Improvement of QoS Routing Method Using Ant Colony Algorithm and Fuzzy Logic in Ad hoc Networks

Ali Haghighi Zadeh, Seyed Javad Mirabedini, Mehdi Sadeghzadeh

Department of computer engineering, Dezful Branch, Islamic Azad University, Dezful, Iran
Alihz83@gmail.com

Department of computer engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran
jvd2205@yahoo.com

Department of computer engineering, Mahshahr Branch, Islamic Azad University, Mahshahr, Iran
Sadegh_1999@yahoo.com

Abstract

The ad hoc network is a system of network elements that combine to form a network requiring little or no planning. This may not be feasible as nodes can enter and leave the network. In such networks, each node can receive the packet (host) and the packet sender (router) to act. The goal of routing is finding paths that meet the needs of the network and effectively use network resources. This paper presents a method for QoS routing in ad hoc networks based on ant colony optimization (ACO) algorithm and fuzzy logic. The advantages of this method flexibility and routing are based on several criteria. The results show that the proposed method in comparison with the algorithm IACA has better performance, higher efficiency and greater throughput. Therefore, the combination of ant algorithm with Fuzzy Logic due to its simplicity fuzzy computing is appropriate for QoS routing.

Keywords: Ad hoc Networks, routing, QoS, ACO, fuzzy logic

I. Introduction

An ad hoc network is a set of mobile nodes which communicate over radio and do not need any infrastructure. This kind of networks is very flexible and suitable for several situations and applications, thus they allow the establishing of temporary communication without pre installed infrastructure. Due to the limited transmission range of wireless interfaces, the communication traffic has to be relayed over several intermediate nodes to enable the communication between two nodes. Therefore, this kind of networks is also called multi-hop ad-hoc networks. Nodes not only have to fulfill the functionality of hosts, but also each node has also to be a router, forwarding packets for other nodes. Researches for designing the QoS routing algorithm have been always the hot topics in ad hoc network. In recent years, scholars have put forward a number of helpful models to solve the problem of QoS, such as protocols, strategies, algorithms, and scheduling mechanism.

ACO is one of the most efficient algorithms for solving QoS routing problem. Ant colony optimization is a meta-heuristic for hard discrete optimization problems that was first proposed in the early 1990s (Dorigo, Maniezzo, & Colomj, 1996). The inspiring source of ACO is the foraging behavior of real ants. When searching for food, ants initially explore the area surrounding their nest in a random manner. As soon as an ant finds a food source, it evaluates quantity and quality of the food and carries some of the found food to the nest.
During the return trip, the ant deposits a chemical pheromone trail on the ground. The quantity of pheromone deposited, which may depend on the quantity and quality of the food, will guide other ants to the food source. The indirect communication between the ants via the pheromone trails allows them to find shortest paths between their nest and food sources. This functionality of real ant colonies is exploited in artificial ant colonies in order to solve discrete optimization problems such as network routing. Most routing methods used ACO just deal with one parameter and considered the other important parameters like delay and delay jitter as constraints problem. It is apparent that there is a need for more efficient algorithm which gives multi constraints involving cost, delay, bandwidth etc. Fuzzy ant (Rozin & Margaliot, 2007) demonstrated by using fuzzy modeling to develop a mathematical model for the foraging behavior of ants. ANTHOC (Thomas, Chellappan, & Jayakumar, 2005) a QoS routing algorithm inspired by ACO has been presented. The probabilistic routing table balances the load and avoids congestion using randomized routing. (Mirabecini, Teshnehlab, & Rahmani, 2007) is a routing algorithm based on ant colony and enhanced by fuzzy logic for network routing. The advantages of such an intelligent algorithm include increased flexibility in the constraints that can be considered in making the routing decision efficiently and the simplicity in taking into account multiple constraints. AFAR (Mirabedini, Teshnehlab, Shenasa, Movaghar, & Rahmani, 2008) algorithm did a better job at dispersing traffic in a more uniform manner throughout the network. It also handles an increased traffic load as well as decreased transmission delay by utilizing network resources more efficiently. FACO (Goswami & Dharaskar, 2009) based on ACO and enhanced by fuzzy logic for network routing in ad hoc network. The different routing metrics are taken into consideration using a fuzzy logic system to compute pheromone capturing interplay of various metrics. Route selection is then performed by selecting the route having highest concentration. In (George & Rajakumar, 2013) proposed technique was tested for travelling sales man problem. This method is tested for different ants, by sending different number of tours and identified different shortest path. The most important parameter in this method is cost. In this paper, we present a QoS routing algorithm in ad hoc networks based on ant colony optimization algorithm and fuzzy logic. The remainder of this paper is organized as follows: In Section 2, we give an overview of Ant Colony Routing algorithm, a general definition about fuzzy logic is presented in section 3, a discussion of our proposed method and its features appears in Section 4. Then, in Section 5, simulation and results are given, and Section 6 concludes the paper.

