Received: November 14, 2016

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel QoS Trust Computation in MANETs Using Fuzzy Petri Nets

Nageswara Rao Sirisala [1]* **C. Shoba Bindu[2]**

[1]*Vardhaman College of Engineering, Hyderabad, Andhra Pradesh, India*
[2]*Jawaharlal Nehru Technological University, Anantapur, Andhra Pradesh, India*
* Corresponding author's Email: nagsirisala@gmail.com

Abstract: Evaluation of node's trust value is certainly advantage in mobile Adhoc networks (MANETs) where the applications run efficiently by involving trustable nodes only. The proposed method "A Novel QoS Trust computation in MANETs using Fuzzy Petri Nets-QTFPN", evaluates node trust value based on its quality of service (QoS) parameters. Here the MANET is represented as Dynamic Adaptive Fuzzy Petri Nets (DAFPN) model with concurrent reasoning algorithm (CRA). In which delivery of each packet from node to node requires evaluation of certainty factor (μ) using fuzzy expert system. This fuzzy inference system uses QoS parameters as fuzzy input variables namely energy, bandwidth, node mobility and reliability. In the routing process the intermediate node's trust values are evaluated based on certainty factor. The concurrent reasoning algorithm can strengthen the proposed method in selection of quality path to destination and reestablishment of path in case of path breaks. The proposed method performance is analyzed theoretically in terms of time and space complexities. The simulation results are taken against node velocity and network size, where the proposed method outperforms the existing protocols.

Keywords: Quality of service, Trustworthiness, Reasoning algorithm, Fuzzy petri nets.

1. Introduction

QoS provisioning is one of the advanced and challenging area in the MANETs. Due to node mobilites and lack of administration, it is not trivial task to establish the route to destination node with the potential intermediate nodes, in terms of quality resources. In multi constrained QoS provision it is hard to define the priority levels among the multiple quality parameters, which is influenced by network conditions. In this paper, the problem is overcome by using fuzzy rule base in aggregation of QoS parameters. Where the fuzzy rules are inferred based on network conditions.

QoS trust of a node in MANET represents how much it is dependable in quality wise. In the literature many people presented different definitions to trust [1,2,3]. Trust is having the context based meaning. In the MANET environment it can be defined as[4], "trust reflects the belief or confidence or expectations on the honesty, integrity, ability, availability and quality of service of target node's future activity/behaviour". Here QoS trust is derived by aggregating quality parameters like bandwidth, energy, Link Expiry Time (LET) and Reliability using fuzzy inference mechanism.

The proposed method uses the Dynamic Adaptive Fuzzy Petri Net (DAFPN) with concurrent reasoning algorithm. DAFPN is a expert system to represent, capture and store fuzzy knowledge with the help of parameters such as threshold value, certainty factor and weight. The concurrent reasoning algorithm (CRA) is a matrix operations based algorithm, which can automate the working procedure of DAFPN.

In this paper, the MANET is represented as DAFPN. In the proposed routing protocol, source node initiates the routing process by sending Route Request (RREQ) packets towards destination node for path establishment. The destination node gathers the topological information through these RREQ packets and runs the CRA to find the quality and trustworthy route. It intimates the path information to source node through Route Reply (RREP) packet.

Motivation: The existing protocols tried to achieve QoS through trust, but did not handle these parameters separately. We addressed this problem by considering node's competency (quality) and reliability (soft security) in its trust computation.

In case of path failure, the existing routing protocols are spending considerable amount of time in route recovery phase. But with the help of concurrent reasoning algorithm, the proposed method can select the alternate quality path immediately without initiating path finding process

Contribution: we used fuzzy inference mechanism to aggregate quality parameters to define node trust value. We modelled MANET topology as DAFPN to apply FPN rules. We introduced the route finding and recovery mechanisms using CRA in unicast and multicast methods.

Advantage of QTFPN: The proposed method is having the following merits

1) The method attains good throughput and packet delivery time, since it selects the intermediate nodes with sufficient energy and bandwidth.
2) It deploys the stable nodes along the path, so could measure less number of path breaks.
3) Since node's attitude is considered in trust evaluation, data can be transferred in secure environment.

The further sections in this paper are organized as follows. In section 2, the related wok is discussed. In section 3, DAFPN properties and rules are discussed. Evaluation of QoS parameters is explained. In section 4, certainty factor (trust value) is evaluated using fuzzy inference system. CRA algorithm is applied for route finding of unicast and multicast. Performance of QTFPN is measured theoretically. In section 5, simulation results are explained. In section 6, conclusion and future scope of proposed method is explained.

Note: In this work, the term Trust refers QoS Trust.

2. Related work

In this section, we discussed recent papers and their drawbacks about QoS and Trust issues in MANETs. The proposed work is compared with exiting methods.

Hui Xia et al.[5] proposed node trust as the combination of it's historical and current trust values. Where historical trust is evaluated based on packet forwarding ratio(FR), ie number of total packets(N_{all}) a node received and number of packets it forwarded correctly(N_{cor}). Node current trust is evaluated using fuzzy logic, where the QoS parameters like battery power, local memory, CPU cycle and bandwidth are considered fuzzy input variables. But it did not give proper explanation of evaluation of QoS parameters.

In the model proposed by Sridhar et al. [6], QoS is achieved through the trust in MANETs. Due to misbehaviour nodes, the QoS parameters are affected. Here the QoS parameter is node residual energy. Due to broadcast nature of wireless medium, a node can observe neighbour node activities like how many nodes it is forwarding correctly among the total received packets. Here a node trust value is calculated based on packet forwarding ratio of RREQ (Route Request), RREP (Route Reply) and Data packets. Nodes having less than threshold trust value are considered as misbehaviour nodes. But the paper not presented route recovery mechanisms.

