Routing Algorithms in VDTN

Mr. Divyesh Devlani[#1], Prof. Milind Penurkar[*2]

[#] ME-II, Department of Information Technology and Engineering, MITCOE, University of Pune
[*]Associate Professor, Department of Information Technology and Engineering, MITCOE, University of Pune
MIT College of Engineering, Kothrud, Pune, India, 411038

[1]devlanidivyesh@yahoo.com,
[2]milindpenurkar@gmail.com

***Abstract*- Sometimes end to end connectivity may not exist, sometimes it is too expensive to connect all the networks and sometimes due to some catastrophic effect, exchanging of messages are difficult and sometimes it can cause delays from hours to days, networking in these type of situations is called DTN. This paper presents two totally different types of routing algorithms for Vehicular Delay Tolerant Networks (VDTN). VDTN is an extension of DTN, DTN stands for Delay Tolerant Networks. This paper describes Controlled Flooding Algorithm, in which flooding of messages will be controlled on group basis so that the algorithm can achieve better delivery ratio with less overhead. Second algorithm is Message Ferrying Algorithm in that individual ferry nodes will collect data from all the sender nodes and then deliver to the destination node when they meet to this node.**

***Keywords*- Delay tolerant network, Vehicular delay tolerant network, Routing algorithms and ONE.**

I. INTRODUCTION

A. DTN:

Delay Tolerant Networks (DTN) [1, 5] (also referred as Intermittently Connected Mobile Networks or Disruption Tolerant Networks), are wireless networks in which at any given time instance, the probability that there is an end-to-end path from a source to destination is unstable. Since most of the nodes in a DTN are mobile, the connectivity of the network is maintained by nodes only when they come into the transmission ranges of each other. If a node has a message copy but it is not connected to another node, it stores the message until an appropriate communication opportunity arises. There are two categories of DTN[1, 14] : Flooding and Forwarding, under which there are many routing algorithms which are used to develop DTN.

There are many examples of such networks in real life. For example, in north part of the Sweden [15], the communication between rural villages of India and some other poor regions. Other fields where this kind of communication scenarios may occur also include satellite communication [16], wildlife tracking [17], military networks and vehicular ad hoc networks [4, 18]. Moreover, such environments can exist even when a stable infrastructure is destroyed by natural disaster or other effects, another example of DTNs is the applications where sensors are attached to seals and whales [19] to collect large number of sensor readings from the oceans. In these applications, the data collected by sensors on seals and whales is transferred to a sink node using the transitive connectivity between the sensor nodes.

As shown in Figure 1, there are three methods used to transfer data between the village and the city: 1st one is traditional modem links, 2nd one is low-earth orbit satellite links, and 3rd one is a mobile “commuter bus” link that will carry messages physically.

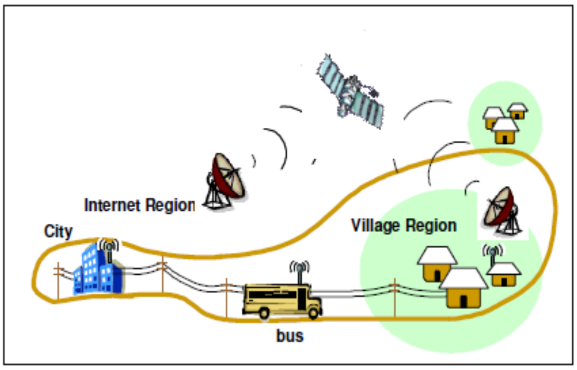

Fig. 1 Example of DTN network

This real-world situation[13] is abstracted here, the cities and villages can be represented by nodes and various connectivity options by edges. In this example if network is down or satellite and modem communication is down then the messages can be transmitted through bus or say through mobile node, here small village which doesn't have connectivity to any network then the only possible solution is to transmit data via bus.