II. Basics Of Ant Colony Routing

A real ant colony is able to find food and follow the shortest path from the nest to the food. As a real ant moves, it deposits a substance called pheromone on the ground. When an ant reaches a point that has more than outgoing branch, the probability that a branch will be selected by an ant is dependent on the amount of pheromone deposited on each branch. An ant will select a branch and deposit more pheromone on this branch; as a result, the probability of selecting this branch will increase. The pheromone on the branches of the shortest path to the food will grow faster than pheromone on other branches. The pheromone is evaporated over time, allowing the system to forget old paths and helping to avoid quick convergence to a sub-optimal solution. A single ant is not intelligent, but the ant colony can find the shortest path. As the ants search for the shortest path, they explore many paths. The longest paths and unexplored paths still have a probability to be visited. If the shortest path fails, the ants will follow a recently explored path. Even if, the first ants used the longer path, the ant colony is able to find the shortest one as the pheromone evaporates with time and the shortest path still has a probability to be visited. The real ant colony is a dynamic self-built
and self-configured system, which is capable of solving its problems efficiently. These features of real ant colony system cater to the requirements of the ad hoc networks. The proposed algorithm is built on the top of this basic routing algorithm.

III. Fuzzy Logic

Fuzzy logic is a superset of conventional (boolean) logic that has been extended to handle the concept of partial truth. It was first introduced by L. Zadeh in the 1960s as a means to model the uncertainty of natural language, and has been widely used for supporting intelligent systems. A key feature of Fuzzy logic is to handle uncertainties and non-linearities, existing in physical systems, similarly to the reasoning conducted by human beings, which makes it very attractive for decision making systems. A fuzzy logic system comprises basically three elements: A fuzzifier, an inference method (rules and reasoning) and a defuzzifier. Fuzzy systems are used to approximate functions. The fuzzy can be used to model any continuous function or system. Figure.1 shows the generalized block diagram of fuzzy system. The basic unit of fuzzy function approximation is “If-then” rules. A fuzzy system is a set of if-then rules that maps input to output.

![Generalized Fuzzy System](image)

Figure.1. Generalized Fuzzy System

The steps involved in the Fuzzy inference system design are as follows:

**Step 1:** Fuzzy Inputs
This step will obtain inputs and normalize them in the range of $(0,1)$, then determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. Fuzzification of the input amounts to either a table lookup or a function evaluation.

**Step 2:** Apply Fuzzy Operator
This step determines the degree to which each part of the antecedent has been satisfied for each rule.

If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number will then be applied to the output function. The input to the fuzzy operator is two or more membership values from fuzzified input variables. The output is a single truth value. The method used may be either AND or OR operation.

**Step 3:** Apply Implication Method
Before applying implication proper weights are assigned to each rule. The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set.

**Step 4:** Aggregate all outputs
Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set. Aggregation only occurs once for each output variable, prior to the fifth and final step, defuzzification. The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule. The output of the aggregation process is one fuzzy set for each output variable.

**Step 5: Defuzzify**

The input for the defuzzification process is a fuzzy set and the output is a single number. The aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set. Finally the output is denormalized and is given as the result.

**IV. Proposed method**

In this paper, proposed algorithm is constructed with the communication model observed in ant colonies and fuzzy logic technique for routing in ad hoc networks. It extends IACA (Ding, Shi, & Wu, 2012) algorithm and fuzzy logic.

**A. Fuzzy Inference System (FIS)**

The Fuzzy Inference System (FIS) for proposed method is a mamdani type system with three inputs and one output. The system inputs are route (or link) distance, delay and bandwidth. Inputs are characterized by the fuzzy membership functions as shown in Figure.2, Figure.3 and Figure.4. The membership functions for the fuzzy sets of inputs are chosen to be triangular. Inputs are normalized between (0,1) before applying to FIS. As shown in Figure.2, Figure.3 and Figure.4, inputs variables distance, delay and bandwidth have three membership functions titled as L, M and H which mean Low, Medium and High respectively.

