Conditional shortest path routing in Delay tolerant networks

1S.Divya,
2M. Venkata Krishna Reddy,
3D.Jamuna.

2. Associate Professor, Dept. of CSE, Jaya Prakash Narayan College of Engineering, Mahabubnagar, AP-509 001
3. Professor & Head of CSE Dept. Jayaprakash Narayan College of Engineering, Mahabubnagar, A.P-509001

Abstract

This article studies Delay tolerant networks (DTNs) where each node knows the probabilistic distribution of contacts with other nodes. Delay tolerant networks are characterized by the sporadic connectivity between their nodes and therefore the lack of stable end-to-end paths from source to destination. Since the future node connections are mostly unknown in these networks, opportunistic forwarding is used to deliver messages. Based on the observations about human mobility traces and the findings of previous work, we introduce a new metric called conditional intermeeting time. We propose Conditional Shortest Path Routing (CSPR) protocol that route the messages over conditional shortest paths in which the cost of links between nodes is defined by conditional intermeeting times. When a node receives a message from one of its contacts, it stores the message in its buffer and carries the message until it encounters another node which is at least as useful (in terms of the delivery) as itself. Through trace-driven simulations, we demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols that use the conventional intermeeting time as the link metric.

Keywords - Delay Tolerant Networks, CSPR, MEED

1. INTRODUCTION

Routing in delay tolerant networks (DTN) is a challenging problem because at any given time instance, the probability that there is an end-to-end path from a source to a destination is low. Since the routing algorithms for conventional networks assume that the links between nodes are stable most of the time and do not fail frequently, they do not generally work in DTN’s. Therefore, the routing problem is still an active research area in DTN’s. Routing algorithms in DTN’s utilize a paradigm called store-carry-and-forward. When a node receives a message from one of its contacts, it stores the message in its buffer and carries the message until it encounters another node which is at least as useful (in terms of the delivery) as itself. Then the message is forwarded to it. Based on this paradigm, several routing algorithms with different objectives (high delivery rate etc.) and different routing techniques (single-copy, multi-copy, erasure coding based etc.) have been proposed recently. However, some of these algorithms used unrealistic assumptions, such as the existence of oracles which provide future contact times of nodes. Yet, there are also many algorithms based on realistic assumption of using only the contact history of nodes to route messages opportunistically.

Recent studies on routing problem in DTN’s have focused on the analysis of real mobility traces (human, vehicular etc.). Different traces from various DTN environments are analyzed and the extracted characteristics of the mobile objects are utilized on the design of routing algorithms for DTN’s. From the analysis of these traces performed in previous work, we have made two key observations. First, rather than being memory less, the pairwise inter meeting times between the nodes usually follow a log-normal distribution. Therefore, future contacts of nodes become dependent on the previous contacts. Second, the mobility of many real objects are non-deterministic but cyclic. Hence, in a cyclic Mobi Space, if two nodes were often in contact at a particular time in previous cycles, then they will most likely be in contact at around the same time in the next cycle. Additionally, previous studies ignored some information readily available at transfer decisions. When two nodes (e.g., A and B) meet, the message forwarding decision is made according to a delivery metric (encounter frequency, time elapsed since last encounter, social similarity etc.) of these two nodes with the destination node (D) of the message. However, all these metrics depend on the separate meeting histories of nodes A and B with destination node D1. Nodes A and B do not consider their meetings with each other while computing
their delivery metrics with D. To address these issues, we redefine the inter meeting time concept between nodes and introduce a new link metric called conditional inter meeting time. It is the inter meeting time between two nodes given that one of the nodes has previously met a certain other node. For example, when A and B meet, A(B) defines its conditional inter meeting time with destination D as the time it takes to meet with D right after meeting B (A). This updated definition of inter meeting time is also more convenient for the context of message routing because the messages are received from a node and given to another node on the way towards the destination. Here, conditional inter meeting time represent the period over which the node holds the message.

To show the benefits of the proposed metric, we adopted it for the shortest path based routing algorithms designed for DTN’s. We propose conditional shortest path routing (CSPR) protocol in which average conditional inter meeting times are used as link costs rather than standard inter meeting times and the messages are routed over conditional shortest paths (CSP). We compare CSPR protocol with the existing shortest path (SP) based routing protocol through real trace- driven simulations. The results demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols. This shows how well the conditional inter meeting time represents inter node link costs (in the context of routing) and helps making effective forwarding decisions while routing a message.

2. EXISTING SYSTEM

A connected hypercube with faulty links and/or nodes is called an injured hypercube. A distributed adaptive fault-tolerant routing scheme is proposed for an injured hypercube in which each node is required to know only the condition of its own links. Despite its simplicity, this scheme is shown to be capable of routing messages successfully in an injured n-dimensional hypercube as long as the number of faulty components is less than n. Moreover, it is proved that this scheme routes messages via shortest paths with a rather high probability, and the expected length of a resulting path is very close so that of a shortest path. Since the assumption that the number of faulty components is less than n in an n-dimensional hypercube might limit the usefulness of the above scheme, a routing scheme based on depth-first search which works in the presence of an arbitrary number of faulty components is introduced. Due to the insufficient information on faulty components, however, the paths chosen by this scheme may not always be the shortest. To guarantee all messages to be routed via shortest paths, the authors propose to equip every node with more information than that on its own links. The effects of this additional information on routing efficiency are analyzed, and the additional information to be kept at each node for the shortest path routing is determined.