B. Routing Algorithms:

There are mainly two types of routing strategies used : 1st one is Flooding[10] of messages and 2nd one is Forwarding[7, 14] of messages. Flooding[10] means to flood the message into the network, means there are more than one copy of same message exist in the network. Epidemic routing[12, 14] is the best example of flooding approach where source node will send same packet to all the nodes it encounters and intermediate mobile nodes will also forward that message to encountered node and so on, in this way message will be delivered to final destination. The overhead is quiet high and delivery of message is quiet good and implementation overhead is less so this is quiet useful method. While in Forwarding[7, 14] approach source node will send a message to only one mobile node when it encounters so only one single copy of message exists in the network. The best example of forwarding is First Contact router[14]. In this case source will generate the message and will send it to first encountered mobile node and source will delete the message from its buffer, so that only one single copy exist in the network. Another algorithm is spray and wait[3], in which flooding approach will be used, but it will flood number of copies only so, overhead will be decreased.

C. VDTN

VDTN stands for Vehicular Delay Tolerant Networks, it is an extension for DTN. In VDTN, nodes buffer and carry packets during network partitions, and forward packets to other nodes when they meet. This store-carry-forward paradigm is suitable for vehicle delay tolerant applications such as sensor data collection, messaging and file transfer in rural areas. VDTNs extend VANETs (i.e. Vehicular Ad Hoc Networks) with DTN capabilities to support long disruptions in network connectivity. The DTN concepts are useful as vehicular networks are characterized by scarce transmission opportunities and intermittent connectivity, particularly in rural or mountainous areas.

II. RELATED WORK

A. Epidemic Router:

Epidemic router is an algorithm[2, 12] that was firstly introduced by Demers et al. to achieve message delivery in spare networks like DTNs. In this algorithm replication mechanism is used for synchronization of database. It was modified by Vahdat et al. [2] and finally proposed as a flooding-based forwarding algorithm for DTNs. In the epidemic routing, all nodes can be source nodes and all nodes can be destination nodes. In this algorithm when a message is created, it will forward that message to all mobile nodes it encounters, hence all the nodes will keep same message so that number of copies of the message in the network will be more and overhead is much greater. The guaranteed delivery is not provided in this algorithm, epidemic routing algorithm can be seen as the best-effort approach to reach the destination. To avoid duplication every message is stored with its unique identifier that are stored at nodes buffer, list table is called summary vector. All nodes first exchange their summary and compare and update the summary vector and then they will forward messages. There are many buffer management policies in which buffer full condition can be handled efficiently, there are mainly two policies provided that are random and FIFO. In this case random policy will be used.

B. Spray And Wait Router:

Developers introduce a new routing scheme, called Spray and Wait[3], that “sprays” a number of copies of single message into the network like one, two or more number of copies depending on network conditions, and then “waits” till one of these nodes meet the destination and it will deliver the message. In this routing approach overall overhead is much lesser then epidemic[2, 12] routing and grater then first contact routing[14], but message delivery ratio is high as compared to both algorithms. In this algorithm source will generate a message and then it will forward a copy of message to peer node or we can say next connected mobile node, and then source node will again send another copy of same message to another connected mobile node and then source will discard that message from its buffer.

C. First Contact Router:

First contact router [14] uses only a single copy of the message and forwards it to the first available contact. In this router source node will generate the message with destination address, after that it will send that message to first encountered node and then delete the message from source node. Guaranteed delivery is not provided here. This algorithm is implemented to achieve less overhead because it has only one copy of message in the network.

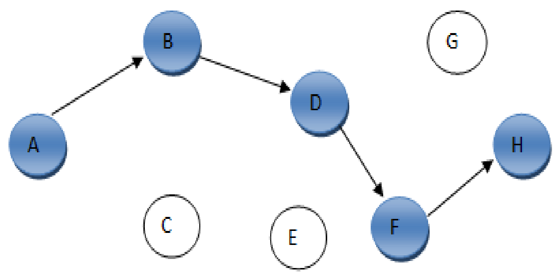

Fig. 2 First Contact Routing

As shown in this figure 2, A is the source node and H is the destination node, message path is A B D F H, where A will send message to B because B is first connection of A, after creation of message.

III. PROPOSED ALGORITHMS

This paper proposes two algorithms, first one is Controlled Flooding Algorithm and second one is Message Ferrying Algorithm.