![Figure 2](image1.png)

*Figure 2. Fuzzy Membership function for Distance*

![Figure 3](image2.png)

*Figure 3. Fuzzy Membership function for Delay*

![Figure 4](image3.png)

*Figure 4. Fuzzy Membership function for Bandwidth*
The rules of the FIS are designed for an optimal performance. TABLE I, TABLE II and TABLE III show rule base for the FIS. In these tables the Values for the amount of goodness from lowest to highest are defined as VL (Very Low), LM, LH, ML, MM (Medium), MH, HL, HM, and VH (Very High).

**TABLE I**

**FUZZY RULES FOR LOW DISTANCE**

<table>
<thead>
<tr>
<th>Delay</th>
<th>Bandwidth</th>
<th>L</th>
<th>M</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td></td>
<td>HL</td>
<td>HM</td>
<td>VH</td>
</tr>
<tr>
<td>M</td>
<td></td>
<td>MH</td>
<td>HL</td>
<td>HM</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>MM</td>
<td>MH</td>
<td>HL</td>
</tr>
</tbody>
</table>

**TABLE I**

**FUZZY RULES FOR MEDIUM DISTANCE**

<table>
<thead>
<tr>
<th>Delay</th>
<th>Bandwidth</th>
<th>L</th>
<th>M</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td></td>
<td>ML</td>
<td>MM</td>
<td>MH</td>
</tr>
<tr>
<td>M</td>
<td></td>
<td>LH</td>
<td>ML</td>
<td>MM</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>LM</td>
<td>LH</td>
<td>ML</td>
</tr>
</tbody>
</table>

**TABLE I**

**FUZZY RULES FOR HIGH DISTANCE**

<table>
<thead>
<tr>
<th>Delay</th>
<th>Bandwidth</th>
<th>L</th>
<th>M</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td></td>
<td>LH</td>
<td>ML</td>
<td>MM</td>
</tr>
<tr>
<td>M</td>
<td></td>
<td>LM</td>
<td>LH</td>
<td>ML</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>VL</td>
<td>LM</td>
<td>LH</td>
</tr>
</tbody>
</table>

There are 27 rules defined for this fuzzy system. For example If (Delay is H) and (Bandwidth is L) and (Distance is H) then (Path_Goodness is VL).

The output of FIS which is a Path Goodness is applied to the software simulation for evaluations. Path Goodness is calculated as follows:
Path Goodness\(_{(t)}\) = \frac{\sum_{j=1}^{M} y \prod_{i=1}^{n_{f}} \mu_{A'_{i}(X)}(x_i)}{\sum_{i=1}^{M} \prod_{i=1}^{n_{f}} \mu_{A'_{i}(X)}(x_i)} \tag{1}

Where \(i\) represents the path index, \(M\) represents the number of fuzzy rule bases used (\(M=27\)), \(n_{f}\) represents the number of membership functions for input variables (\(n_{f}=3\)), \(y\) represents center of fuzzy set and \(\mu_{A'_{i}(X)}(x_i)\) represents The Fuzzy value of membership functions.

Design of Fuzzy Inference System is the process of formulating the mapping from a given input to an output using fuzzy logic. Mamdani type inference expects the output membership functions to be fuzzy sets. After the aggregation process, there is a fuzzy set for output variable as shown in Figure 5. All of the membership functions for the fuzzy sets of inputs and output are chosen to be triangular for its easiness in computation, Clarity and noise tolerance.

B. Proposed Algorithm implementation

1) Initialization: set timer \(t := 0\); Set cycle counter \(NC := 0\); Set maximum number of iteration \(NC_{\text{MAX}}\); Initialize the pheromone \(\tau_{ij} (t) = c\) of each link; \(m\) ants start from the source node.

2) Simplify network: checking goodness of each link and delete the link that do not satisfy link goodness.

3) Set the tabu index \(s = 1\); for \(k = 1\) to \(m\) ants; \(\text{tabu}_{k}(s) = \text{source}_\text{node}\). the node walked through by ant \(k\) is recorded by tabu table \(\text{tabu}_{k}\).