In the model proposed by BoWang et al.[7], trust is estimated through the direct interactions and recommendations of neighbour nodes. Link delay is considered as QoS parameter. Link delay is calculated using ETX (expected transmission count) method. In ETX every node sends probe packets in regular intervals. The Link delay between two nodes is calculated based on loss of probe packets between them. But here the probe packets increased the control overhead.

Zafar Sherin et al.[8] proposed QoS trust by applying meta heuristic genetic algorithm. In the first phase the genetic algorithm is used to provide QoS by identifying qualified route to destination route. In second phase, trust is ensured for the packet reaching to destination. Every intermediate node adds some trust weigh to the packet while forwarding towards destination. The destination node can find whether the packet is come across through the trust worthy path based on trust weigh values in the packet. But the usage of genetic algorithm made it as complex work.

Ing-Ray Chen et al.[9] proposed trust as a composite metric like SQTrust, which is the combination of Social trust and QoS trust. Social trust is evaluated based on social relationships among the entities like honesty, friendship, privacy and similarity. QoS Trust is evaluated based on node's competence, cooperation and reliability. But majorly it focused on theoretical work and done less discussion on implementation.

Fei Hao, et al. [10] proposed context based trust (MobiFuzzyTrust) between two nodes. Here the MobiFuzzyTrust is evaluated in three stages. In first stage, the similarity for a pair of nodes is measured in different contexts like prestige-based, familiarity-based and similarity-based (location wise and time wise). In second stage, the measured similarity is converted into trust value using trust models. In third stage this crisp trust value is mapped to fuzzy

linguistic terms using fuzzy membership functions. But it taken many assumptions while applying the concepts to MANETs.

Shoba et al. [11] proposed a weightage based trusted QoS protocol. Here bandwidth is considered as quality parameter. This protocol fixed the weightages of data and control packets forward ratios in trust computation. Hence it is not able to support dynamic network changes.

Antesar et al. [12] proposed recommendations based trust. If the target node is not in direct interaction, then the trust recommendations of neighbour nods about target node are considered. But some malicious nodes may give false recommendations. Here the clustering technique is used to reduce the effect of bias recommendations. It verifies three points over the recommender node. i) it's level of confident, ii) deviation of its trust value from threshold value iii) closeness centrality value with evaluator node. Here trust is proposed for security purpose, didn't discuss any QoS issues.

Unlike the existing protocols the proposed method did not use any additional control packets. It established trust with the conventional packets only. QTFPN not used any fixed weightages in trust computations so it is adaptable to dynamic network changes in MANETs. With the help of CRA, the proposed method is having good route recovery mechanism when compared with other protocols. Due to lot of assumptions, some of the existing methods lost their practicality; we didn't take any assumptions while mapping DAFPN to MANETs. Some protocols used complex methods in trust computation, but we used light weight trust computation methods based on node interactions with other nodes.

3. System model

In this section we discussed DAFPN working rules and evaluation of QoS parameters, which helps the reader in understanding the proposed method.

3.1 Definition of DAFPNs

As discussed in the paper [13], the Fuzzy Petri Net (FPN) is a graph and its structure follows a bipartite directed graph. It is the combination of two types of entities namely places and transitions, where places are represented by circles and transitions are represented by bars. Places may or may not have tokens associated with degree of trust-DoT (i.e trust) values in the range of [0,1]. Directed arcs connect the input places to transitions and transitions to output places. Every transition contains a certainty factor (μ) value which lies in [0,1] and Threshold value (τ). A transition fires if $\mu > \tau$. After firing the transition, token is transferred from input place to output place and DoT of output place is evaluated. In [14], DAFPN is explained as the extension of FPN to handle the knowledge based system with dynamic nature. A DAFPN structure is defined with 11 parameters.

$DFPN(P;T;I;O;D;\alpha;\beta;W;U;Th;M)$ Here

- $P = \{p_1, p_2, \ldots . p_m\}$ Represents a set of m places
- $T = \{t_1, t_2 \ldots . t_n\}$ Represent a set of n transitions.
- $I : P \times T \rightarrow [0,1]$ is an input matrix with order $m \times n$. If place p_i is having directed edge to the transition t_j then element of I matrix, $I_{ij} = 1$ otherwise $I_{ij} = 0$.
- $O: T \times P \rightarrow [0,1]$ is an output matrix with order $m \times n$. If transition t_j is having directed edge to place p_i then $O_{ij} = 1$ otherwise $O_{ij} = 0$.
- $D = \{d_1, d_2, \ldots d_m\}$ Represents a set of propositions.
- $\propto: P \rightarrow [0,1]$ is a function which maps the places to real values [0 1].
- $\beta: P \rightarrow D$ is a function which maps the places to propositions.
- $W: P \times T \rightarrow [0,1]$ is an input function and is represented as a $m \times n$ dimensional matrix. In the matrix an entry value $w_{ij} \in [0,1]$ is the weight associated with input place. For one transition the sum of weights for all input places is 1.
- $U: T \times P \rightarrow [0,1]$ is an output function and is represented as an $m \times n$ dimensional matrix. An entry value in matrix U, $\mu_{ij} \in [0,1]$ is the value of certainty factor (μ) defining how much a transitiont_j can influence its output places p_i.
- $Th: O \rightarrow [0,1]$ is an output function and is represented as an $m \times n$ matrix, an entry in the matrix $\tau_{ij} \in [0,1]$indicates the output threshold of the place p_i from transition t_j. $\tau_{ij} = \infty$, if there is no edge.
- M is the dynamic input and directly influences the dynamic behaviour of DAFPN. $M = (\propto (p_1), \propto (p_2), \ldots, \propto (p_m))^T$ The initial marking is denoted by M_0.

