

# Efficient Routing using Bubble Rap in Delay Tolerant Network

Patel Jaimini  
Department of computer Engg.  
SPCE college, Visnagar  
GTU ,Gujarat ,India

Rajan Patel  
Department of Computer Engg  
SPCE college Engg Visnagar  
GTU, Gujarat , India

## ABSTRACT

Intermittently connected mobile ad-hoc networks are kind of wireless networks where end to end connectivity unlikely exists. Routing in such environment is carried out through the store-carry-forward mechanism. Thus, message delivery relies on node's mobility, their contact opportunities and contact patterns. Intermittently connected mobile ad-hoc networks can be viewed as dynamic graph, where mobile nodes are vertices and contact between nodes are edges. Such graph constantly evolves as node's connection, disconnection and reconnection and reveals the dynamic behaviour of the network. Detecting and dealing with congestion in delay tolerant networks is an important and challenging problem. Current DTN forwarding algorithms typically direct traffic towards particular nodes in order to maximize delivery ratios and minimize delays, but as traffic demands increase these nodes may become unusable.

In this thesis we attempted to improve BUBBLE RAP algorithm performance by using congestion control, forwarded message exchange and message priority update mechanism. We proposed Congestion Control algorithm (CCA) to reduce the congestion occurring on highly social nodes and to improve the forwarding efficiency significantly compared to oblivious forwarding schemes. In CCA algorithm, the congestion level of node and message priority are utilized to construct a Congestion Metric (CM). In the forwarding process, the concept of message forwarding to lower CM valued nodes is added together with the BUBBLE RAP strategy to improve the delivery probability as well as together with that a try to reduce the congestion by dropping the messages over nodes having low centrality links with destination is also made.

## Keywords

Quality of Node, Bubble Rap, Delay Tolerant Network, Conjection Level.

## 1. INTRODUCTION

Delay-tolerant networks (DTNs) are partitioned wireless ad hoc networks with intermittent connectivity. Within DTNs, a store and forward policy are often applied. Therefore, data are typically stored for extended periods of time before they are transmitted to their next hop. The goal, when routing decisions are made in DTNs, is typically to maximize the probability of the data reaching the final destination while minimizing a communication cost. Asymmetric data rates, high error rates, and long delays are key features of DTNs. Thus, these networks are not ideal to support real time services. Nevertheless, some kind of QoS should be ensured at some level, because critical data are often transferred over DTNs. Hence, there is a research area to design routing algorithm aim of select a path for better performance[1].

The DTN architecture proposes the introduction of another layer just above the transport layer, called as the bundle layer

in order to ferry data across the DTN [3]. Instead of packet-switching, DTNs use the concept of store-and-forward message (or bundle) switching [3]. This is because, at any given instant, there may not be any route to the next hop. In this case, the node must buffer the message in persistent storage, until a contact becomes available[2]. Once the next hop stores the bundle in persistent storage, it is said to have taken custody of the bundle, and the upstream node can delete its own copy of the bundle. Instead of waiting for the next hop to become available, the DTN gateways may themselves be mobile. This extension of the message switching concept is referred to as store-carry-forward routing [5].

Routing protocol in a DTN[4] is to provide a better delivery ratio with low overhead and small delay that is tolerable. A lot of routing approaches have been proposed in literature. The routing protocols in DTN can be divided broadly into two main categories: 1) Flooding Strategy 2) Forwarding Strategy. In Flooding Strategy Replicates the message to enough nodes so that the destination nodes must receive it. In Forwarding Strategy Uses knowledge about the network to select the best path (shortest one) to the destination.

We proposed to modify the existing Bubble rap algorithm in the category of congestion control social-based routing protocols. With the proliferation of mobile devices, exploring the social ties to develop routing algorithms that can improve the delivery rates and try to reduce congestion level of network has huge significance. The motivation behind this thesis is to develop such a routing algorithm that can improve the delivery performance in a social network as well as reduce the congestion level of network by forwarding in the congestion free nodes.

The rest of the paper is structured as follows. Section 2 discusses background theory Section 3 discuss related work goes over related work Section 4 proposed work and section 5 presents some conclusion and future work.

## 2. BACKGROUND THEORY

### 2.1. Basic Bubble rap Algorithm[7].

The basic concept behind the algorithm is the varying roles and popularities that people play in the societies. The first phase of the algorithm is to forward messages to nodes that are popular in the society. Popular nodes have frequent encounters with other nodes in the network and thus can relay the messages quickly. People form communities wherever they are set to live and therefore in the second phase of the algorithm nodes that are part of the destination node's community is detected to use them as relays.

The Bubble Rap algorithm is based on two main assumptions:

- Every node has to belong to at least one community and hence the Bubble Rap allows single node communities also.

- Every node in the network has two centralities. A global centrality depicting the popularity of the node within the network and a local centrality which shows the popularity of the node within the community.

