentication methods to ensure the confidentiality, integrity and non-repudiation property of routing information. Although this scheme can resist almost all the external nodes attack, it cannot deal with the selfish behavior of internal uncooperative nodes effectively.

Trust mechanism based secure routing schemes are better to detect and avoid abnormal nodes, especially selfish nodes. Therefore, many kinds of trusted routing algorithm have been proposed in MANET, such as the routing algorithm based on subjective logic [7] and the routing algorithm based on D-S evidence theory [8]. However, there are still some important issues that should be further considered, such as how to consider various decision factors (DF), rather than a simple DF (i.e., packet forwarding rate)? How to deal with recommendation trust from different nodes? How to adopt incentive mechanisms (the positive behaviors is rewarded and the negative behaviors is punished)? How to treat the influence of historical trust on current trust?

In order to solve the problems above, we propose a trusted multipath routing protocol (called T-AOMDV) based on the dynamic trust mechanism, by extending AOMDV protocol in MANETs. In this protocol, considering the influence of various decision factors on node trust, we present a new trust management framework with multiple decision factors based on Triangular fuzzy attribute hierarchy method (F->AHM) [9], in which multiple decision factors including direct trust, recommendation trust, reward and penalty function, historical trust value, are incorporated to reflect trust relationship's complexity and uncertainty from different perspectives. F->AHM is used to make the weight classification of multi-attribute decision more scientific. In the process of route selection, both the path hop counts and routing trust value are considered. Finally, the proposed routing protocol is simulated and analyzed.

II. TRUST MANAGEMENT BASED ON MULTIPLE ATTRIBUTE DESCISION-MAKING

A. *Design of Trust management framework*

In this section we design a new trust management framework which is mainly composed of four sequentially connected modules, namely, "Trust collection", "Trust computing", "Trust decision" and "Routing protocol" , as shown in Fig. 1.

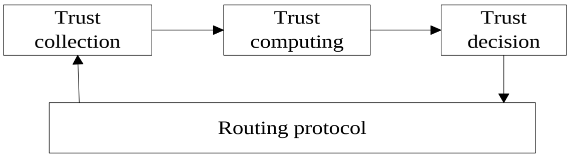
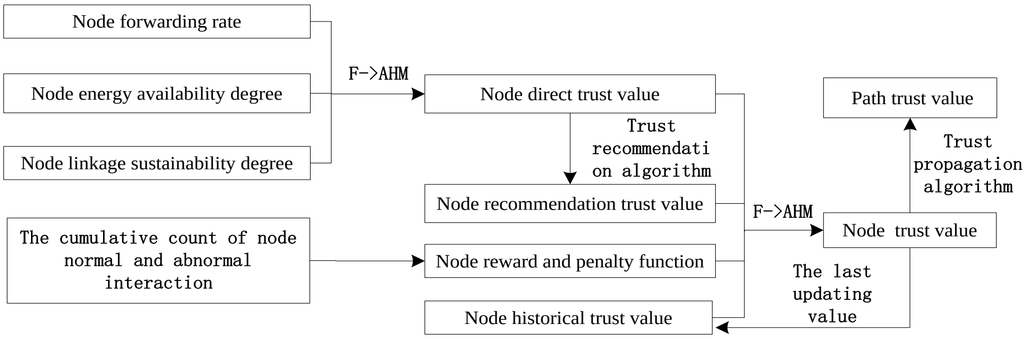


Fig. 1 The overall framework of trust management

Trust collection module mainly monitors nodes and collects node's resource attributes and behavior parameters as the evaluation factors of node trust. By this module, any nodes can extract trust factors from their neighbor nodes to calculate their trust values.

Trust computation module mainly is responsible for the computation and integration of multiple decision factors. Here multiple decision attributes include four aspects, 1) "node direct trust value", the calculation of node direct trust value based on F->AHM, 2) "node recommendation trust value", the calculation of node recommendation trust value based on neighbor recommendation algorithm, 3) " node reward and penalty function " , the calculation of node reward and penalty function based on the success rate of interaction and the number of attacks, 4) "node historical trust value", the calculation of node historical trust value based on the trust storage of neighbor list. Multiple decision attributes are solved through F->AHM. Finally, the node trust value is calculated by weighted product and the path trust value is obtained by "short board principle". As shown in Figure 2.