3.2 Weighted fuzzy production rules of DAFPN

The fuzzy production rules of DAFPN are explained in [14]. In those rules, the antecedent part is represented by input places and consequent part is represented by output place. The rule is applied on firing of transition between input and output places.

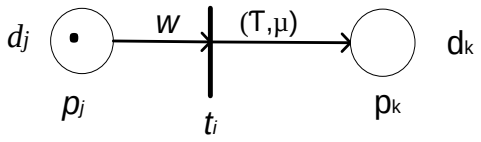

Figure.1 DAFPN type 1 rule

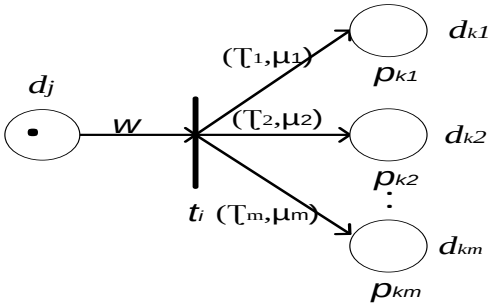

Figure.2 DAFPN type 2 rule

Here we discussed some of the rules, those are necessary to understand the proposed method.

In Figure 1, the form of type-1 rule is IF d_j THEN d_k (w, τ, μ). Here the degree of trust of place p_j is d_j i.e. $\propto (p_j) = d_j$. After firing the transition t_i (if $\mu > \tau$), the token is copied from place p_j to p_k. Then the degree of trust for place p_k is evaluated as $d_k = d_j \times w \times \mu$.

In Figure 2, the form of type-2 is IF d_j THEN d_{k1} AND d_{k2} AND ... AND d_{km}. Here the transition has multiple outputs; trust of every output is evaluated as $d_{ki} = d_j \times w \times \mu_i$.

3.3 Evaluation of QoS parameters

Here we discussed the evaluation of quality (energy, bandwidth and link expiry time) and reliability parameters.

3.3.1 Node energy calculation

In forwarding a data packet, a node has to receive and transmit it to next hop node. In the papers [15], energy is calculated as show in eq(1).

$$E_{total} = 2 \times E_{act} \times k + E_{amp} \times r^2 \times k \quad (1)$$

Here E_{act} is transmitter/receiver activation energy. An amplifier requires $E_{amp} \times r^2$ amount of energy to transfer K-bit data over r distance.

3.3.2 Link band width

According to TDMA, the link bandwidth for a pair of nodes is defined based on its common free transmitting /receiving time slots. In [16, 17], link bandwidth evaluation using TDMA is explained.

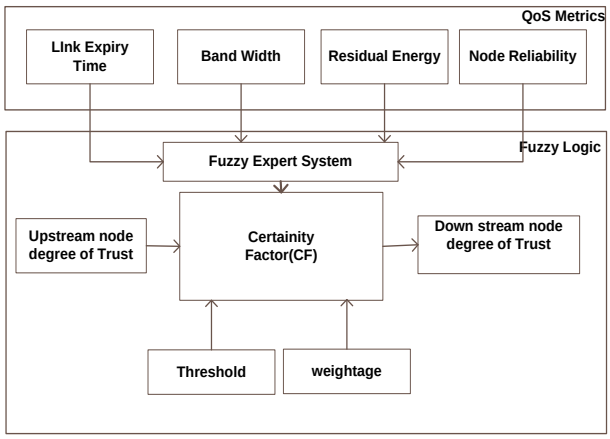

Figure.3 Framework of the proposed scheme

3.3.3. Link expiry time (LET)

In [18], the link expiry time for a pair of nodes is calculated based on their velocities and moving directions. Let A and B are neighbor nodes with the distance d . (x_1, y_1), (x_2, y_2) are present locations of A and B. S_1 and S_2 are their speeds. $\theta_1$and θ_2 are their directions of movement. Then the link expiry time between A and B is computed in eq(2).

$$LET = \frac{-(pq+rt) + \sqrt{(p^2+r^2)d^2 - (pt-rq)^2}}{(p^2+r^2)} \quad (2)$$

$p = s_1 \cos\theta_1 - s_2 \cos\theta_2$, $q = x_1 - x_2$,
$r = s_1 \sin\theta_1 - s_2 \sin\theta_2$ and $t = y_1 - y_2$.

3.3.4 Estimation of node reliability

A node can assess the neighbor node reliability based on the number of packets it received and forwarded correctly. In the paper [19], a node reliability (r) is estimated as a random variable using Bayesian inference theory and the value lies between [0, 1]. Lets a node has forwarded a number of packets correctly among the b number of received packets then expectation of reliability is like in eq(3)

$$E[r] = \frac{\alpha_n}{\alpha_n + \beta_n} \quad (3)$$

Here $\alpha_n = \alpha_{n-1} + a_{n-1}$, $\beta_n = \beta_{n-1} + b_{n-1} - a_{n-1}$
$\alpha_0 = \beta_0 = 0$

4. Novel QoS trust computation in MANETs using fuzzy petri nets

In Figure 3, the framework of the proposed scheme is explained. The DAFPN uses the QoS metrics as fuzzy input variables in evaluation of certainty factor(μ). Based on threshold, weightage and upstream node's trust, a transition will be fired and the downstream node trust value is evaluated.

The QTFPN selects the nodes with high trust values along the path to destination.

In this section first basic elements like μ , τ are explained. Next using these elements unicast and multicast routing is explained with CRA algorithm over a example network.

4.1 Fuzzy based certainty factor (μ) evaluation

In Figure 4, the fuzzy inference system considers the parameters like bandwidth, reliability, node residual energy and LET as fuzzy input variables to evaluate fuzzy output variable i.e certainty factor (μ). If $\mu > \tau$, then the transition t_i is fired and the downstream node p_k degree of trust (DoT) is evaluated.

