Routing in Delay Tolerant Network Using Genetic Algorithm

Priti Bhardwaj 1, Rahul Johari 2

1 M.Tech Scholar, Computer Science and Engineering, GGSIP University, Delhi, INDIA,
2 Assistant Professor, Computer Science and Engineering, GGSIP University, Delhi, INDIA.

ABSTRACT

Routing in Delay Tolerant Networks (DTN) is a major problem because of intermittent connectivity between the nodes and no prior knowledge about the network is known to us. So we use the concept of Anycast routing where group to group communication takes place. Members of different group would communicate through DTN (Delay Tolerant Networks). Anycast routing says that for successful delivery of a message the only thing required is to deliver the message either to the destination node of that group or to any node of that particular group. We use the same concept for our simulation and use Genetic Algorithm for routing also. Here we use random fitness function and also use the concept of crossover in the algorithm. We implement two different models to simulate the routing strategy.

Key Words: Delay tolerant network, Genetic Algorithm, Routing Algorithm.

I. INTRODUCTION

This Delay/Fault Tolerant Mobile Sensor Network (DFTMSN) are established on adhoc basis without any predefined configuration. Fault Tolerant Networking (DTN) has grown as a research area that is focused on addressing the delivery of the message requirements absolute to challenged networks. There is no end to end connectivity in these types of networks. Routing in DTNs [3, 4] is one of the principal components and remains open for examination in the DTN architecture. The transmission of message would not be sent in one hop. It may use one hop or more than one hop. The message is being stored in the buffer of the node and will sent the message when it get the opportunity to send or we can say
the destination node is available for communication. No prior information is known about the network dynamics. Many of the principles of DIN architecture are reviewed by Fall and Farrell [5], being highlighted design decisions that have persevered through repeated analyses. An anycast routing algorithm for DTNs based on genetic algorithms (GAs) is presented and analysed [1]. The anycast routing delivery is a major utility in DTNs and it has not been very well explored yet. Anycast routing works on networks where some nodes require a route to any member from a certain group of utility nodes. We use the concept of crossover in genetic algorithm and use a random fitness function for the deletion of the nodes. Persistent DTN for routing can be taken in our algorithm as destination and source shows their existence for the communication.

II. PROPOSED METHOD

Routing in Delay tolerant network has been a great issue from a long time. So here we have tried to carry out the routing from a group to another group. A node has two features: node-id and group-id. A node belonging to a particular group will send a message to a node of another group. Here we have used model semantics [1] which specify the type of delivery of message in which we have added DTN contacts:

1. Current Membership Model according to which message will be transferred at a particular point of time: PREDICTED
2. Temporal Interval Membership model according to which message will be transferred during a given interval of time: ON-DEMAND
3. In this paper we have taken Persistent DTN contact in which we have source and destination defined. Genetic algorithm is being used in this paper [2] as it is a search and optimization technique and is based on the natural selection and evolution. As the genetic algorithm maintains a population of chromosomes, we have maintained a population of nodes. Reproduction technique of genetic algorithm, i.e. crossover is being applied on two groups to change the position of node-id. Random fitness function is being used to minimize the population of the nodes in the network.

III. GENETIC ALGORITHMS

Genetic algorithms (GAs) are global search and optimization techniques modeled from natural selection, genetic and evolution [6]. A genetic algorithm maintains a population of candidate solutions, where each candidate solution is usually coded as binary string called a chromosome. A set of chromosomes forms a population, which is evaluated and ranked by fitness evaluation function. The fitness evaluation function play a critical role in GAs because it provides information how good each candidate. The initial population is usually generated at random. The evolution from one generation to the next one involves mainly three steps: fitness evaluation, selection and reproduction. First, the current population is evaluated using the fitness evolution function and then ranked based on their fitness. A new generation is created with the goal of improving the fitness. Simple GA uses three operators with probabilistic rules: reproduction, crossover and mutation. First selective reproduction is applied to the current population so that the string makes a number of copies proportional to their own fitness. This results in an intermediate population. Second, GA select "parents" from the current population with a bias that better chromosome are likely to be selected. This
is accomplished by the fitness value or ranking of a chromosome. Third, GA reproduces "children" (new strings) from selected parents using crossover and/or mutation operators. Crossover is basically consists in a random exchange of bits between two strings of the intermediate population.

IV. METHODOLOGY

We have used MATLAB environment to carry out our simulation. We have taken a network of nodes divided into groups, each group containing.

Nodes=number of nodes/number of groups. Each node is identified by its node-id and group-id to which it belongs. We have used Genetic Algorithm to apply crossover between groups of nodes. By applying crossover between the groups, particular node-id will be swapped. When we take the source and destination node, we are considering their group-id also, so that the particular node-id will be picked up from their particular group. We have used models according to which message will be transferred: Current Membership Model according to which a particular message will be transferred at a particular point of time and Temporal Interval model according to which a particular packet will be transferred during a given interval of time. We have used a timer function which is transferring the message in T time, where T=message delivery time – message transmission time. Here we have used a random Fitness Function which is decrementing the population of nodes in the network. When the message is being transferred between a particular source and destination node then we are removing those nodes from the network and store them in a separate buffer. Again a different source and destination node will be used for message delivery.