Node forwarding rate
Node energy availability degree
Node linkage sustainability degree
F->AHM
Node direct trust value
Trust recommendation algorithm
Node recommendation trust value
The cumulative count of node normal and abnormal interaction
Node reward and penalty function
Node historical trust value
F->AHM
Node trust value
Path trust value
Trust propagation algorithm
The last updating value

Fig. 2 Trust computing module

Trust decision module completes the judgment and management of node trust. According to the trust decision policy, it can judge and detect unreliable nodes.

Routing protocol module achieves route discovery and route selection on the basis of the results of trust decision module and avoids unreliable nodes and paths for establishment of secure routing.

B. *Calculation of multiple decision factors (DF)*

In the proposed trust management framework, node trust includes four decision attributes. The definition of each decision attributes are shown in Section 2.1. Then we will give the calculation method of four decision attributes.

(1) According to the characteristics of MANET, three trust factors are chosen to compute node direct trust, namely, "node's energy availability degree", "node's forwarding rate" and "node's survival time availability degree ". These trust factors are defined as follows:

❶ *Factor of node energy availability degree NEAD* $_{i,j}$. It is defined as the ratio of the residual energy E_a to initial energy E_{init}, that is $NEUD=E_a/E_{init}$.

❷ *Factor of node forwarding rate NFR* $_{i,j}$. It is defined as the ratio of the total number of node's forwarding packets DN_f to those of node's receiving packets DN_r, that is $NFR=DN_f/DN_r$.

❸ *Factor of node linkage sustainability degree NNLSD* $_{i,j}$. It is defined as the ratio of the average linkage sustainability time to the time window *Ts*. The $NNLSD_{i,j}$ during the period *Ts* is calculated by the following equation：

$$NNLSD_{i,j} = \frac{(\sum_{t=0}^{t=Ts} \delta_{i,j})/k}{Ts} \qquad （1）$$

If two nodes enter each other's wireless transmission range $\delta_{i,j}=1$, else $\delta_{i,j}=0$. And *Ts* represents the time window (it also equals the updating period of node trust value), *k* represents the on-off times of neighbor nodes. The larger the value *k* is, the smaller the value of $NNLSD_{i,j}$ is. For example, in Fig.3, the on-off times of node *i* and node *j* are *k* (*k*=3) times during the period *Ts*, the $NNLSD_{i,j}$ is:

$$NNLSD_{i,j}=\frac{(\sum_{t=0}^{t=Ts}\delta_{i,j})/k}{Ts}=\frac{(T_2-T_1+T_4-T_3+T_6-T_5)/3}{Ts} \qquad (2)$$

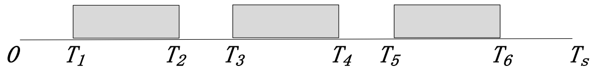


Fig. 3 Average encounter time

(2) Node recommendation trust value is defined as the trust value which is recommended by the third entity. But the restrictive condition is that the third entity must have a direct communication relationship with the two nodes X_i and X_j, besides, node X_i and node X_j have a direct communication (or node X_i and node X_j are neighbors) Thus, the third entity may be a node or a link which consisted of several nodes. In the topology, we regard a node as a special form of link. The topology is shown in as follows.

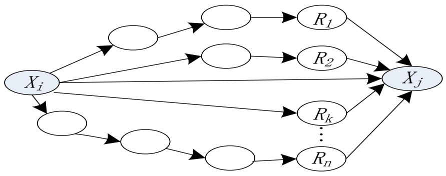


Fig.4 Recommendation trust topology

Assume that there are *n* recommendation paths, which form a set {R_1，R_2...R_n}，$T_1(R_k, X_j)$ represents the direct trust value of the *kth* recommender R_k to the node X_j. $T_2(X_i, X_j)$ represents the recommendation trust of node X_i to the node X_j. So $T_2(X_i, X_j)$ can be defined as follows.

$$\begin{cases} T_2(X_i,X_j)=\begin{cases}\dfrac{\sum\limits_{k=1}^{n}\left(\lambda_k\times T_1(R_k,X_j)\right)}{\sum\limits_{k=1}^{n}\lambda_k} & n>0\\ 0 & n=0\end{cases} \\ \lambda_{\text{k}}=\begin{cases}0 & L=0\\ T_1\left(X_i,R_k\right) & L=1\end{cases}\end{cases} \qquad (3)$$

Where λ_k is defined as the recommendation weighting factor for different recommendation paths and *L* is defined as the hop-counts from the assessment node X_i to the recommendation node R_k. Because the recommendation nodes R_k is always not the neighbors of assessment node X_i (such likeR_1 ,R_2 ,R_n), so different hop-counts in different recommendation paths leads to different recommendation trust values. Therefore, the rapid decrease of node trust value in the dot production transmission mode makes the recommendation trust from the recommendation node R_k very low. Therefore, we just consider the recommendation paths when *L* is equal to 1(only one node R_k *(1<k<n*) between node X_i and node X_j. When *L=0*, X_i and X_j are neighbors and the value of $T_2(X_i, X_j)$ is equal to 0.