3. PROPOSED SYSTEM

In this proposed system, we redefine the intermeeting time concept between nodes and introduce a new link metric called conditional intermeeting time. It is the intermeeting time between two nodes given that one of the nodes has previously met a certain other node. This updated definition of intermeeting time is also more convenient for the context of message routing because the messages are received from a node and given to another node on the way towards the destination. Here, conditional intermeeting time represent the period over which the node holds the message. To show the benefits of the proposed metric, we propose conditional shortest path routing (CSPR) protocol in which average conditional intermeeting times are used as link costs rather than standard intermeeting times and the messages are routed over conditional shortest paths (CSP). We compare CSPR protocol with the existing shortest path (SP) based routing protocol through real trace- driven simulations. The results demonstrate that CSPR achieves higher delivery rate and lower end-to-end delay compared to the shortest path based routing protocols. This shows how well the conditional intermeeting time represents internode’s link costs (in the context of routing) and helps making effective forwarding decisions while routing a message. Routing algorithms in DTN’s utilize a paradigm called store-carry-and-forward. We generated the multiple messages from a random source node to a random destination node at each t seconds. Clearly, CSPR algorithm delivers more messages than SPR algorithm.

4. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it’s constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

a) Main Modules:
Shortest path routing protocols for DTN’s are based on the designs of routing protocols for traditional networks. Messages are forwarded through the shortest paths between source and destination pairs according to the costs assigned to links between nodes. Two common metrics used to define the link costs are minimum expected delay (MED) and minimum estimated expected delay (MEED). They compute the expected waiting time plus the transmission delay between each pair of nodes.

Routing decisions can be made at three different points in an SP based routing: i) at source, ii) at each hop, and iii) at each contact. In the first one (source routing), SP of the message is decided at the source node and the message follows that path. In the second one (per-hop routing), when a message arrives at an intermediate node, the node determines the next hop for the message towards the destination and the message waits for that node. Finally, in the third one (per-contact routing), the routing table is recomputed at each contact with other nodes and the forwarding decision is made accordingly. In these algorithms, utilization of recent information increases from the first to the last one so that better forwarding decisions are made; however, more processing resources are used as the routing decision is computed more frequently.

b) Network Model:

We model a DTN as a graph \( G = (V, E) \) where the vertices (\( V \)) are mobile nodes and the edges (\( E \)) represent the connections between these nodes. However, different from previous DTN network models, we assume that there may be multiple unidirectional (\( E_u \)) and bidirectional (\( E_b \)) edges between the nodes.

c) Conditional Shortest Path Routing

Our algorithm basically finds conditional shortest paths (CSP) for each source-destination pair and routes the messages over these paths.

SIMULATIONS:

To evaluate the performance of our algorithm, we have built a discrete event simulator in Java. In this section, we describe the details of our simulations through which we compare the proposed Conditional Shortest Path Routing (CSPR) algorithm with standard Shortest Path Routing (SPR). Moreover, in our results we also show the performance of upper and lower performance boundaries with Epidemic Routing and Direct Delivery. We used two different data sets: 1) RollerNet traces which includes the opportunistic contacts of 62 iMotes which are distributed to the rollerbladers during a 3 hour skating tour of Paris on August 20, 2006, 2) Cambridge Dataset which includes the Bluetooth contacts of 36 students from Cambridge University carrying iMotes around the city of Cambridge, UK from October 28, 2005 to December 21, 2005. To collect several routing statistics, we have generated traffic on the traces of these two data sets. For a simulation run, we generated 5000 messages from a random source node to a random destination node at each \( t \) seconds. In RollerNet, since the duration of experiment is short, we set \( t = 1s \), but for Cambridge data set, we set \( t = 1 \) min. We assume that the nodes have enough buffer space to store every message they receive, the bandwidth is high and the contact durations of nodes are long enough to allow the exchange of all messages between nodes. These assumptions are reasonable in today’s technology and are also used commonly in previous studies. Moreover, we compare all algorithms in the same conditions, and a change in the current assumptions is expected to affect the performance of them in the same manner. We ran each simulation 10 times with different seeds but the same set of messages and collected statistics after each run.

5. RESULTS

The implementation can be shown in a stand alone system (OS: Windows XP) where the system acts as client and server.

6. CONCLUSIONS

We introduced a new metric called conditional inter meeting time inspired by the results of the recent studies showing that nodes’ inter meeting times are not memory less and that motion patterns of mobile nodes are frequently repetitive. Then, we looked at the effects of this metric on shortest path based routing in DTN’s. For this purpose, we updated the shortest path based routing algorithms using conditional intermeeting times and proposed to route the messages over conditional shortest paths. Finally, we ran simulations to evaluate the proposed algorithm and demonstrated the superiority of CSPR protocol over the existing shortest path routing algorithms.

ACKNOWLEDGEMENT

The authors express their deep gratitude to the Principal and the Management members of JPNCE for their encouragement and extensive support in preparing and publishing of this paper.
REFERENCES


[19] E. Daly and M. Haahr, Social network analysis for routing in disconnected delay-tolerant manets,