A. Controlled Flooding:

Algorithm:

1. Divide n mobile nodes into m groups;
2. for each message i in S’s buffer
    Scan destination address of the message;
3. Find hop_count of a message;
    if hop_count is grater then 2 then set mgsttl to 0;
4. for each mobile node that comes in contact with another Mobile node
    If destination address and encountered mobile node address are in same group
        Send message to encountered mobile node;

else
do nothing;

Description of Algorithm:

This algorithm will control flooding of messages on group basis[6]. Message will be sent to destination group member only. Suppose there are 5 groups of mobile nodes. if one mobile node wants to send a message to another mobile node which is in another group, then it will send that message to destination group member only. In Controlled Flooding suppose there are 5 groups of mobile nodes, each group contains 20 nodes, so there are total 100 mobile nodes. Suppose there are 5 groups A, B, C, D, E, suppose any mobile node from group A wants to send a message to a mobile node in group B, then source node will send that message to group B members only, it will not send that message to any other group member.
Hop count of message will also be calculated in this algorithm, hop count means number of nodes through which message has passed. If hop count is greater than 1 then msgttl(message time to live) will be set to 0, so message will be discarded soon.

B. Message Ferrying:

Algorithm:

1. Divide n static nodes into m groups;
2. Designate one of the nodes as ferry node from mobile nodes to each group;
3. for each message i in S's buffer
Scan destination address of the message;
4. for each ferry node that comes in contact with a source node
If destination address and ferry node address are in same group
Send message to ferry node and
Delete message from Source node;
else
do nothing;

Description of Algorithm:

In Message Ferrying Algorithm there are set of mobile nodes which has specific known route and they are called Ferry nodes. In this approach all nodes are static nodes except Ferry nodes and all nodes are divided into groups. All the nodes are totally disconnected from each other. Ferry nodes are only mobile nodes and they travel through map and reach to all static nodes, each group has one individual Ferry node.

If some static node wants to send a message to another static node which is in another group then it will generate the message and will wait till ferry node comes in contact. After connecting with ferry node, source node will check ferry node group id. If ferry node group id is same as destination node group id then source will send that message to that ferry node else it will do nothing. After receiving message, ferry node will travel through map and will deliver message when it comes in contact with destination node.

IV. SIMULATOR USED

There are many types of simulators available for implementing algorithms like QualNet, NS2, ONE and many more. For DTN, ONE simulator[8, 9, 11] will be used. ONE stands for Opportunistic Network Environment. ONE is java based simulator specially designed to implement DTN algorithms.

V. SIMULATION AND RESULTS

Simulation and results are based on three parameters:

i. **Message Delivery Ratio:** In a delay-tolerant network, the most important network performance metric is the delivery ratio. However, in DTNs, a message is rarely actually "lost." Rather, the network was unable to deliver messages within an acceptable amount of time. Thus, we define the delivery ratio as the fraction of generated messages that are correctly delivered to the final destination within a given time period.

ii. **Overhead Ratio:** Second parameter is overhead ratio, it is calculated from following equation

$$\text{Overhead} = \frac{\textit{messages generated} - \textit{messages delivered}}{\textit{messages delivered}}$$

iii. **Average Delivery Latency:** Third parameter is the latency, it is the time between message generation and message reception. This metric is important since many applications can benefit from a short delivery latency, even though they will tolerate long waits. Many applications also have some time window where the data is useful. For example, if a DTN is used to deliver e-mail to a mobile user, the messages must be delivered before the user moves out of the network.

A. Controlled Flooding:

SCENARIO SETTINGS: For this algorithm 4500×3400 m^2 size of map with 100 mobile nodes will be used and there are five different node groups A, B, C, D, E each group has 20 mobile nodes, time of simulation is 21600 seconds (12 hours), Each message will be generated in 40 to 60 seconds, size of the message varies from 500KB to 1MB. MapBasedMovement movement model is used due to which mobile node will move randomly in the map. Group A is source node group and group B is destination node group, msgttl is 300 minutes. The settings used to run the simulations are summarized in Table 1.