4) Repeat until the tabu table \(\text{tabu}_{k}\) is filled up: set \(s := s + 1\); According to the following rules, Ant \(k\) in node \(i\) selects the next node \(j\) at time \(t\):

\[
\begin{align*}
j &= \left\{ \begin{array}{ll}
\arg \max_{k \in \text{allowed}_i} \{ [\tau_{i,j}]^\alpha [\eta_{i,j}]^\beta \} & \text{if } q \leq q_0(t) \\
\text{choose } j \text{ according to (4)} & \text{otherwise}
\end{array} \right.
\end{align*}
\tag{2}
\]

\[
q_0(t) = \begin{cases} q_1 & t \leq T_1 \\ q_2 & T_1 < t \leq T_2 \\ q_3 & t > T_2 \end{cases}
\tag{3}
\]

\[
P_{ij}^k (t) = \begin{cases} \sum_{k \in \text{allowed}_i} \left[ \tau_{ij} \right]^\alpha \left[ \eta_{ij} \right]^\beta & \text{if } j \in \text{allowed}_k \\ 0 & \text{otherwise} \end{cases}
\tag{4}
\]

Where \(q\) is a uniform random number in the interval \([0, 1]\). \(q_1, q_2, q_3\) are constants and \(q_1 < q_2 < q_3\). \(\eta_{i,j} = \frac{1}{\text{Path Goodness}_g}\) is a heuristic function. \(\tau_{ij}\) represents the amount of pheromone. We use \(\alpha\) and \(\beta\) to represent the relative importance of the amount of pheromone.
and the predictable information respectively. The nodes selected by ant $k$ are not been visited by ant $k$ before, the allowable set is denoted by allowed$_k$=$\{0,1…n-1\}$ - tabu$_k$.

5) The local update rule for pheromone
When ant $k$ passes the route $(i, j)$, the pheromone of the route will be updated according to follow:

$$\text{For every edge } (i, j)$$
$$\text{For } k := 1 \text{ to } m$$
$$\tau_{ij}(t) = (1 - \rho)\tau_{ij} + \rho e^{cons}$$

Where $0 < \rho \leq 1$ it shows the speed of the old pheromone evaporation in the route. it avoid unlimited accumulation of pheromone.

6) The global update rule for pheromone
After all ants complete a search successfully, we choose the global best ant in the current iteration.

$$\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij} + \rho\Delta\tau_{ij}(t)$$

$$\Delta\tau_{ij} = \begin{cases} 
\frac{Q(t)}{L_{best}} & (i, j) \in \text{tabu}_k \\
0 & \text{otherwise}
\end{cases}$$

$$Q(t) = \log(t + 1)$$
$$NC + +$$
$$\text{For every edge } (i, j) \Delta\tau_{ij} = 0$$

Where $L_{best}$ represents best path that has been achieved so far.

7) Stop condition
If $NC < NC_{\text{MAX}}$, then clear the tabu table of each ant, and go to 3); otherwise, output the optimal route, and stop the algorithm.

V. Experimental and simulation results

A. Experimental model

We conducted experiments to evaluate and compare the performance of proposed algorithm and IACA (Ding et al., 2012). In these experiments, we used software MATLAB 2013a and network simulator Opnet Modeler 14.5. An ad hoc network with randomly generated 20 nodes shown in Figure.6 is considered. The node_1 is source of packet traffic generations and the node_20 is destination. In order to evaluate the network performance the traffic sources uniform and exponential is considered and the simulation time is 600 seconds.
B. Experimental Result

The performance of proposed method with IACA is evaluated according to the following metrics:

1. **End-to-end Delay**: Delay incurred by a packet being transmitted between a source and destination node. End-to-end Delay is included transmission delay, propagation delay, processing delay and Queuing delay.

2. **Throughput**: number of bits sent by a source node that arrive at the destination node.

In order to simulate network performance metrics in both uniform and exponential traffic are examined.
• **End-to-end Delay**

![End-to-end Delay for uniform traffic](image1)

**Figure.7.** End-to-end Delay for uniform traffic

![End-to-end Delay for exponential traffic](image2)

**Figure.8.** End-to-end Delay for exponential traffic

Figure.7 and Figure.8 illustrated that both algorithms have a little difference because of effect of processing time for using fuzzy control mechanism.
• Throughput

As shown in Figure 9 and Figure 10, it is observed that proposed algorithm has higher throughput than IACA due to its capability to update routing tables by the smart adaptive fuzzy method.

VI. Conclusions
In this paper, we have proposed a QoS routing algorithm based on ant colony optimization and enhanced by fuzzy logic in ad hoc networks. The benefits of such a heuristic algorithm include raised flexibility in the parameters that can be considered in making the routing decision efficiently and the simplicity in taking into consideration multiple parameters. The results of this experiment indicate proposed algorithm has higher throughput than IACA as well as the computational load of a fuzzy control routing system is not very great. Hence integration ant colony algorithm with fuzzy logic due to its simplicity fuzzy computing would be appropriate choice for QoS routing in ad hoc networks.
References