(3) There is a large number of unreliable services in the open network environment，such as network deceptions，forgery behaviors，etc. The introduction of reward and penalty factors can reflect the incentive to normal behaviors and the punishment to malicious behaviors and effectively avoid the network attacks from the malicious nodes. The Reward and punishment factor function is shown as below.

$$T_3\left(X_i,X_j\right)=\frac{\alpha\times S\left(X_i,X_j\right)+\beta\times F\left(X_i,X_j\right)}{H_{total}},\ \alpha,\beta\in\left[0\ 1\right] \qquad (4)$$

Where $S(X_i, X_j)$ represents the cumulative count of normal interaction between node X_i and node X_j; $F(X_i, X_j)$ represents the cumulative count of abnormal interaction between node X_i and node X_j; H_{total} denotes the total count of node interaction, α and β is the reward and punishment factor and $\alpha+\beta=1$.

(4) Node historical trust value has the following characteristics:

I. The effects of node historical trust on node trust declines over time.

II. The closer the current moment is, the greater the effects of node historical trust on node trust is.

III. The historical trust value holds the Markov effect.

Because the trust value above is calculated in time stamp, the historical trust calculated in this time stamp has an important influence on the trust value calculated in next time stamp, so we put the trust value calculated in last time stamp as the trust factor of this trust calculation and give it some weight. The specific calculation is shown as follows.

$$T_4(X_i,X_j)=T'(X_i,X_j) \qquad (5)$$

Where $T'(X_i, X_j)$ is the trust value calculated in the last time interval. According to Markov effect, we can regard the node trust value calculated in the last time stamp as node historical trust value calculated in the time stamp.

C. Computation of node trust

The calculation of node trust value is related to the integration of four decision factors, namely the problem of the weight ratio of each decision factor. Traditional weighted average method or the geometric mean are more subjective and does not take into account the actual objective demand of each decision attribute. Aiming at this problem, Professor Chen proposed an attribute hierarchical model (AHM) to solve weighting problem.. However, there is a weak point that the AHM model needs to have a clear proportion evaluated by the experts, which is often not consistent with the actual situation. In the actual problem, decision-makers may not use an exact value m_{ij} on the comparison of each two criteria, but tending to express with the "neighborhood interval of m_{ij}". The triangular fuzzy number (l_{ij}, m_{ij}, u_{ij}) (m_{ij} represents the value of the maximum likelihood, l_{ij} and u_{ij} is the lower and upper limit of the fuzzy numbers) has a unique advantage about the expression of “neighborhood interval of m_{ij}”, it not only maintains the value interval of parameters, but also highlights the value of the maximum likelihood as well. Therefore, the triangular fuzzy attribute hierarchical model (F->AHM) [9] is introduced to calculate our weight, which makes the solution of multiple hierarchy weighting more practical.

The steps that F->AHM determines the decision attribute weight are as follows:

① As same as the general AHM model, the step begins with a detailed analysis of the trust assessment problem and then establishes the hierarchical structure of node trust in Fig.5.

