Trust Based Random and Energy Efficient Routing (TRER) in MANET

S. Beski Prabaharan
Research Scholar, Department of Computer Science
Bharathiar University, Coimbatore, India.
E-mail: beskip@gmail.com

Dr. R. Ponusamy
Principal, Rajiv Gandhi Engineering College
Nemili, Sripurempudur, Chennai, India.
E-mail: r_ponusamy@hotmail.com

Abstract
Routing in MANET varies considerably from the other networks due to the fact that MANET, being an ad-hoc network does not follow a specific topology and the nodes are dynamic. Further, power consumption is another major aspect, which needs to be kept in check, as the depleted nodes tend to become selfish. This paper presents a metaheuristic based routing algorithm that generates routes dynamically, following the concept of equal load distribution in the network. The local search component of ACO is modified using Simulated Annealing to provide an effective and energy efficient node selection mechanism. Experiments show that the algorithm exhibits effective load distributions and also provides dynamic random paths.

Keywords: Dynamic routing; Ad-hoc networks; ACO; Trust based routing

INTRODUCTION
MANET (Mobile Ad-hoc NETworks) are a class of systems that require route generation dynamically, depending on the nodes that are currently in the transmission range. Hence it is mandatory for the routing protocols to generate dynamic routes for every transmission. Further, due to the power constrained nature of such networks, it becomes mandatory for the routing protocol to distribute the routes such that power utilization of nodes in the network is distributed, rather than getting concentrated on certain specific nodes alone. Continuous usage of certain nodes will lead to some nodes getting depleted and hence will start exhibiting selfish behavior.

A hybrid DTN-MANET routing method is proposed in [5]. This method is a specialization that is applied on dense and dynamic wireless networks. The protocol defined here uses DTN between disjoint group of nodes during the routing process. It is a decentralized architecture that uses the topological information exchange between nodes to form the routes. Adaptability proves to be the major advantage of this system. A QoS aware multicast routing algorithm is presented in [6]. It utilizes the greedy and the family competition approaches for speeding up the convergence and to integrate the diversity of population. A link stability based multicast routing technique is presented in [7]. This is a mesh based algorithm that maintains link stability in the network. Other mesh based algorithms include ODMRP [11], FGMP [12], CAMP [13], NSMP [14] and DCMP [15]. Several tree based routing techniques [8-10] that are proactive and provides effective multicasting in MANETs.

In the recent times, trust based routing methods have also gained prominence. Trust is usually calculated by past experience and observed actions [17]. OSLR based routing scheme that uses trust levels of nodes during the selection process is presented in [16]. It presents a trust reasoning model that is based on fuzzy Petri net to evaluate these trust values. Other trust reasoning models that use OSLR are presented in [18,19,20]. A cooperative routing strategy on the basis of trust and energy efficiency is presented in [23]. Dynamic trust and energy values are used to identify the nodes as selfish or altruistic. The traditional AODV algorithm is modified by incorporating trust and energy values. Effective routes eliminating selfish nodes were obtained.

Emergent intelligence or group intelligence based routing mechanism is described in [21]. A logical clustering is created and the agents interact with each other. Inter and intra cluster communications are facilitated to collect details about the nodes in the network. Though this method is helpful, it relies on several static agents for appropriate working, hence are suitable only for specific type of networks. A method working exclusively on the basis of power consumption is presented in [22]. This method is based solely on creating assurance networks, providing high performance even in the dynamic environments. A route splitting algorithm is presented in [22] that provides sustained performance adapting the nodes to battery faults. A routing method that embraces selfish nodes to provide dynamic routes in MANET is proposed in [24]. This method involves a new metric called the path goodness metric that analyzes the nodes and find its suitability for adding it in the route path. Similar methods include [25-27], which performs dynamic routing by considering the trustworthiness of nodes prior to the transmission.

TRUST BASED RANDOM AND ENERGY EFFICIENT ROUTING (TRER)
The trust based random energy efficient routing is designed to be a secure and random routing method that provides security and effective load distribution to all the nodes in the network.
Route selections are performed using a modified Ant Colony Optimization (ACO) algorithm, incorporated with trust details. The usage of ACO, a metaheuristic algorithm makes this method robust and fast while comparing this method with other routing methodologies.