### 2.1.1 Detecting Human Communities

Human society is characterized by a set of people forming socially meaningful relationships in which significant patterns can be recognized. These relationships can be grouped into communities. In a social network, the term community simply means a group of nodes that have high chances of meeting each other. The human community can be mapped into the DTN network because in a PSN devices are carried by humans and therefore the node movement depends upon the social characteristics of the people carrying these devices. For example, politicians of the same party form a community because their chances of encountering each other are very high taking into consideration a day.

How to detect the node communities in a PSN trace is the question that arises while designing a community based social-based forwarding algorithm. There are prominent centralized community detection algorithms in literature namely, K-Clique community detection algorithm by Palla et al.

### 2.1.2 Centralized K-Clique community detection

In Palla et al defines a K-Clique community as a union of all k-cliques (complete subgraphs of size k) that can be reached from each other through a series of adjacent k-cliques. Two k-cliques are said to be adjacent if they share k – 1 nodes. As k is increased, the k-clique communities shrink, but on the other hand become more cohesive since their member nodes have to be part of at least one k-clique. In the original Bubble Rap paper, the authors through repeated simulations have shown that a k=5 is an appropriate value for the algorithm.

### 2.1.3 Distributed Bubble Rap

A centralized routing algorithm will not work in practical applications and therefore the authors in [15] have suggested a distributed Bubble Rap algorithm. The algorithm is distributed in the sense that each node can detect its own community and calculate its own global/local centralities. For distributed community detection Hui et al has proposed three algorithms namely SIMPLE, K-CLIQUE and MODULARITY in. The authors proved in [29] that their distributed algorithms could achieve 85% accuracy as that of the centralized algorithm.

## 3. RELATED WORK

There are many proposed Social Based Routing Algorithms in DTN. One of the Popular routing is Label Routing. It was done by Hui and Crowcroft [8]. The algorithm was based on community labels in PSNs. To reduce high overheads caused by the blind forwarding techniques that cannot be afforded by a PSN, label routing uses labels as a property of nodes for forwarding. Each node has a label that denotes the community to which the node belongs to. For the purpose of deciding whether a message has to be forwarded to the encountered node or not, the label routing matches the label of the destination node of the message with the label of the encountered node. If both are the same, then the message is forwarded to the encountered node and it can act as a relay to take the message to its original recipient.

Another Popular Routing is SimBet Routing .The SimBet routing protocol was proposed by Daly and Haahr in [6]. This

algorithm uses between's centrality and similarity metrics for choosing a relay node that can deliver a message to the destination node. The between's centrality can identify good 'bridge' nodes that can facilitate good connections between nodes in the network and similarity can identify the node with most number of neighbours with the destination node.

Another Popular Routing is Bubble Rap forwarding. The Bubble Rap social-based forwarding algorithm was proposed by Hui et.al [7]. This routing protocol relies on two social metrics namely community and centrality. Each node in the network is supposed to have two centralities, global and local centrality. The global centrality depicts the popularity of the node across the network and the local centrality reflects the popularity of the node within its own community. Every node is assumed to belong to a community.

Another Popular Routing is Friendship based routing. It was Proposed by Bulut et al[9] using friendship-based communities. To calculate the friendship-based communities a new social metric termed as Social Pressures Metric (SPM) is introduced in. The approach considers friends as nodes which contact each other frequently and have long-lasting and regular contacts.

## 4. PROPOSED APPROACH

The proposed algorithm intends to address the following problem in the Bubble Rap social-based forwarding algorithm. In the original Bubble Rap algorithm that was, the main problem is the waiting of a message in a node if the carrying node is not a community member of the destination. If a popular node encounter does not happen, then the message waits for a long time resulting in either the expiry of the message or a long delay. Also, in Bubble Rap already delivered messages are not deleted from other nodes' buffer, and the highly social nodes also get congested in Bubble Rap resulting in dropping of messages. For solving the issue of congestion, the proposed algorithm uses the Social Congestion Metric calculation during each node encounter.

### 4.1 Algorithm to detect Congestion Level of Node

Since DTN is intermittently connected and the topology is dynamically changing, it is required that nodes can detect and control the global congestion of the network. In this thesis work, we use the ratio of dropping and receiving number of messages of the node as the metric to indicate the congestion level. It has proved that this metric can reflect the global congestion Degree of the whole network.

Node i calculates the number of dropping messages  $N_{drop}$  and the number of receiving messages  $N_{rec}$  in the period of time T, then use the ratio of the two values as the metric of congestion level, denoted by CL (i). To avoid the temporary fluctuation, we take the Exponentially Weighted Moving Average (EWMA) value of the current congestion level and the last one as the latest value. The latest congestion level is defined as formula:

$$CL(i) = \alpha \times \frac{N_{drop}}{N_{rec}} + (1-\alpha) \times CL(i)'$$

Where  $N_{drop}$  is the number of dropping messages,  $N_{rec}$  is the number of receiving messages.  $CL(i)'$  is the congestion level in the last period of time, and  $\alpha$  is the smoothing factor which equals to 0.9 to keep the freshness of congestion level .