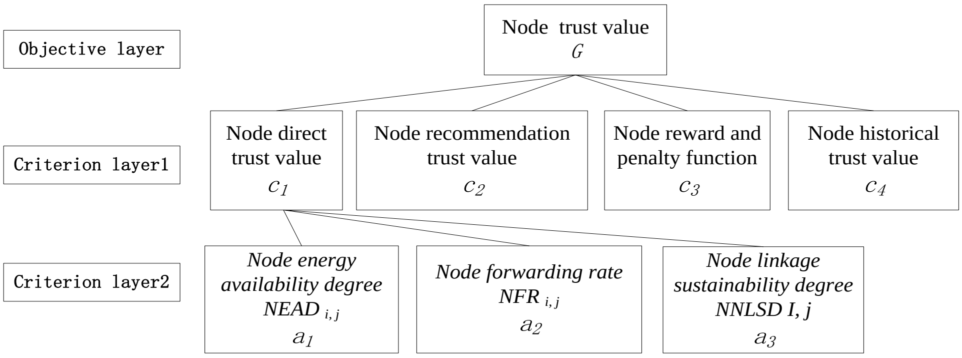

Objective layer
Node trust value
G
Criterion layer1
Node direct trust value
c1
Node recommendation trust value
c2
Node reward and penalty function
c3
Node historical trust value
c4
Criterion layer2
Node energy availability degree
NEAD i, j
a1
Node forwarding rate
NFR i, j
a2
Node linkage sustainability degree
NNLSD I, j
a3

Fig. 5 Gradual hierarchical structure of node tru

② In order to facilitate expert’s judgment, the comment sets in the form of language is given and it is transferred into the expert judgment matrix (Table 1) based on the triangular fuzzy number.

Table 1 Expert fuzzy attribute judgment matrixes

G	*c₁*	*c₂*	*c₃*	*c₄*
c₁	(0,0,0)	(0.9,1,1)	(0.9,1,1)	(0.9,1,1)
c₂	(0,0,0.1)	(0,0,0)	(0.5,0.7,0.9)	(0.3,0.5,0.7)
c₃	(0,0,0.1)	(0.1,0.3,0.5)	(0,0,0)	(0.1,0.3,0.5)
c₄	(0,0,0.1)	(0.3,0.5,0.7)	(0.5,0.7,0.9)	(0,0,0)

c₁	*a₁*	*a₂*	*a₃*
a₁	(0,0,0)	(0.9,1,1)	(0.9,1,1)
a₂	(0,0,0.1)	(0,0,0)	(0.5,0.7,0.9)
a₃	(0,0,0.1)	(0.1,0.3,0.5)	(0,0,0)

③ Gather the expert fuzzy attribute judgment matrixes of object set evaluated by different experts and calculate the final expert fuzzy attribute judgment matrix. Assume *L* experts participate in the evaluation of index sets, according to the formula as follow.

$$c_{ij}=\left(l_{ij},m_{ij},n_{ij}\right)=\frac{1}{L}\left(\sum_{k=1}^{L}l_{ij}^{k},\ \sum_{k=1}^{L}m_{ij}^{k},\ \sum_{k=1}^{L}n_{ij}^{k}\right) \quad (6)$$

$$a_{ij}=\left(l_{ij},m_{ij},n_{ij}\right)=\frac{1}{L}\left(\sum_{k=1}^{L}l_{ij}^{k},\ \sum_{k=1}^{L}m_{ij}^{k},\ \sum_{k=1}^{L}n_{ij}^{k}\right) \quad (7)$$

We can figure out the final expert fuzzy attribute judgment matrix of *G* and *c₁*:

$$\left[c_{ij}\right]_{4\times4}=\begin{array}{c|cccc|c} G & c_1 & c_2 & c_3 & c_4 & W_G \\ \hline c_1 & \left(l_{11},\ m_{11},\ n_{11}\right) & \left(l_{12},\ m_{12},\ n_{12}\right) & \left(l_{13},\ m_{13},\ n_{13}\right) & \left(l_{14},\ m_{14},\ n_{14}\right) & W_c\left(1\right) \\ c_2 & \left(l_{21},\ m_{11},\ n_{11}\right) & \left(l_{22},\ m_{22},\ n_{22}\right) & \left(l_{13},\ m_{13},\ n_{13}\right) & \left(l_{24},\ m_{24},\ n_{24}\right) & W_c\left(2\right) \\ c_3 & \left(l_{31},\ m_{31},\ n_{31}\right) & \left(l_{32},\ m_{32},\ n_{32}\right) & \left(l_{33},\ m_{33},\ n_{33}\right) & \left(l_{34},\ m_{34},\ n_{34}\right) & W_c\left(3\right) \\ c_4 & \left(l_{41},\ m_{11},\ n_{11}\right) & \left(l_{42},\ m_{42},\ n_{42}\right) & \left(l_{43},\ m_{43},\ n_{43}\right) & \left(l_{44},\ m_{44},\ n_{44}\right) & W_c\left(4\right) \end{array}$$