After a transmission request is launched, ACO initializes the base nodes. Our method uses the Ant Density algorithm, which is a variant of the Ant System [1]. When the base nodes are provided to the Ant System, the next node selection is done probabilistically on the basis of the distance and pheromone intensity. Hence it can be assured that this method tends to produce dynamic and random routes. Figure 1 shows the TRER architecture.

Consider \( m \) to be the number of ants and \( n \) to be the number of nodes in the network, then the probability that an ant \( m \) will select a node is given by

\[
p_{ij}(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_k [\tau_{kj}(t)]^\alpha [\eta_{kj}]^\beta}
\]

where \( \tau_{ij} \) is the pheromone intensity in the edge \( ij \) and \( \eta_{ij} \) is the visibility range of the edge \( ij \). \( \alpha \) and \( \beta \) are the weights provided to the pheromone trial and the visibility respectively.

Larger quantity of ant trial signifies that the route is better and more ants starts choosing the path. An evaporation parameter \( \rho \) is used to signify the amount of pheromone that should disappear in a unit time. This parameter makes sure that a path that does not imply to a best route disappears in time. This helps in efficient convergence of the solution.

Let \( \tau_{ij}(t+1) \) be the intensity of trail on path \( ij \) at time \( t+1 \), given the formula

\[
\tau_{ij}(t+1) = \rho \tau_{ij}(t) + \Delta \tau_{ij}(t, t+1)
\]

\( \Delta \tau_{ij} \) is the pheromone level to be deposited on the path \( ij \) and \( \rho \) is the evaporation parameter.

The amount of pheromone to be deposited and the point at which the pheromone is to be deposited differs on the basis of the variant that is being used. The Ant Density model deposits a total pheromone of quantity \( Q \) (determined by the user according to the application) on the path that has been selected.

\[
\Delta \tau_{ij}(t, t+1) = \begin{cases} Q & \text{if } k-th \text{ ant goes from i to j between } t \text{ and } t+1 \\ 0 & \text{otherwise} \end{cases}
\]

The Ant Density algorithm as described in [2] is shown below

Step 1. Initialize Transmission

Step 2. For every edge \( (i,j) \) set an initial value \( \tau_{ij}(t) \) for trial intensity and \( \Delta \tau_{ij}(t+1) := 0 \)

Step 3. Place \( m \) ants on the base node

Step 4. Repeat until tabu list is full

Step 4.1. For every ant do

Step 4.1.1 Calculate the probability of selection for each node using (1)

Step 4.1.2 Shortlist 5 destination nodes \( (L_1) \)

Step 4.1.3 For each node in \( L_1 \)

a. Calculate the probability of selection of each node using (1)

b. Shortlist 5 destination nodes \( (L_2) \)

Step 4.1.4 Identify the best destination node from \( L_2 \) \( (l_{sel}) \) based on the trust values

Step 4.1.5 Identify its corresponding \( L_1 \) node \( (l_{sel}) \)

Step 4.1.6 tabu(s) := \( l_{sel} \)

Step 4.1.7 Move the k-th ant to \( l_{sel} \)

Step 4.1.8 Set the value for \( \Delta \tau_{ij} \) using (3)

Step 4.1.9 For every edge \( (i,j) \) compute \( \tau_{ij}(t+1) \) according to equation (2)

Step 5. Identify and use the shortest route found by the ants

In the current paper, the working of ACO is modified to incorporate the trust base and randomness. Regular implementations of ACO select a single node using the Cumulative Distributive Function and the probability values corresponding to each node. This proposal modifies it by...
considering two levels for the selection of a node. In the first level \( (L_1) \), five nodes are selected. Each of these nodes has their own probability for getting selected as the destination node. In the second level \( (L_2) \) of processing, five nodes are selected for each of the \( L_1 \) nodes totalling to twenty-five nodes.

The \( L_2 \) nodes are analyzed based on the trust levels. Trust levels are calculated using the weighted sum method. Bandwidth, stability, successful transmission rate and transmission rate of failed transactions were used to determine the trust level of a node. Each of the properties are allocated with their specific weights. The trust value of a node is identified using the weighted sum method (3,4) as,

\[
A_{i}^{WSM-score} = \sum_{1}^{m} w_j a_{ij}, \text{ for } i = 1, 2, 3, \ldots, m. \tag{4}
\]

where \( w_j \) is the weight corresponding to the criteria, \( a_{ij} \) is the actual value of the property for \( m \) properties.