In MANET, the attenuation rate of a node's QoS parameters is linear in data transmission, so we used triangular membership functions to measure fuzzy input and out variables as shown in Figure 5. It follows four fuzzy sets like Very Low, Low, Medium and High. Fuzzy expert system follows two phases like fuzzyfication and defuzzification[20]. Here in the fuzzyfication process, all the QoS parameters are aggregated. In defuzzification, the certainty factor(μ) is calculated. Both phases works based on the fuzzy rule base[21] as shown in table 1. These fuzzy rules are framed based on network conditions and inferred from experimental results.

4.2. Threshold value

The threshold value (τ) is defined for every transition. In the proposed model for a pair of adjacent nodes, it is defined as a function of QoS resources attenuation rate. If the nodes are having high attenuation rate of their QoS resources, then the transition between them will have higher threshold value vice versa.

$$Th = f(\Delta E, \Delta BW, \Delta LET, \Delta R) \quad (4)$$

In eq (4), Δ is a attenuation rate of QoS parameters.

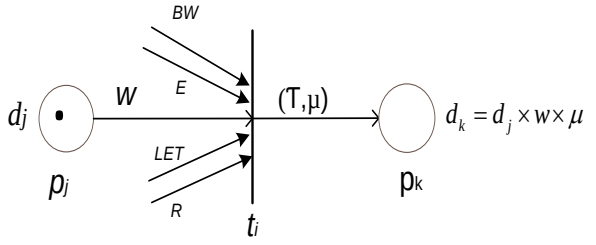

Figure.4 Certainty factor computation

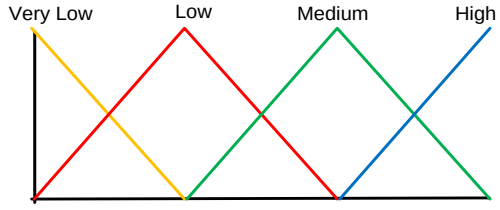

Figure.5 Fuzzy triangular membership function

Table 1. Fuzzy rule base

Band Width	Energy	Reliability	LET	Certainty Factor
High	High	High	High	High
Medium	High	High	Medium	Medium
Low	Low	Low	Medium	Low
Verry Low	Verry Low	Medium	Medium	Verry Low

4.3 DAFPN based routing algorithm

1) Whenever the Source node ns wants to share data with destination node, it sends the RREQ packets by setting its degree of trust value as $1, \propto (n_s) = 1$.
2) Since all transitions have single input, weightage of input is 1, i.e w=1.
3) On receiving RREQ packet, each intermediate node evaluates the Certainty Factor (μ). While forwarding the packet it adds its ID, W, Th and μ values.
4) On receiving multiple RREQ packets, a destination node can gather the topological information like μ, W, Th values for every link in the network.
5) The Destination node runs the concurrent reasoning algorithm and calculates the best route's trust value. By tracing it back, it can find the path to source node.
6) The Destination node sends the RREP packet in that path to source node. Once receiving RREP packet a source node establish the path and starts the data transfer.

4.4 Concurrent reasoning algorithm (CRA)

//Input: I, O, U, W, Th are the matrices (as defined in the section 3.1) with order m × n, where rows represents places(nodes) and columns represents transitions. M_0 is an initial marking vector(with initial DoT (i.e trust) of source as 1).

//Output: M_k is an output vector with final DoT values of all nodes.

{

For iteration k=1, and $M_0 = [1,0,0,0,0,0,0,0]$

1) Represent every transition with its input place DoT $T = M_0 \circ I$, Where ∘ represents normal matrix multiplication.
2) In the matrix $(T - Th)_{m \times n}$, find the positive entries and consider corresponding

position entries in U matrix, i.e U_{new} .

3) Find the new marking(trust of places)
   i) Compute $X = U_{new} \otimes T$, where $\otimes$ represents matrix multiplication operation. But instead of sum of elementary products, it will consider maximum of elementary products.
   ii) New marking $M_1 = M_0 \oplus X$, here $\oplus$ represents the maximum of two matrices.
4) If $M_1 = M_0$, then stop process, otherwise repeat the process for next iteration$(k+1)$

}

4.5 QTFPN routing with an example network

In Figure 6, the MANET topology is described with 8 nodes, where 1 and 8 are source and destination nodes. In figure 7, the network topology is converted as DAFPN, where nodes and links are represented as places and transitions respectively. Each output arc from transition is associated with certainty factor-μ, threshold-τ parameters. (τ, μ) values are taken based on simulation results for reader explanation purpose.

To find the route to destination node-8, a source node-1 sets the RREQ packet with its degree of trust(DoT) as 1 and sends the packet to neighbour nodes 2,3,4. On receiving of the RREQ packets, every node evaluates the QoS parameters and applies the fuzzy inference mechanism to estimate the certainty factor(μ) .Using eq(4) calculate the threshold value (τ). Here the (τ, μ) parameters for the neighbour nodes (2, 3, 4) are (0.3,0.8),(0.4,0.9) and (0.5,0.7) respectively. Every intermediate node adds the (τ, μ) values to the RREQ packet before forwarding to the next hop nodes. Through the RREQ packets, destination node-8 collects the network information and runs the concurrent reasoning algorithm. In CRA algorithm, destination node-8 estimates the DoT values for every node in the network and traces the route with the nodes with higher DoT values. Here the established path is 1-3-5-8.