V. PROPOSED ALGORITHM

1. Initialize a network of given number of nodes with their node-id.
   1.1 Make a number of groups of nodes in the network.
   1.2 Each group containing nodes=number of nodes in the network / number of groups.
   1.3 Each node is identified by their group-id to which they belong along with their node-id.

2. Apply Crossover between groups
   2.1 Accept the groups from the user to apply Crossover.
   2.2 Accept the node-id from the user to apply Crossover
   2.3 Crossover is created between groups.

3. While (Count < Number of nodes in network)
   3.1 Count=0;
   3.2 Initialize the Source and Destination node.
   3.3 Accept the Message Transmission Time from System clock.
   3.4 Accept the Message Delivery Time from the user.
   3.5 T= Message Delivery Time – Message Transmission Time
   3.6 IF (Message Delivery Time > Message Transmission Time)
     3.6.1 Accept the model choice from the user according to which message will be transferred:
       3.6.1.1 Current membership model (at particular point of time).
       3.6.1.2 Temporal interval membership model (A given interval of time is given)
3.6.2. Accept the message from the user.
   3.6.2.1. Message will be transferred from source to destination in T time.
   3.6.2.2. Destination will store the message in its own buffer space.
   Else
   Message will be discarded. (Invalid delivery time)
3.7 IF (Message is transferred)
   3.7.1 Count=Count+2;
   3.7.2. Apply a random fitness function (We are decrementing the nodes)
       3.7.2.1 Delete the last source and destination from the initial network of nodes.
       Else
       Go to step 3.
4. End of While loop
5. END

VI. SIMULATION AND RESULTS

Following are the results of our simulation of the algorithm

![MATLAB Command Window]

FIG:1
We are initializing a network of 20 nodes
FIG-2

We are making 4 groups with their group-id, each group containing 5 nodes.

FIG-3

We are selecting group 2 and 4 to apply crossover between them. And thus crossover takes place between node-id 14 and node-id 30.
We are selecting the Current Membership Model for message delivery. The message will be delivered after 20 seconds.

The message (‘DTN’) will be transferred and stored in destination node’s buffer. Source node 6 and destination node 19 will be deleted.
Fitness Function is applied to delete the source node-id 6 and destination node-id 19 so that the population of nodes will be minimized.

<table>
<thead>
<tr>
<th>FIGURE NUM</th>
<th>DESCRIPTION</th>
<th>USER INPUT</th>
<th>OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>We are initializing a network of 20 nodes.</td>
<td>User enters the node-id of the nodes, no of Groups.</td>
<td>A network is created</td>
</tr>
<tr>
<td>2</td>
<td>We are making 4 groups of the nodes.</td>
<td>No of groups given by the user</td>
<td>4 groups are created with group-id 1,2,3,4 with 5 nodes in each group.</td>
</tr>
<tr>
<td>3</td>
<td>We are applying crossover (concept of genetic algorithm) between the groups</td>
<td>User enters the group no 2 and 4 for crossover. and node-id 14 and 30.</td>
<td>Node-id 14 of group no 2 and node-id 30 of group no 4 are swapped.</td>
</tr>
<tr>
<td>4</td>
<td>We are selecting the model for message delivery.</td>
<td>User enters the delivery time of message and check if it is greater than the transmission time.</td>
<td>Message is delivered in 20 seconds and stored in destination node buffer.</td>
</tr>
<tr>
<td>5</td>
<td>We are transferring the message to destination node</td>
<td>User has entered the message</td>
<td>The message will be stored in destination node buffer.</td>
</tr>
<tr>
<td>6</td>
<td>Fitness function is applied to decrement the population of the nodes.</td>
<td>User has entered the source node-id 6 and destination node-id 19.</td>
<td>Source node-id 6 and destination node-id 19 have been deleted from the original network of nodes.</td>
</tr>
</tbody>
</table>

**Table: 1**
VII. CONCLUSION AND FUTURE WORK

We have developed a simple routing algorithm using Genetic Algorithm in this paper. Our algorithm is being simulated for a given number of nodes divided into a particular number of groups. Message is being transferred from source group to destination group as each node is identified by its node-id and group-id. We have used the concept of Genetic Algorithm between groups to apply crossover between node-id of corresponding groups. A random fitness function is being applied to delete the last source and destination nodes. The results we have obtained affirmed the potential of the algorithm. As future works, we can take the Security issue of the message being transferred. Also we tend to develop different algorithms using genetic algorithm to find the multicrossover point. We can also develop different techniques to minimize the routing overhead. Priority of message could be taken into account. GA will be more studied to find out more efficient routing algorithms.

VIII. ACKNOWLEDGEMENT

We extend our sincere thanks and gratitude to the administration of GGSIP U University for providing academic environment to pursue our research activities.

REFERENCES