TABLE I
SIMULATION SETTINGS

Simulation time	21600 (6Hrs)
Transmission speed	250kBps
Transmission range	80 Meters
Node speed	2.5 to 4.0 m/s
Buffer size	25 MB

SCREEN SHOT:

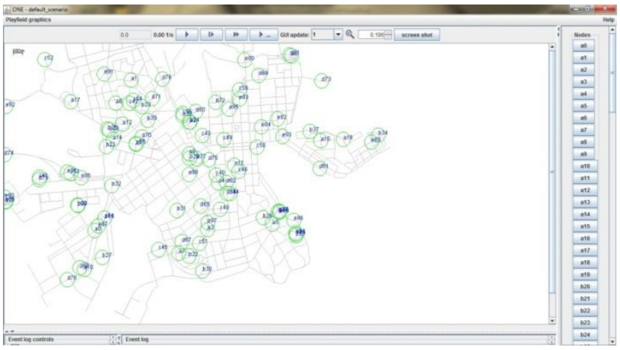

Fig. 3 Screen Shot for Controlled Flooding

RESULTS: All four routers Epidemic Router, Spray And Wait Router, First Contact Router and Controlled Flooding Router will use this simulation settings, then the results for all four routers are shown below in charts. As shown in figure 4 controlled flooding algorithm has good delivery ratio like Epidemic router, and as shown in figure 5 overhead is lower as compared to Epidemic Router, then figure 6 describes the average latency.

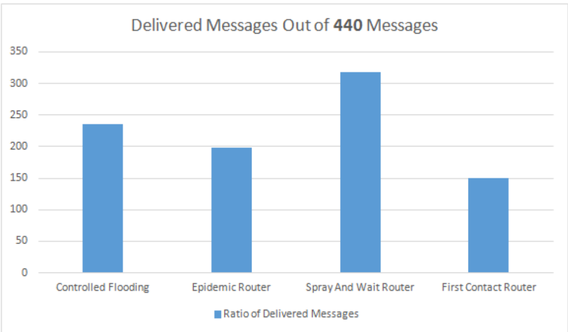

Fig. 4 Message Delivery Ratio for Controlled Flooding

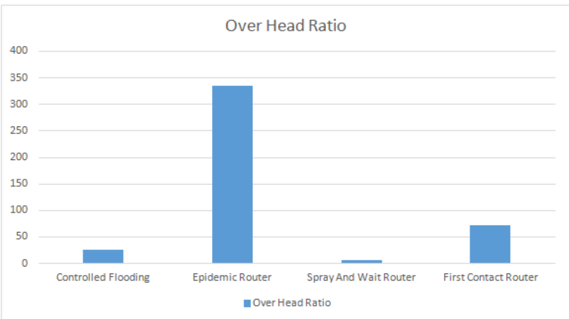

Fig. 5 Overhead ratio for Controlled Flooding

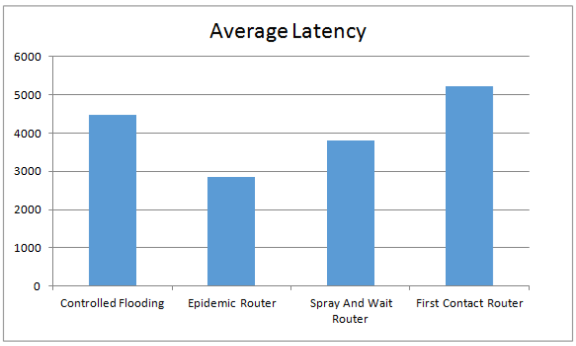

Fig. 6 Average Latency for Controlled Flooding

Implementation of Controlled Flooding is quite easy and it has less work to do and it gives good results as compared to other algorithms. Spray and Wait Router has also good results but it has much implementation overhead as compared to Controlled Flooding Router. Control Flooding Router has less implementation overhead and good results.