**pseudo-code :**

```

Nrecv = 0;
Ndrop = 0;
CL = 0;
CLold = 0;
last_update_time = 0;
α = 0.5;
if (message received)
{
    ++ Nrecv;
}
if (message dropped)
{
    ++ N drop;
    updateCL();
}
updateCL ( )
{
    Avgk = Ndrop / Nrecv ;
    if (last_update_time == 0)
        CL = Avgk;
    else
        CL = α * CLold + (1 - α) * Avgk;
    timeDiff = (current_time - last_update_time);
    if (timeDiff >= 3600)
    {
        Nrecv = 0;
        Ndrop = 0 ;
        CLold = this time_CL;
        last_update_time = this _time;
    }
}

```

#### 4.2 Improve message priority base on TTL

- In Bubble Rap algorithm the main problem is, if the carrying node is not a community member of the destination then the message waits for a long time resulting in either the expiry of the message or a long delay. To overcome the problem we propose small algorithm.
- Source or Custodian node wait until some cut off value of message TTL to come in contact with message destination's community (LABEL) Node.
- If source or custodian node didn't come in contact with message destination's community in define time period.
- On contact with any other community node check the Global rank and congestion level and if it less then defined threshold value then forward it to

encounter node with multi hop multi copy forwarding mechanism.

#### 4.3 Exchange Deliver message vector

In the original Bubble Rap [15], the message is deleted from the original carrier once it enters the destination community member. However, it does not delete the delivered messages from other nodes' buffer in the network if they are still carrying it. We give the problem solution in following steps.

- Each node maintains two vectors one for delivered message and second for forwarded messages to its destination community.
- When two nodes of local community came in contact, first they exchange both vector.
- Base on new vector value they delete messages from buffer.

#### Pseudo code for Improve Bubble Rap

```

begin
    for each EncounteredNode_i do
        if (LabelOf(currentNode) == LabelOf(destination)) then
            if
                (LabelOf(EncounteredNode_i) == LabelOf(destination))
                and
                (LocalRankOf(EncounteredNode_i) >
                 LocalRankOf(currentNode))
            then
                EncounteredNode_i.addMessageToBuffer(message)
            else
                if (LabelOf(EncounteredNode_i) ==
                   LabelOf(destination) && CL < Threshold )
                or
                (GlobalRankOf(EncounteredNode_i) >
                 GlobalRankOf(currentNode) && CL < Threshold)
                Or
                (GlobalRankOf(EncounteredNode_i < Current) &&
                 (CL < Thresold) && Msg TTL < Cutoff )
                then
                    EncounteredNode_i.addMessageToBuffer(message)
            End

```

#### 5. CONCLUSIONS AND FUTURE WORK

In this paper, we have tried to overcome some limitations of Bubble Rap social-based forwarding in PSNs by using different mechanisms. The main contribution of this paper is the calculation of Congestion Level of encounter nodes which was not used in the original Bubble Rap algorithm. Base on congestion level forward messages and reduce the congestion of highly social nodes. Also, the naïve holding of messages in the buffer until a popular node is not encountered was removed by using a message TTL based threshold which forwards the messages from the buffer after a particular threshold. The exchange of Forwarded Message Vector (FMV) to delete those messages which are deleted provides efficient buffer management in an area like PSNs where buffer size is limited.

#### 6. REFERENCES

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets," in Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications - SIGCOMM '03, 2003.

- [2] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, "Delay-tolerant networking: an approach to interplanetary Internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, Jun. 2003.
- [3] F. Warthman, "Delay Tolerant Networks (DTN)," *prius.ist.osaka-u.ac.jp*, no. March, 2003.
- [4] A. Gupta, "Routing in Delay Tolerant networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems - SenSys '08*, 2009.
- [5] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges," *IEEE Communications Surveys &*
- [6] E. M. Daly & M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets", in *MobiHoc '07 Proc. 8th ACM International Symposium on Mobile ad-hoc networking and computing*, 2007.
- [7] P. Hui, J. Crowcroft & E. Yoneki, "Bubble rap: social-based forwarding in delaytolerant networks", *Mobile Computing, IEEE Transactions on*, vol. 10, pp. 1576-1589, November, 2011.
- [8] P. Hui & J. Crowcroft, "How small labels create big improvements", in *International Workshop on Intermittently Connected Mobile Ad-hoc Networks in conjunction with IEEE PerCom 2007*, March 19-23, 2007.
- [9] E. Bulut and B. K. Szymanski, "Friendship based routing in delay tolerant mobile social networks," in *Proc. IEEE Global Telecommuni-cations Conference (GLOBECOM)*, , Dec, 2010