$$\left[a_{ij}\right]_{3\times3}=\begin{array}{c|ccc|c} c_1 & a_1 & a_2 & a_3 & W_{c_1} \\ \hline a_1 & \left(l_{11},\ m_{11},\ n_{11}\right) & \left(l_{12},\ m_{12},\ n_{12}\right) & \left(l_{13},\ m_{13},\ n_{13}\right) & W_a\left(1\right) \\ a_2 & \left(l_{21},\ m_{21},\ n_{21}\right) & \left(l_{22},\ m_{22},\ n_{22}\right) & \left(l_{23},\ m_{23},\ n_{23}\right) & W_a\left(2\right) \\ a_3 & \left(l_{31},\ m_{31},\ n_{31}\right) & \left(l_{32},\ m_{32},\ n_{32}\right) & \left(l_{33},\ m_{33},\ n_{33}\right) & W_a\left(3\right) \end{array}$$

④ Calculate the final weight. Firstly, according to the formula as follow.

$$W_c\left(i\right)=\left(\frac{\sum_{j=1}^{4}l_{ij}}{\sum_{i=1}^{4}\sum_{j=1}^{4}l_{ij}},\frac{2\sum_{j=1}^{4}m_{ij}}{4\left(4-1\right)},\frac{\sum_{j=1}^{4}n_{ij}}{\sum_{i=1}^{4}\sum_{j=1}^{4}n_{ij}}\right) \quad (8)$$

$$W_a\left(i\right)=\left(\frac{\sum_{j=1}^{3}l_{ij}}{\sum_{i=1}^{3}\sum_{j=1}^{3}l_{ij}},\frac{2\sum_{j=1}^{3}m_{ij}}{3\left(3-1\right)},\frac{\sum_{j=1}^{3}n_{ij}}{\sum_{i=1}^{3}\sum_{j=1}^{3}n_{ij}}\right) \quad (9)$$

we can calculate the relative fuzzy attribute weight vector $W_G=\left[W_c\left(1\right),W_c\left(2\right),W_c\left(3\right),W_c\left(4\right)\right]^T$, $W_{c1}=\left[W_a\left(1\right),W_a\left(2\right),W_a\left(3\right)\right]^T$,where T represents transpose. Then according to the formula as follow.

$$W(i)=\frac{1}{4}\left(\frac{\sum_{j=1}^{n} l_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} l_{ij}}+2\times\frac{2\sum_{j=1}^{n}\mu_{ij}}{n(n-1)}+\frac{\sum_{j=1}^{n} n_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} n_{ij}}\right) \qquad (10)$$

We also can calculate the relative non-fuzzy attribute weight vector. Finally, according to the formula as follow.

$$\overrightarrow{W(i)}=\frac{W(i)}{\sum_{i=1}^{n}W(i)} \qquad (11)$$

We obtain the normalization of the relative non-fuzzy attribute weight vector of G and c_1:

$$\overrightarrow{W}_G=\left[\overrightarrow{W_c(1)},\overrightarrow{W_c(2)},\overrightarrow{W_c(3)},\overrightarrow{W_c(4)}\right]^T$$

$$\overrightarrow{W}_{c1}=\overrightarrow{W_c(1)}=\left[\overrightarrow{W_a(1)},\overrightarrow{W_a(2)},\overrightarrow{W_a(3)}\right]^T$$

According to the weight $\overrightarrow{W}_{c1}$,we can obtain node direct trust value c_1 and finally we can get node trust value G by the weight $\overrightarrow{W}_G$.

Node trust value is defined as the dot product of the decision attribute weight and the decision factors, which is calculated by the following formula.

$$T(X_i,X_j)=\sum_{i=1}^{4}\left[\overrightarrow{W_G}\times T_i(X_i,X_j)\right] \qquad (12)$$

In each time interval period, node must calculate and update trust value once. In our trust model, trust values are limited in a continuous range from 0 to 1. The trust value of 0 signifies complete distrust whereas the value of 1 implies absolute trust. When there is no interaction between two nodes, the initial trust value is set to 0.55 (less trustworthy node). That is, we adopt a limited optimistic view on unknown nodes. A threshold ζ termed as the black-list trust threshold, is used to detect malicious nodes. In other words, if the trust value of a node is smaller than ζ , it will be regarded as a malicious node.