B. Message Ferrying:

SCENARIO SETTINGS: For this algorithm 4500×3400 m^2 size of map with 80 stationary nodes and 4 mobile nodes as ferry nodes will be used and there are four different node groups A, B, C, D each group has 20 stationary nodes and each group has one ferry node , time of simulation is 21600 seconds (12 hours), Each message will be generated in 150 to 170 seconds, here size of the message varies from 500KB to 1MB, MapRouteMovement movement model is used due to which ferry node will travel via specific fixed route. In this scenario any node can be source node and any node can be destination node except ferry node, msgttl is 300 minutes. The settings used to run the simulations are summarized in Table 2.

TABLE II
SIMULATION SETTINGS

Simulation time	21600 (6Hrs)
Transmission speed	250kBps
Transmission range	100 Meters
Buffer size	25 MB
Ferry Node Speed	4.5 to 9.5 m/s
Number of nodes	84

SCREEN SHOT:

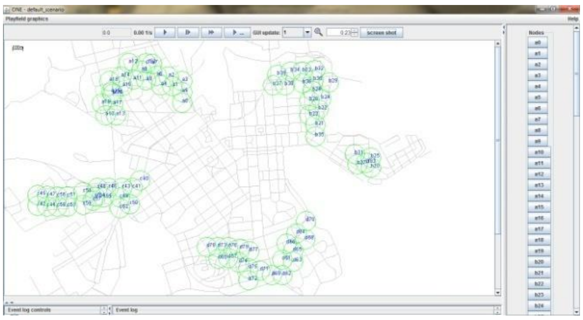

Fig. 7 Screen Shot for Controlled Flooding

RESULTS: All four routers Epidemic Router, Spray And Wait Router, First Contact Router and Message Ferrying Router will use this simulation settings, then the results for all four routers are shown below in charts. As shown in figure 8 Message Ferrying Algorithm has good delivery ratio, figure 9 shows overhead which is quite lower and then figure 10 describes average latency comparison with other existing algorithms.

Message Ferrying algorithm works for small village areas where internet connectivity is not available.

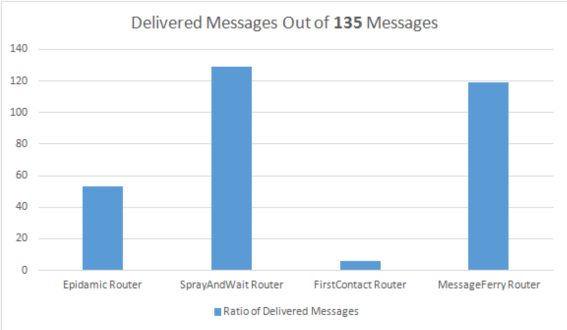

Fig. 8 Message Delivery Ratio for Message Ferrying

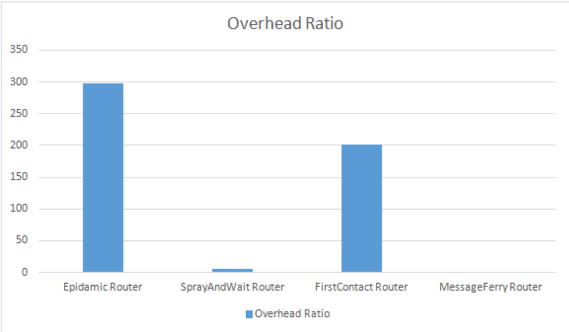

Fig. 9 Overhead ratio for Message Ferrying

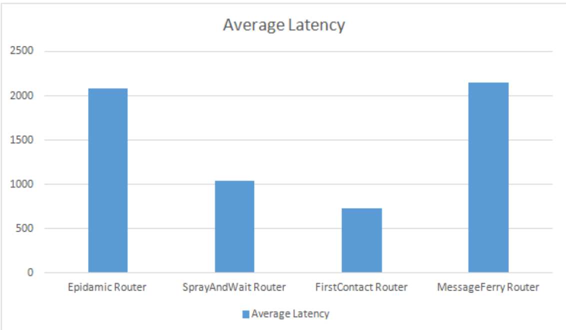

Fig. 10 Average Latency for Message Ferrying

In small villages regular buses are there, we just have to install small router on the bus so if someone wants to send a message to another person then he/she will generate the message and then store the message in its buffer, when particular bus comes in contact with source node then he/she will send that message to the router of that bus and then router will store the message in its buffer. Bus will deliver the message when it reaches the destination node. In this way we can connect small villages. It is ideal in catastrophic conditions when entire communication goes down. There are so many other applications for this approach.