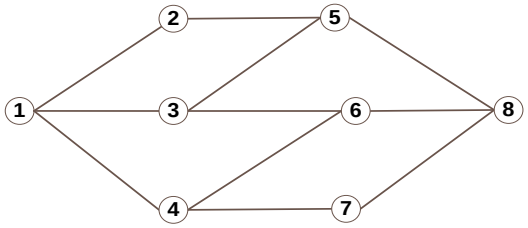

Figure.6 MANET network topology

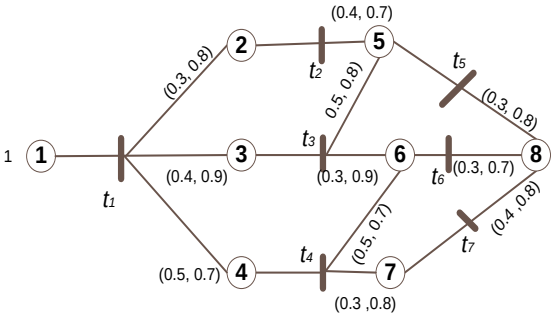

Figure.7 DAFPN modelled MANET topology

4.5.1 CRA algorithm of unicast routing

Here the CRA is explained over the example network shown in figure 7 and the steps followed as per section 4.4. Where in every matrix, rows represents nodes (1,2,3,4,5,6,7,8) and columns represents transitions $(t_1, t_2, t_3, t_4, t_5, t_6, t_7)$. Initially, source node DoT is 1 and for remaining nodes it is 0. i.e $M_0 = [1,0,0,0,0,0,0,0]$. Since every transition is having single input, matrix W is optional. The remaining matrices I, O, Th, U are defined as below.

$$I = \begin{pmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0\\ 0&0&1&0&0&0&0\\ 0&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ 0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0 \end{pmatrix} \quad O = \begin{pmatrix} 0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 0&1&1&0&0&0&0\\ 0&0&1&1&0&0&0\\ 0&0&0&1&0&0&0\\ 0&0&0&0&1&1&1 \end{pmatrix}$$

$$U = \begin{pmatrix} 0&0&0&0&0&0&0\\ 0.8&0&0&0&0&0&0\\ 0.9&0&0&0&0&0&0\\ 0.7&0&0&0&0&0&0\\ 0&0.7&0.8&0&0&0&0\\ 0&0&0.9&0.7&0&0&0\\ 0&0&0&0.8&0&0&0\\ 0&0&0&0&0.8&0.7&0.8 \end{pmatrix} \quad Th = \begin{pmatrix} \infty&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.3&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.4&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.5&\infty&\infty&\infty&\infty&\infty&\infty\\ \infty&0.4&0.5&\infty&\infty&\infty&\infty\\ \infty&\infty&0.3&0.5&\infty&\infty&\infty\\ \infty&\infty&\infty&0.3&\infty&\infty&\infty\\ \infty&\infty&\infty&\infty&0.3&0.3&0.4 \end{pmatrix}$$

Step-1:

$$T = M_0 o I = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \begin{pmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0\\ 0&0&1&0&0&0&0\\ 0&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ 0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0 \end{pmatrix}$$

$$T = [1\ 0\ 0\ 0\ 0\ 0\ 0\ ]$$

Step 2:

$$T - Th = \begin{pmatrix} 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0\\ 1&0&0&0&0&0&0 \end{pmatrix} - \begin{pmatrix} \infty&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.3&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.4&\infty&\infty&\infty&\infty&\infty&\infty\\ 0.5&\infty&\infty&\infty&\infty&\infty&\infty\\ \infty&0.4&0.5&\infty&\infty&\infty&\infty\\ \infty&\infty&0.3&0.5&\infty&\infty&\infty\\ \infty&\infty&\infty&0.3&\infty&\infty&\infty\\ \infty&\infty&\infty&\infty&0.3&0.3&0.4 \end{pmatrix}$$

$$U_{new} = \begin{bmatrix} 0&0&0&0&0&0&0\\ 0.8&0&0&0&0&0&0\\ 0.9&0&0&0&0&0&0\\ 0.7&0&0&0&0&0&0\\ 0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0 \end{bmatrix}$$

Step 3:

$$X = U_{new} \otimes T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.8 \\ 0.9 \\ 0.7 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

New marking

$M_1 = M_0 \oplus X = [1 \quad 0.8 \quad 0.9 \quad 0.7 \quad 0 \quad 0 \quad 0 \quad 0]$

Since $M_1 \neq M_0$ continue for next iteration.

$M_2 = (1 \quad 0.8 \quad 0.9 \quad 0.7 \quad 0.72 \quad 0.81 \quad 0.56 \quad 0)$

$M_3 = (1 \quad 0.8 \quad 0.9 \quad 0.7 \quad 0.72 \quad 0.81 \quad 0.56 \quad 0.576)$

$M_4 = (1 \quad 0.8 \quad 0.9 \quad 0.7 \quad 0.72 \quad 0.81 \quad 0.56 \quad 0.576)$

Since M3=M4 stop the process. Here M_4 represents the trust values of all the nodes. Destination node trust value 0.576, represents the best route trust value form source to it.

4.5.2 Trace out the route

1) For the destination node find the transition from which it gains the high degree of trust in the matrix(X).
2) For the identified transition find the corresponding input node in the input matrix(I).
3) For the input node, find the transition from which it gains the high degree of trust in the matrix(X).
4) Repeat this process till the source node is identified.

Here the best path from source to destination is 1-3-5-8, which has the trust value 0.576.

4.5.3 Route recovery process

The proposed method provides good route recovery system. In figure 6, if the link between node 3 and 5 is broken since node mobility, then the established path from source to destination (1-3-5-8) gets disconnected. The destination node traces out the path for next highest DoT value. In Figure 8, the alternative route is established through 1-3-6-8 with DoT value 0.567.

