Routing Optimization using GA in DTN

Atul Sharma, Kamlesh Kumari

Department of Computer Science & Engineering, UIET, KUK, Haryana, India

Abstract - Delay Tolerance Networks (DTNs) are infrastructure less wireless networks. DTNs are wireless networks where disconnections and delays occur frequently due to phenomena such as node mobility, power outages etc. In order to achieve data delivery, a Store – Carry – Forward approach is used. In store carry and forward approach node stores the message in its buffer and forward the message to next intermediate node when that node in the range of network. In this paper we first describe delay tolerance network and its different routing schemes. Then we discuss Genetic Algorithm (GA) in Delay Tolerance Network. In GA the working algorithm is discussed.

Keywords: Delay Tolerance Network (DTN), Genetic Algorithm (GA) Routing Schemes.

I. INTRODUCTION

Delay-tolerant networking (DTN) is a kind of computer network architecture that addresses the technical issues in heterogeneous networks lacking continuous end to end network connectivity. Examples of such networks are that operate in mobile or extreme terrestrial environments or interplanetary networks [1,2]. A delay-tolerant network (DTN) is a network of regional networks. It is an overlay on top of regional networks, including the Internet. DTNs accommodate long delays between and within regional networks, and support interoperability of regional networks by translating between regional network communications characteristics. Communication between nodes in DTNs could be either scheduled or opportunistic, depending on the knowledge that we have about the initial status and the time evolution of the network.

A. Characteristics of DTN [4,5]

Delay Tolerant networks may be characterized by the combination of the following characteristics:

Intermittent Connectivity: If there is no consistent end-to-end path between the source and destination -a phenomenon known as network partitioning, end-to-end communication using the TCP/IP protocols does not work. Other protocols are required.

Asymmetric data rates: The Internet does support some forms of asymmetric bi-directional data, as in cable TV or asymmetric DSL access. But if asymmetries increase then they will hinder traditional interactive protocols such as TCP.

High error rates: The delay tolerant networks have high error rates. Ambiguous mobility patterns: Unlike the case with public bus services that maintain fixed routes or planetary trajectories, future behavior of a node is not fully known for many DTN applications. It is widely assumed, however, that node mobility patterns (while random) are generally recurrent.

Long or variable delay: Long propagation delays between nodes, in addition to variable queuing delays at node buffers, all create end-to-end path delays that far exceed the threshold levels usually tolerated by Internet protocols and applications that rely on quick return of acknowledgements.

B. Routing Strategies

Data is delivered in a DTN using a store-carry-forward model. The main categories of routing schemes for delay tolerance networks are given below:

Epidemic routing

The epidemic routing is one of the simplest and earliest routing schemes for DTN [8]. In this routing strategy, whenever two nodes come in contact with each other, they exchange all the messages they currently carry at that point of time. In other words, the packets are spread like a viral epidemic. So this routing strategy is fastest
possible routing scheme. In epidemic routing, the data delivery results in inefficient use of the network resources such as power, bandwidth, and buffer at each node[4,5].

Location Based Routing

In some cases, the location of the nodes may be known, that can be used in case of opportunistic forwarding of messages in DTN. The location information of the nodes may be known in either a physical (for example, from GPS devices attached to nodes or through a location service) or a virtual coordinate space (designed to represent network topology taking obstacles into account). When an encounter occurs, the node forwards data to another node only if it is closer to the destination [4,5].

Spray and Wait Routing

In spray and wait routing there are two phases.

First phase is Spray phase in which it limits the generation of duplicity of messages to minimize the use of network resources.

Second phase is Wait phase in which sender wait for the response of sent messages which are sent in spray phase if there is no response of delivery of message then sender again send the messages to destination.

Spray and Wait routing overcomes the problem occurred in epidemic based routing.

In Spray and Wait routing, duplicity of message is less as compared to epidemic based routing. Also bandwidth consumption is less in Spray and Wait routing as compared to epidemic [4,5].

First Contact Routing

In this node send a message randomly using any available contact .If none of the path is available then the message wait for that particular path until become available and is assigned to first available contact.

II. GENETIC ALGORITHM

Genetic algorithm (GA) is a subclass of evolutionary algorithms (EA) which generate solutions to optimization problems using techniques inspired by natural evolution such as selection, crossover, mutation and inheritance.

Four steps have to be considered to solve a problem with GA. The first step is initialization in which the evolution usually starts from a population of randomly generated individuals. Each individual or chromosome is a set of genomes. In each generation, we need to evaluate the chromosomes by a fitness function and determine how suitable each of them is to be chosen for the next generation.

The population of subsequent generations is formed by selecting and modifying the proper prior chromosomes. These modifications are based on crossover and mutation. Crossover is a method to combine two chromosomes to produce new offspring. The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. In mutation we alter one or more genome values in a particular chromosome. Mutation is done in order to prevent the population from stagnating at any local optima. The procedure of breeding offspring from previous generation will continue until a termination condition has been reached. This condition may be either finding a solution that satisfies minimum criteria or reaching a fixed number of generations.

In the following subsections we explain how we used the genetic algorithm for our purpose to improve the simulated network parameters.

Initialization

In this step we need to construct the first generation of the chromosomes. In our case of study each chromosome contains five genomes. As shown in figure 1, we put the coefficients of the proposed parameters in the chromosome. Also we considered the threshold of the evaluation formula as the fifth genome.

<table>
<thead>
<tr>
<th>α</th>
<th>β</th>
<th>γ</th>
<th>δ</th>
<th>Threshold</th>
</tr>
</thead>
</table>

Fig 1. Chromosome Structure
According to this fact that we have no idea about the proper values of these genomes at the beginning of the simulation, many individual solutions are randomly generated to form an initial population. As you will see in the simulation & result section, the value of each coefficient varies from 0 to 1 that causes the value of evaluation formula to vary in the range of 0 to 4.

Selection
After forming the first generation, we need to choose some chromosomes from current generation’s population for inclusion in the next generation’s population. This step of genetic algorithm is called selection. There exist different methods to select chromosomes from a particular population. Roulette-wheel selection [9], Tournament selection [10] and Top percent selection [8] are examples of these methods. In all these selection algorithms a fitness function is used to determine the optimality of a chromosome, so the particular chromosome can be ranked against all the other chromosomes.

Reproduction
After selecting the proper chromosomes as the parents we need to breed new solutions from them for the next generation. The method that we used for breeding child solutions from a pair of parent solutions is One-point crossover. As it is shown in figure 2, a single crossover point on both parents’ chromosome is selected. Then we interchange the two parent chromosomes at this point and produce two new offspring. After reproduction of new chromosome we need to apply mutation on some of them. Mutation should be applied to avoid the solutions from trapping in local optima. Mutation occurs during evolution according to a user-definable mutation probability. This probability should usually be set fairly low (about 0.01%) to prevent the algorithm to turn into a primitive random algorithm.

Termination
The genetic processes for breeding new generations of solutions discussed in the previous subsections will continue until a termination condition has been reached. It is because these processes are time consuming and they should be stopped after breeding particular generations. For example in our simulation which we will discuss in the next section the algorithm will be terminated after breeding four generation of solutions. Figure 3 illustrates the complete flowchart of discussed algorithm.

III. CONCLUSION AND FUTURE SCOPE

In this Paper we discuss various routing schemes like Epidemic, Prophet, Spray and wait etc. And various Movement Models also discuss basics of
GA and Swam Intelligence. Then we provide a basic GA algorithm that describes the working principle of GA. In future it is intended to create new GA based algorithm and implement it in DTN.

REFERENCES