D. Computation of path trust

When a source node discovers a path to the destination node with the help of forwarding nodes, the trust value of the path should be computed according to the trust values of nodes along the path. According to the "cask principle", at time *t,* the trust of a path *P* denoted by $T_p(t)$ is equal to the minimum value of node trust value (except the source node and destination node) in the path, that is

$$T_p(t)=\min\left(\left\{T_{jk}(t)\mid n_j,n_k\in P\ and\ n_j\rightarrow n_k\ and\ n_k\neq N_d\right\}\right) \qquad (13)$$

in which，n_j and n_k are any two adjacent nodes among the path *P*，$n_{j\rightarrow}n_k$ means that n_k is the next-hop node of n_j , N_d is the destination node in the path *P.*。

In particular, because the destination node only receives data packets and its forwarding rate is equal to 0. So we regard the trust value as 1 in the destination node. Similarly，the source node only sends data packets and we also regard the value of the source node as 1.

III. TRUSTED MULTIPATH ROUTING PROTOCOL

On-demand multipath routing protocol (AOMDV) is currently the mainstream routing protocol in MANET network, which has many advantages. Based on AOMDV protocol，we will design a trust-based on-demand routing protocol (TAOMDV). The proposed protocol includes the node routing discovery algorithm and the source node routing selection algorithm.

1. Node routing discovery algorithm：

1. The source node broadcast an RREQ control packet to all its neighbor nodes.

2. At the same time, the timer update node historical trust value (*nb_history_trust)* and node current trust value (*nb_current_trust)* in node neighbors list, and when a new node is added, the value of node historical trust value and node current trust value are updated through node trust calculation function in section 2.3. The nodes that the updating time beyond the time stamp or its trust value is lower than the threshold ζ are immediately removed. This effectively avoids the existence of low trust nodes in the neighbor list and it also avoids the existence of low trust nodes in process of establishing path.

3. Neighbor nodes check the routing table entries to the destination and then check its freshness as well.

4. If the fresh routing table entries exist, the node replies an RREP. Otherwise, it continues to broadcast the RREQ to its neighbor nodes meanwhile establishes the reverse route and updates the path trust value (AT) in RREQ.

5. Until the destination node receives the RREQ packet, it can reply an RREP packet to the source node immediately, where the path trust value (AT) is added to the field

domain of RREP package and is initialized to 1. In the process of transmission to the source node, intermediate nodes update the value *AT* as follows.

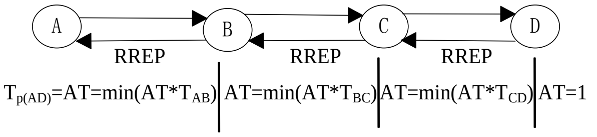


Fig. 6 Updating rule of path trust transmission

When receive (*rrep*) /*when a node receives an RREP*/
AT=min(*AT,T*(*index, nexthop*))
/**T*(*index, nexthop*) indicates the trust value of node *index* tonode *nexthop**/

6. When the updating value of *AT* reaches to the source node, it should be the minimum value of node in this path. According to "cask principle", it is also the final path trust value.

7. Source node receives multiple RREP packets and extracts the value of *AT* in RREP packet. Each value of AT has a corresponding trusted path and the value of AT represents the trust value of path. Finally, according to the trust value (AT), the hop count (HC) and the path trust threshold (RT), we can determine the path to send data packets.

2. Source node routing selection algorithm:

```
 Path1：Hop count：$HC_1$, Path trust value：$AT_1$
 Path2：Hop count：$HC_2$, Path trust value：$AT_2$
 RT：Path trust threshold
begin
 If ($AT_1$>=RT&&$AT_2$>=RT)
 {
     If (|$HC_1$-$HC_2$|<=3)
       {
         Using equation：$Bp=AT/(HC)^{3/2}$
         If ($Bp_1$>=$Bp_2$)   Choose Path1，
         else   Choose Path2；
        }
    Else
       {
        if (AT1>= AT2 )   Choose Path1
        else   Choose Path2；
       }
 }
Else if ($AT_1$<=RT||$AT_2$<=RT)
  Delete Path1 and. Choose Path2 or Delete Path2 and.
  Choose Path1
Else ($AT_1$<=RT&&$AT_2$<=RT)
Delete Path1 and Path2, Restart the routing discovery process
end.
```

IV. SIMULATION AND ANALYSIS

To evaluate the performance of AODV, TAODV, AOMDV and TAOMDV, we have conducted a comprehensive test using NS-2 network simulator.