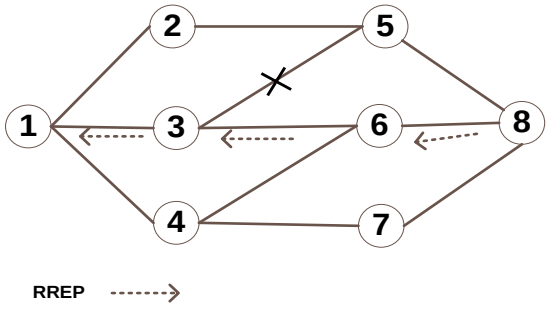

Figure.8 Route recovery using CRA

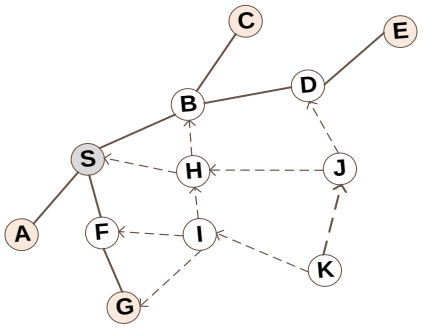

Figure.9 (a) Multicast tree

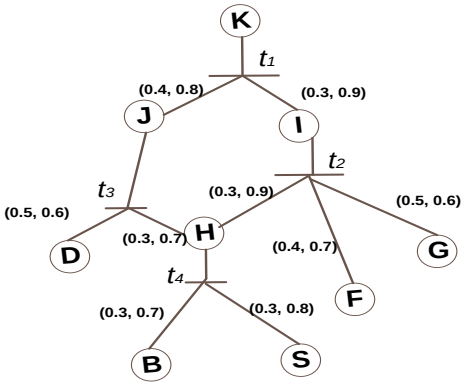

Figure.9 (b) DAFPN modelled multicasting routing

4.6 QTFPN extension of multicast routing

In multicast routing protocols [22,23], the source data is transferred to group of receiver nodes. Where the intermediate nodes should have enough quality resources to forward the packets to group of node.

In Figure 9(a), the multicast tree is formed, where node S is the source, nodes (A, C, E, G) are receiver nodes and nodes (B, D, F) are forwarding nodes. Node K wants to join the group and broadcasts the join request packets. On receiving join request packets, the group members S,B,D,F and G send reply packets(RREP) to the node K. Through the RREP packets, the node K collects the topological information of the group members. It constructs the FPN model as shown in Figure 9(b), and runs the CRA algorithm to find the group member node with highest DoT value to join the multicast group.

4.7 Concurrent reasoning algorithm of multicast routing

Here the algorithm is worked out for the multicast network in figure 9-b, where (τ, μ) values are taken based on simulation results. Here $M_O = [1,0,0,0,0,0,0,0,0]^T$. In the below matrices, rows represents [K, I, J, G, F, H, D, B, S] and columns represents(t_1, t_2, t_3, t_4).

$$I=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&0\\0&0&0&0\\0&0&0&1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\quad O=\begin{bmatrix}0&0&0&0\\1&0&0&0\\1&0&0&0\\0&1&0&0\\0&1&0&0\\0&1&1&0\\0&0&1&0\\0&0&0&1\\0&0&0&1\end{bmatrix}$$

$$U=\begin{bmatrix}0&0&0&0\\0.9&0&0&0\\0.8&0&0&0\\0&0.6&0&0\\0&0.7&0&0\\0&0.9&0.7&0\\0&0&0.6&0\\0&0&0&0.8\\0&0&0&0.7\end{bmatrix}\quad Th=\begin{bmatrix}0&0&0&0\\0.3&0&0&0\\0.4&0&0&0\\0&0.5&0&0\\0&0.4&0&0\\0&0.3&0.3&0\\0&0&0.5&0\\0&0&0&0.3\\0&0&0&0.3\end{bmatrix}$$

$$U_{new}=\begin{bmatrix}0&0&0&0\\0.9&0&0&0\\0.8&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$$

As per CRA algorithm in sec 4.5.1, the trust values of nodes are evaluated like,

$T=[1\quad 0\quad 0\quad 0]$

$M_1=[1,0.9,0.8,0,0,0,0,0,0]$

$M_2=[1,0.9,0.8,0.54,0.63,0.81,0.48,0,0]$

$M_3=[1,0.9,0.8,0.54,0.63,0.81,0.48,0.648,0.561]$

$M_4=[1,0.9,0.8,0.54,0.63,0.81,0.48,0.648,0.561]$

Here among the group members(S,B,D,F,G), B is having highest DoT value. The new node K joins the tree through the node B.

4.8 Time and space complexities of proposed method

The proposed method is the extension of AODV protocol with the additional work of Concurrent Reasoning Algorithm (CRA) but not using any additional packets in the routing. In the table 2, the time and space complexity of proposed method is described. In monitoring, each of m nodes gathers their q number of neighbor's information through HELLO packets, so it is the order of $O(m \times q)$.

Table 2. Performance of QTFPN

Action	Time Complexity	Space Complexity
Neighbour monitoring	$O(m \times q)$	It uses HELLO packets
QoS metric estimation	$O(m^2)$	No messages required
CF Estimation	$O(n \times f^i)$	No messages required
CRA	$O(m \times n \times k)$	$O(m \times n \times p$

Table 3. Simulation parameters

Simulation Tool	ns2.35
Simulation Area	1400×1400
Network Size	10-50
Node Velocity	10-60 m/sec
Simulation Time	600 sec
Transmission range	250 m

On receiving request packet a node evaluates the QoS parameters, so it is the order of $O(m^2)$. At every transition, fuzzy based μ is evaluated, it is the order of $O(n \times f^l)$, here n, f, l are number of transitions, fuzzy sets and QoS parameters. The time and space complexity of CRA algorithm is $O(m \times n \times k)$ and $O(m \times n \times p)$ respectively, where k is the number of iterations and p is the number of matrices.