After the selection of the level 2 node, its corresponding \( L_1 \) node is identified \( (L_{1sel}) \). This node is identified as the destination node to be added to the tabu list. A similar methodology is followed and the route from the base node to the destination node is identified by each ant. Since the application uses \( m \) ants, \( m \) different routes are obtained from the ant system. The best route is identified by finding the route exhibiting the shortest distance from the base node to the destination node.

**RESULTS**

Simulations were carried out using 30, 50 and 100 nodes and the results obtained are recorded. The ant system was initialized with 10 ants, trail importance \( (\alpha) \) to 0.5, visibility \( (\beta) \) to 5, \( \rho \) to 0.7 and trail amount to be deposited \( (Q) \) to 1000. Distance and the time based calculations are carried out on ACO by obtaining the shortest path including all the nodes in the network. Comparisons are carried out between normal ACO algorithm implementing Ant Density strategy and the hybridized ACO algorithm implementing simulated annealing and random selection strategy including the trust levels.

Figures 2, 3 and 4 show the total distance covered in each of the paths, when considering 30 nodes, 50 nodes and 100 nodes in the network. It could be observed that though the 30 and 50 node networks exhibit slight differences, the network with 100 nodes exhibit a huge difference when considering normal ACO and the modified ACO.

![Figure 2: Distance (30 Nodes)](image-url)
Figures 5, 6 and 7 show the time taken for finding the final path. It could be observed from the graphs that the time taken by ACO is much lesser in all the three networks when compared to the networks using modified ACO.
Figure 5: Time (30 Nodes)

Figure 6: Time (50 Nodes)
Figure 7: Time (100 Nodes)

Figure 8 shows the randomness levels during the creation of routes. Randomness is calculated by identifying the edges and by checking for its reuse in the past two transmissions. Figure shows that the network with 30 nodes exhibit lesser randomness when compared to network with 100 nodes. Still, the network with 30 nodes exhibit a maximum level of randomness of 80%, which is an excellent randomness quotient. Further, network with 100 nodes exhibit a randomness level of 100% for most of the transactions. This shows the efficiency of our algorithm in performing the process of exploration to identify new edges. This helps us achieve the dynamicity promised by the algorithm.

Figure 8: Randomness Level

Figure 9 shows the node reuse levels of nodes in the networks containing 30, 50 and 100 nodes. Reuse of a node is calculated using the past two transmissions, by providing specific starting node and destination node. Several disjoint paths have been identified by the algorithm and it has been observed that the reuse levels are maintained very low and the maximum reuse level of 35% is observed, while the lowest of 0% has been achieved by our algorithm.
Figure 9: Node Reuse Levels

Figure 10 shows the percent of nodes used for the path construction. It could be observed that the transmissions concerning 50 and 100 nodes uses < 35% of the total available nodes for the process of path construction between two nodes. Energy efficiency is brought about in this manner.

Figure 10: Nodes Used for Path Construction

From the results it could be observed that though the tradeoff happens in terms of time and distance, the security aspect is improved to a huge extent. The level of randomness in determining the paths provides safe and non-repeatable paths. Huge minimization of node reuse is the major reason for the increased time and distance. Load distribution over the network plays a major role in maintaining the stability of the network. The creation of selfish nodes is reduced and most of the nodes remain altruistic. This helps extend the lifetime of the network. Further, the total percentage of nodes used for path construction are also maintained low, which leads to energy conservation in the nodes.

CONCLUSION
This paper presents a method that dynamically generates a path with minimal node reuse and appropriate load distribution. Randomness is incorporated into the system such that a path does not follow the edges used by the two prior transmissions and minimizes the usage of the nodes used in the prior transmissions. This leads to an appropriate distribution of load all through the network, hence reduces the power depletion of certain centralized nodes. This process avoids many nodes from turning into selfish nodes and hence the network stability is maintained to its maximum extent. Usage of Simulated Annealing as a local search mechanism has provided added benefits to this method in the form of reduction in the number of nodes being analyzed. The downside existing in this approach is that the time taken for determining the routing path and the distance of the routing path remains high when compared to the regular ACO algorithm. Our further contributions will concentrate on minimizing this gap to provide the best possible tradeoffs from both the perspectives. This method does not consider the traffic type during transmissions. It considers the entire traffic as being a single type and hence delay sensitive packets might experience problems. Hence our other contributions will also deal with accounting the
type of traffic, such that packets that are delay sensitive occupy faster routes when compared to throughput sensitive packets.

REFERENCES