A. Experiment setup

NS-2 simulator [12] is used to evaluate the performance of these on-demand routing protocols in different conditions. The simulation parameters in NS-2 are listed in Table 2

Table 2 Simulation parameter

Parameters	**Value**
Topology size	1000×1000m
Number of nodes	30
Packet transmission interval	30
Packet size	512byte
Malicious node variation	0-14
Traffic type	CBR
Path trust threshold *RT*	0.6
reward factor α	0.6
penalty factor β	0.4

We use four metrics to evaluate the performance of the routing protocols, which are the most important for best effort route and transmit protocols in Ad hoc networks.

1 Average End-to-end Latency：The average time taken by the data packets from sources to destination. It includes buffer delays during a route discovery, queuing delays at interface queues, retransmission delays at MAC layer and propagation time.

2 Packet Delivery Ratio：The proportion of the data packets delivered to destination nodes to those sent by source nodes.

3 Routing Packet Overload：The ratio of the number of

control packets including route request/reply/update/error packets) to the number of data packets.

4 Routing setup time：The time taken by the source node broadcasting the first RREQ packet to the source node receiving the first RREP packet.

B. Simulation results

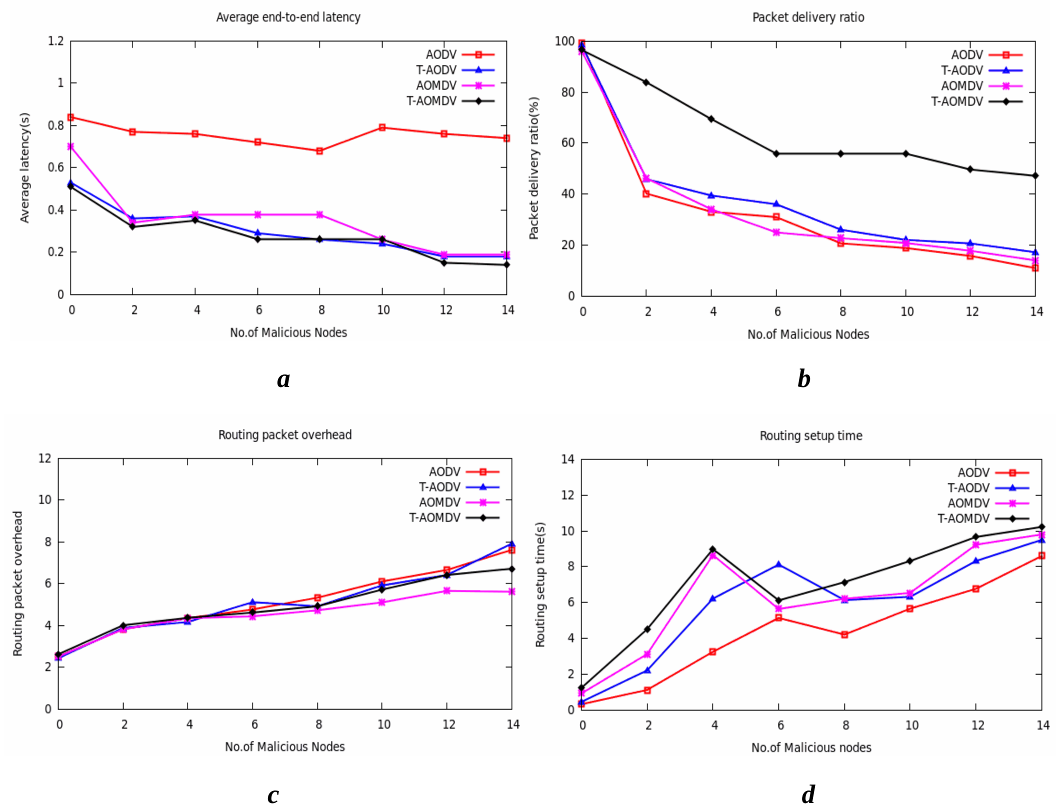


Fig. 7 Performance with a varying number of malicious nodes

a Average delivery ratio *b* Packet delivery ratio

c Routing Packet Overhead *d* Routing setup time

we evaluate these protocols by varying number of malicious nodes. As shown in Fig. 7*a*, the average latency of both the two protocols decline slowly with the increasing of the number of malicious nodes. There is only a tiny distinction between the average latency of T-AOMDV and that of AOMDV. This is because the availability of alternative routes reduces the delay caused by route rediscoveries. These multiple candidates contribute to reduce the end-to-end latency to a great extent. Besides, the trust mechanism can detect malicious nodes and thus average end-to-end latency is improved. In Fig.7*b*, the delivery ratios in both the protocols degrade sharply as the number of malicious nodes increases. The delivery ratio of T-AOMDV drops from 99% to 51% as the number of malicious nodes varies from 0 to 14. Less packet delivery ratio means less network throughput. Malicious nodes essentially limit interaction between nodes in the network. However, in T-AOMDV, intermediate nodes have several routes to destination so that when detecting grey hole or black hole attack, they can try alternative routes to forward packets and thus packet delivery ratio is improved. In Fig.7*c*, When the number of malicious nodes increases to 14(47% of the whole nodes), the routing packet overhead of T-AOMDV is approximately 6.7. Normally, T-AOMDV generates about 4.9 control packets on average for every data packet whereas AOMDV creates about 4.45 control packets. The increased control packets in T-AOMDV are primarily because of its route discovery mechanism that broadcasts more RREP packets to look for trustworthy routes to destination. In Fig.7*d*, the TAODV and TAOMDV need to judge the neighbor node whether to be trusted or not. It can consume several time. When the number of malicious nodes increases to 14, the routing setup time of AOMDV and TAOMDV are approximately 10s.