5. Simulation results

The simulation is conducted in network simulator (ns2.35) for the metrics packet delivery time, throughput and packet delivery ratio. The results are taken by varying the node velocity (10-60 m/sec) and number of nodes (10-50). Simulation parameters are described in table 3. The performance of proposed method (QTFPN) is compared with the recent existing protocols ETQ[6] and TBQP [8].

Simulation metrics are defined as,

Packet delay: packet delay is the ratio of total time taken by all the packets to reach destination to the number of packets.

Throughput: It is defined as the amount of data delivered to destination in unit time.

Packet Delivery Ratio: It is the fraction of number of data packets reached to destination to the total number of packets generated.

In figure 10, the packet delivery time (Sec) is increased with network size. In case of route failures, the protocol has to deploy increased number of control packets in route recovery phase. This delayed packet delivery at destination side. Since the proposed method taken node velocities and movement directions into consideration, it can reduce the route failures. So the proposed method can reduce this delay over the existing methods ETQ and TBQP.

In the figure 11, throughput (Kbps) decreases with the increase in network size. If the network size is increased, a node bandwidth is shared with neighbour nodes, so a node bandwidth is decreased. Hence the throughput is decreased. The proposed method considered bandwidth as a QoS parameter, so it includes the intermediate nodes with threshold

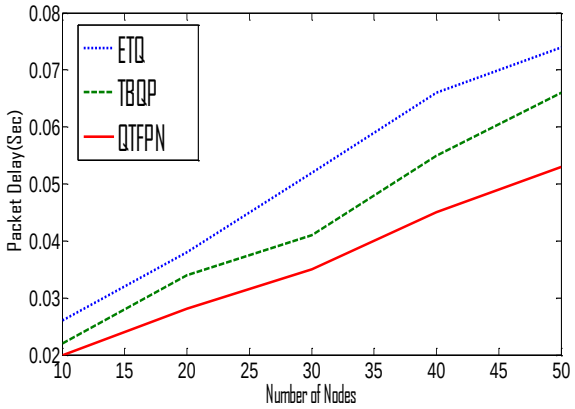

Figure.10 Network size Vs packet delay

level of bandwidth. So QTFPN could attain the good results over the existing method.

In figure 12, the packet delivery time (Sec) is increased at higher node velocities. If the nodes are moving with higher velocities, the links may get disconnected frequently. In case of route failures, data packets are accumulated at intermediate nodes and will be transferred after route recovery. This causes the delay in data delivery. The proposed method can quickly find the alternate path by

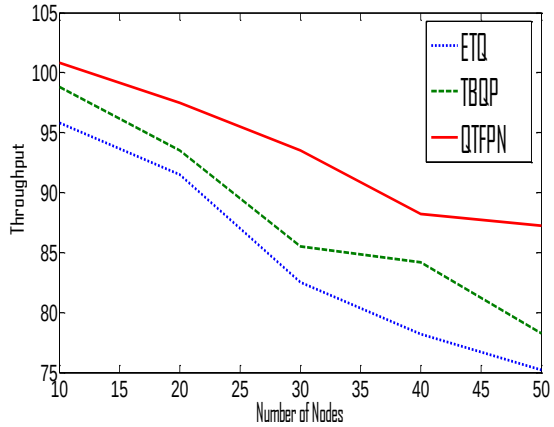

Figure.11 Network size Vs throughput

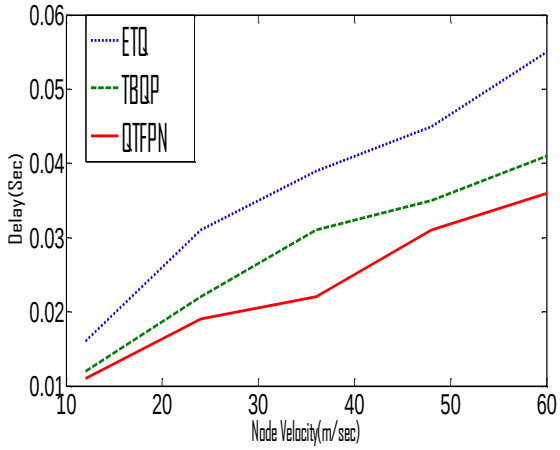

Figure.12 Node velocity Vs delay

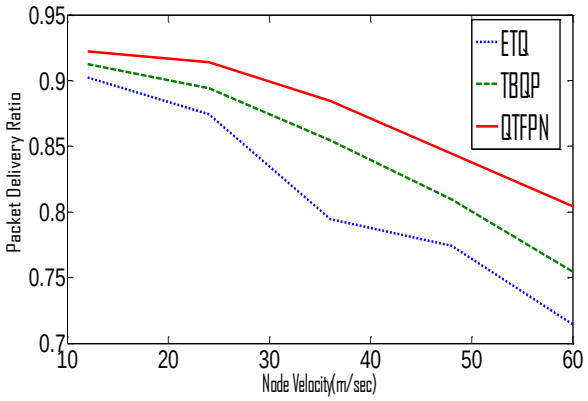

Figure.13 Node velocity Vs packet delivery ratio

running concurrent reason algorithm; hence it could measure less delay comparatively with existing protocols.

In Figure 13, the packet delivery ratio at the destination node is decreased at higher node velocities. If the intermediate nodes are not having sufficient energy and bandwidth, then PDR is decreased. The unreliable (malicious) nodes do unnecessary packet drops, which affects the PDR. The proposed method QTFPN evaluates the node trust in terms of energy, bandwidth and reliability. So it could measure the good PDR values than existing methods.

6. Conclusion and future work

The proposed method QTFPN includes the intermediate nodes which are trustable. The MANET is modelled as a DAFPN. In the QTFPN, the route establishment and recovery system is automated with help of concurrent reasoning algorithm. In computation of node trust value, the QoS parameters like energy, bandwidth, mobility and reliability are aggregated using fuzzy system. It is shown that the proposed method is also extendable for multicast routing. The improved performance of QTFPN is compared with the existing methods theoretically and practically (simulation). The MANETs and Social networks are related in their working principles. The proposed trust model can be applicable to social network in estimation of target node trust value before activities.