VI. CONCLUSION AND FUTURE WORK

Delay tolerant networking (DTN) is a network where no end-to-end connectivity from source to destination exists, transmission of messages can occur by using above routing algorithms.

- By using controlled flooding algorithm one can minimize overhead and can increase message delivery ratio with less implementation overhead.
- By using message ferrying algorithm one can connect totally disconnected rural areas and can make communication possible. Message delivery ratio is quite high and too much overhead will be reduced here.

REFERENCES

- Mengjuan Liu, Yan Yang, and Zhiguang Qin, "A Survey of Routing Protocols and Simulations in Delay-Tolerant Networks", Springer-Verlag Berlin Heidelberg 2011.
- Amin Vahdat and David Becker ,"Epidemic Routing for Partially-Connected Ad Hoc Networks", Department of Computer Science Duke University
- Thrasyvoulos Spyropoulos, Konstantinos Psounis, Cauligi S. Raghavendra "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks", SIGCOMM’05 Workshops, August 22–26, 2005, Philadelphia, PA, USA.
- Paulo Rog´erio Pereira, Joel J. P. C. Rodrigues, Joan Triay, "From Delay-Tolerant Networks to Vehicular Delay-Tolerant Networks", IEEE 2011
- Kevin Fall, "A Delay-Tolerant Network Architecture for Challenged Internets" SIGCOMM’03, August 25-29, 2003
- Padma Mundur, Matthew Seligman, "Delay Tolerant Network Routing: Beyond Epidemic Routing"IEEE 2008
- Vijay Erramilli, Mark Crovella, "Forwarding in Opportunistic Networks with Resource Constraints", CHANTS’08, September 15, 2008, San Francisco, California, USA.
- Ari Keränen, Jörg Ott, Teemu Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation", SIMUTools 2009, Rome, Italy.
- Xu Liu _and Yuanzhu Chen, "Report of A DTN Simulator - THE ONE", Department of Computer Science Memorial University of Newfoundland, May 11, 2013
- Khalil Massri, Alessandro Vernata, Andrea Vitaletti, "Routing Protocols for Delay Tolerant Networks a Quantitative Evaluation", MSWiM’12, October 21–25, 2012, Paphos, Cyprus.
- http://www.netlab.tkk.fi/tutkimus/dtn/theone/.
- A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry,"Epidemic Algorithms for Replicated Database Maintenance",Sixth Symposium on Principles of Distributed Computing, pages 1–12, August 1987.
- Michael Demmer, Eric Brewer, Kevin Fall, Sushant Jain, Melissa Ho, Robin Patra, "Implementing Delay Tolerant Networking", IRB-TR-04-020, Dec.28 2004
- Evan P.C. Jones, Paul A.S. Ward, "Routing Strategies for Delay Tolerant Networks", University of Waterloo 200 University Avenue West Waterloo, Ontario, Canada.
- A. Doria, M. Uden, and D. P. Pandey, "Providing connectivity to the saami nomadic community", 2nd International Conference on Open Collaborative Design for Sustainable Innovation (dyd 02), Bangalore, India, Dec 2002.
- G. E. Prescott, S. A. Smith, and K. Moe, "Real-time information system technology challenges for NASAs earth science enterprise", 20th IEEE Real-Time Systems Symposium, Phoenix, Arizona, Dec 1999.
- P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet", ACM ASPLOS, 2002.
- J. Ott and D. Kutscher, "A disconnection-tolerant transport for drive-thru internet environments",IEEE INFOCOM, 2005.
- T. Small and Z. Haas, "The shared wireless infostation model - a new ad hoc networking paradigm (or where there is a whale, there is a way)", Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003), June 2003.