The simulation results above shows that this scheme maintains the packet delivery ratio better and avoids malicious nodes attacks effectively through adaptive selection of safe and reliable routing.

V. CONCLUSIONS

The trusted multipath routing algorithm with multiple decision attributes based on the triangular fuzzy number is introduced in this paper and it emphasizes the influence of multiple decision attributes on routing trust and models for each attribute of node trust relationship, determines the attribute weight factor through the F->AHM and establishes the routing trust model in the end. In addition, if the trust value changes in the trusted interval and even if its variation is very small, it will not affect the determination of black-list trust threshold. However, the trust value changes near the demarcation point of trustworthy interval and untrustworthy interval, if the range of variation is too small, the black-list trust threshold will be very difficult to determine. Therefore, how to avoid this problem will be a great challenge.

VI. ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China(61373172，61173088); The Fundamental Research Funds for the Central Universities(No. K50511010009); and The Foundation of Kunshan Innovation Institute of Xidian Univ. (2013).

REFERENCES

[1] Dey, H., Datta, R. A Threshold Cryptography Based Authentication Scheme for Mobile Ad-hoc Network. 1st International Conference on Computer Science and Information Technology. Vol.132.pp 400-409 (2011)

[2] Eissa, T. Abd Razak, S. Ngadi, MA . A Novel Lightweight Authentication Scheme for Mobile Ad-hoc Network. Arabian Journal for Science and Engineering.
Vol.37.pp 2170-2192. (2011).

[3] Jian W, Yanheng L. Yu J. Building a trusted route in a mobile ad hoc network considering communication reliability and path length. :Journal of Network and Computer Applications 34 1138–1149. (2011)

[4] Zouridaki C.E-Hermes：A Robust Cooperative Trust Establishment Scheme for Mobile Ad Hoc Networks[J]. : Ad Hoc Networks,7(6):1156-1168. (2009)

[5] Narula, P .Dhurandher, SK . Misra, S .Woungang, I
.Message security in mobile ad-hoc networks: Using trust-based multi-path routing approach. International Conference on Computer Engineering and Systems. pp.XXIV-XXIX.(2007)

[6]Qin.ZW. Jia.ZP. Chen.XH. Fuzzy Dynamic Programming based Trusted Routing Decision in Mobile Ad Hoc Networks. 5th IEEE international Symposium on Embedded Computing pp.180(2008)

[7]Li, XQ ; Lyu, MR ; Liu, JC A trust model based routing protocol for secure ad hoc networks. 2004 IEEE AEROSPACE CONFERENCE PROCEEDINGS, VOLS 1-6 Pages: 1286-1295 (2004)

[8] Li Xiaoqing, Li Hui, Yang Kai, and Ma Jianfeng. A Secure Routing Protocol Based on D-S Evidence Theory in Ad Hoc Networks [J]. Journal of Computer Research and Development. 48(8):1406-1413，(2011)(in Chinese)..

[9] Su Shi bin, Huang Rui hua, Attribute Hierarchical Mode based on Triangular Fuzzy Number [J]. :Systems Engineering- Theory &Practice. (2006) (in Chinese).