References

[1] H. Xia, and J. Yu, "Applying trust enhancements to reactive routing protocols in mobile ad hoc networks", *Journal of Wireless Networks*, Vol.22, pp. 2239–2257, 2016.

[2] M.S. Khan, and M.I. Khan,"MATF: a multi attribute trust frame workfor MANETs", *EURASIP Journal*

on Wireless Communications and Networking, Vol.16, pp.1-17, 2016.

[3] V. Jayalakshmi, and T.A. Razak, "Trust Based Power Aware Secure Source Routing Protocol using Fuzzy Logic for MANETs", *IAENG International Journal of Computer Science*, Vol.43, pp.98-107, 2016.

[4] K.Govindan, and P. Mohapatra, "Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey", *IEEE Communications Surveys & Tutorials*, Vol.14, No.2, pp.279-298, 2012.

[5] L. Ju, E.H. Sha, H.Xia and Z. Jia, "Trust prediction and trust based source routing in mobile ad hoc networks", *Ad Hoc Networks Elsevier*, Vol. 11, No. 7, pp. 2096-2114, 2013.

[6] S. Sridhar and B. Ramachandran, "Energy-and Trust-Based AODV for Quality of Service Affirmation in MANETs-ETQ", in *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*, Vol.324, pp. 601-607, 2014.

[7] B. Wang, X. Chen, and W. Chang, "A light-weight trust-based QoS routing algorithm for ad hoc networks", *Journal of Pervasive and Mobile Computing*, Vol.13, pp.164–180, 2014.

[8] S. Zafar, and M. K. Soni, "Trust based QoS protocol (TBQP) using meta heuristic genetic algorithm for optimizing and securing MANET", *International Conference on Optimization Reliabilty and Information Technology (lCROIT)*, pp. 173-177, 2014.

[9] I.R. Chen, J. Guo, F. Bao, and J.H. Cho, "Integrated Social and Quality of Service Trust Management of Mobile Groups in Ad Hoc Networks*", International Conference on Information Communications and Signal Processing (ICICS)*, pp.1-5, 2013.

[10] F. Hao, G. Min, M. Lin, and C. Luo, "MobiFuzzyTrust: An Efficient Fuzzy Trust Inference Mechanism in Mobile Social Networks", *IEEE Transactions on Parallel and Distributed Systems*, Vol.25, No.11, pp.2944-2955. 2014.

[11] C.S. Bindu, and S. Nageswararao, "Weightage based trusted QoS protocol in Mobile Adhoc Networks", *IEEE Global Conference on Wireless Computing and Networking* (GCWCN), pp.283-287, 2014.

[12] A.M. Shabut, K.P. Dahal, and S.K.. Bista, "Recommendation Based Trust Model with an Effective Defence Scheme for MANETs", *IEEE Transactions on Mobile Computing*, Vol.14, pp.2101-2115, 2015.

[13] Z. Hu, H. Ma, G. Wang, and L. Liao, "A reliable routing algorithm based on fuzzy Petri net in mobile ad hoc networks", *Journal of Central South University*, Vol.12, pp.714-719. 2005.

[14] H.C. Liu, and L. Liu, "Knowledge Acquisition and Representation Using Fuzzy Evidential Reasoning and Dynamic Adaptive Fuzzy Petri Nets", *IEEE Transactions On Cybernetics*, Vol. 43, No. 3, pp.1059-1072, 2013.

[15] P. Sethuraman, and N. Kennan, "Refined trust energy-ad hoc on demand distance vector (ReTE-AODV) routing algorithm for secured routing in MANET", *journal of wireless networks.* Vol.22, pp.1-11, 2016.

[16] C.C. Hu, H. Wu, and G.H. Chen, "Bandwidth-Satisfied Multicast Trees in MANETs", *IEEE Transactions On Mobile Computing*, Vol. 7, No. 6, pp.712-723, 2008.

[17] K.P. Shih, W.H. Liao, and Y.C. Tseng, "A TDMA-based bandwidth reservation protocol for QoS routing in a wireless mobile ad hoc Network", *IEEE International Conference on Communications*, pp. 3186-3190, 2002.

[18] N.C. Wang, and Y.F. Huang, "A Power-Aware Multicast Routing Protocol for Mobile AdHoc Networks With Mobility Prediction", *wireless personal communications*, Vol.43, pp.1479–1497, 2007.

[19] Z. Wei, and H. Tang, "Security Enhancements for Mobile Ad Hoc Networks With Trust Management Using Uncertain Reasoning", *IEEE Transactions on Vehicular Technology*, Vol.63, No.9, pp.4647-4658, 2014.

[20] T.C. Chiang, C.F.Tai, and T.W. Hou, "A knowledge-based inference multicast protocol using adaptive fuzzy Petri nets", *Elsevier Expert Systems with Applications*, Vol. 36, pp. 8115–8123, 2009.

[21] S. Nageswararao, and C.S. Bindu, "Uncertain Rule Based Fuzzy Logic QoS Trust Model in MANETs", *International Conference on Advanced Computing and Communications-ADCOM*, pp.55-60, 2015.

[22] H.Wu, and X. Jia, "QoS multicast routing by using multiple paths/trees in wireless ad hoc networks", *Elsevier Ad Hoc Networks*, Vol.5, pp. 600-612, 2007.

[23] S. Baolin, and L. Layuan, "On the reliability of MAODV in ad hoc networks", *IEEE international symposium on microwave, antenna, propagation and EMC technologies for wireless communications*, pp. 1514–1517, 2005.