ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)


# Ways of Disseminating Messages in Delay Tolerant Networks

[1]Sweety Soni, [2]Shruti Gupta
[1,2]RPIIT College Karnal, Harayana, India

**Abstract**

Delay-Tolerant Networks (DTNs) have the great potential to connecting devices and regions of the world that are presently under-served by current networks. A vital challenge for Delay Tolerant Networks is to determine the routes through the network without ever having an end to end, or knowing which "routers" will be connected at any given instant of time. The problem has an added constraint of limited size of buffers at each node. In this project we try to maximize the message delivery rate without compromising on the amount of message discarded. The amount of message discarded has a direct relation to the bandwidth used and the battery consumed. The more the message discarded more is the bandwidth used and battery consumed by every node in transmitting the message. we have proposed an algorithm where the messages are disseminated faster into the network with lesser number of replication of individual messages. The history of encounter of a node with other nodes gives noisy but valuable information about the network topology. Using this history, we try to route the packets from one node to another using an algorithm that depends on each node's present available neighbors'/contact and the nodes which it has encountered in the recent past.

**Keywords**

Delay Tolerant Network, Route Metrics, Routing

## I. Introduction to DTN

Today's Internet has been very successful at connecting communicating devices round the globe. It has been made possible by using a set of protocols, which is widely known as TCP/IP protocol suite. Every device on the innumerous sub-networks that comprise the Internet uses this protocol for transferring the data from source to destination with the minimal possible delay and high reliability. If at any instant there is no path between the sources to destination, then TCP/IP fails to work properly or might even stop working completely [11-12]. Delay Tolerant Networks (DTNs) enable data transfer when mobile nodes are only intermittentlyconnected.DTN routing relies on mobile nodes to forward packets for each other, the routing performance (e.g., the number of packets delivered to their destinations) depends on whether the nodes come in contact with each other or not.

## II. Concept of DTN

A Delay Tolerant Network can be considered as an overlay on the existing regional networks. This overlay is called as the bundle layer. This layer is intended to function above the existing protocol layers and provide the function of a gateway when two nodes come in contact with each other. The main advantage of this kind of protocol is flexibility. It can be easily linked with the already existing TCP/IP protocol networks or can be used to link two or more networks together. The position of the bundle layer can be seen in the following fig. 1.

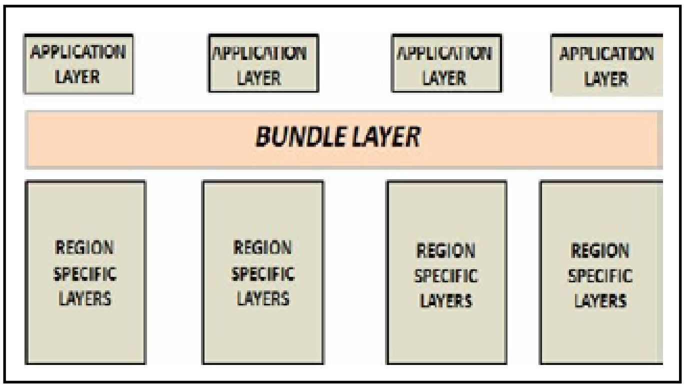


Fig. 1: The Position of the Bundle Layer

Bundles are also called as messages. The transfer of data from one node to another can be made reliable by storing and forwarding entire bundles between nodes. The bundles comprise of three things, source node's user-data, control information (e.g., source node ID, destination node ID, TTL etc.), a bundle header. Besides Bundle transfer, custody transfer is also done. The custodian node for a bundle keeps the message until it is successfully transferred to the next node and it takes the custody for that message or until the TTL of the message expires.

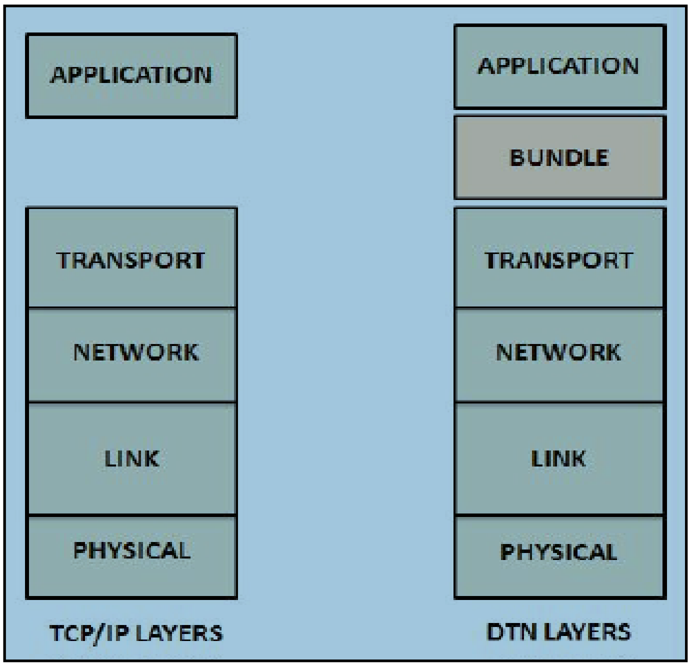


Fig. 2: Comparison of TCP/IP and DTN layers

### A. Related Work

Work on routing protocols for multi-hop wireless networks shows that it is possible to automatically route in networks, even when nodes are mobile and the link quality varies. There is a huge body of work on routing protocols [1, 12-13, 15] and metrics [4-5, 16] for this environment. However, these protocols and metrics find end-to-end paths, and do not support communication between nodes in different network partition. One of the earliest proposals for routing in disconnected networks is epidemic routing [19]. It

 ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

relies on replicating messages through random exchanges between nodes until all nodes have a copy of every message. Each node has a buffer where it stores these messages. When it comes into contact with another node, the two nodes exchange messages until their buffer contents are synchronized. This approach can achieve high delivery ratios, and operates without knowledge of the communication pattern. It is well-suited to networks where the contacts between nodes are unpredictable. Unfortunately, it is very expensive in terms of the number of transmissions and buffer space. In particular, it does not appear that this approach can scale as the number of messages in the network grows. The critical resource in epidemic routing is the buffer. An intelligent buffer management scheme can improve the delivery ratio over the simple FIFO scheme [3]. The best buffer policy evaluated is to drop packets that are the least likely to be delivered based on previous history. If node A has met B frequently, and B has met C frequently, then A is likely to deliver messages to C through B. Similar metrics are used in a number of epidemic protocol variants [3, 10, 18]. This approach takes advantage of physical locality and the fact that movement is not completely random. However, these protocols still transmit many copies of each message, making them very expensive. An approach that uses a single copy of each message is presented by Jain et al. [9]. They assume that the contact schedule is completely known in advance, and use this knowledge to create a number of routing metrics. Their results show that the efficiency and performance increases with the amount of information used for the metric. The weakness of this approach is that each node must have access to accurate schedule data. To provide this information, the routing must be manually configured with the contact schedules, which must be repeated each time the schedule changes. Handorean et al. explore alternatives for distributing connectivity information, but they still assume that each node knows its own connectivity perfectly [7]. It is unclear how the performance will be affected if the schedules are imprecise, which is the typical case in the real world, or if there are failures which alter the communication pattern.

## III. Routing Design

Our routing Algorithm focuses mainly on high delivery rate of messages in such a way that the average number of messages forwarded is the least. Here each node eith the help of Packet Tracer has to maintain a table which contains the details about the nodes which are currently in contact and the nodes which were in contact previously. The previously contact nodes should be associated with the time of last contact. The entries in the table are as follows, < node id, node availability, last contact time, list of nodes in contact with this node >.For example the table maintained by node NOD001 is

| Nodes in Contact | Availability | Last Contact time | Nodes in Neighbor's Contact |
|---|---|---|---|
| NOD003 | Y | 99 | NOD001, **NOD999**, NOD077, NOD055 |
| NOD006 | Y | 99 | NOD001, NOD098, NOD074, NOD545 |
| **NOD999** | N | 86 | NOD022, NOD938, NOD001, NOD834 |
| NOD675 | N | 74 | NOD883, NOD001 |
| ... | ... | ... | ... |

Fig. 3: The Table Maintained by NOD001

## Algorithm

Let the set of Nodes be N.
Let Mji be the jth message in the buffer of Ni.
For each Node Ni € N do
For each Message $M_{ji}$ € Buffer ($N_i$)
If (Dest(Mji) €Available_Contacts(Ni))
- Forward Message to Dest($M_{ji}$)

Else if (Dest($M_{ji}$) €
Available_Contacts(Available_Contacts (Ni)))
- Forward Message to $N_f$, where Dest($M_{ji}$) € Available_Contacts($N_f$)

Else
- Forward Message $M_{ji}$ to a moving node Nm via $N_f$,

where $N_f$ € Available_Contacts ($N_i$)
and $N_m$ Available_Contacts ($N_f$)
- ∀ (Available_Contacts ($N_i$)

∪Available_Contacts ($N_f$) )
//Do_Not_Recieve($M_{ji}$)
- If ($N_m$ does not exist)

//Forward Message $M_{ji}$
to a moving node Nmax where $N_{max}$ = Maximum_dissimiliar_contacts (Ni, $N_{max}$)
- ∀ (Available_Contacts (Ni) ∪ Available_Contacts (Nmax) )

//Do_Not_Recieve ($M_{ji}$)

## Explanation

1. Let us take the fig. 2 as a reference. First the forwarder/sender node checks if the destination is in contact with it. If yes then forward it. Else go to step 4.
2. Then check if the destination node is present in Nodes in Neighbor's Contact column. If yes then forward the message to the corresponding Node in contact. Else go to step 4.
3. If the forwarder/sender node (eg: NOD999) was available to the sender node (eg: NOD001) some time back, and this node (eg: NOD999)is now available to another node(eg: NOD003) in contact with the sender, then it infers that that particular node(eg: NOD999) is on the move, away from the sender. Thus the packet is forwarded to this node (eg: NOD999).
4. If a forwarder/sender node does not find any mobile nodes, then it should forward to those in-contact nodes which are farthest from it. The node which has the maximum number of dissimilar nodes in contact (comparing with that of sender) is the most distant.
5. A node is prohibited from sending the packet to the previous node or its contact nodes. This helps further in pushing the packets towards the destination. For Example, Node S sends the data via Node B to a moving Node X(Note the final destination is not Node X but Node D).Then the Node X is not allowed to forward the packet to,

- Node S or Node S's current in-contact node.
- Node b or Node B's current in-contact node.
- Current in-contact nodes of (Common in-contact nodes of S and B)

6. Similarly, every node tries to forward the packet to a candidate set of nodes which are likely to be farthest from it or moving away from it. This way the packet reaches to the destination.

## IV. Simulation & Results

### A. Simulation

The simulation for the proposed algorithm were done using,

- 15 Nodes in a region of 25 X 25 grid.
- 30 Nodes in a region of 50 X 50 grid.

In the grid the nodes were randomly deployed. Every node has a set of destinations to travel sequentially, but the path taken by the nodes to reach these destinations were again random. Messages were injected into the system randomly at any instant during the simulation. Using MATLAB, the graphs plotted were for,

- Max. Hop Limit vs. No. of Message Delivered (out of 100).
- Max Hop Limit vs. Average No of Messages Discarded.

### B. Results

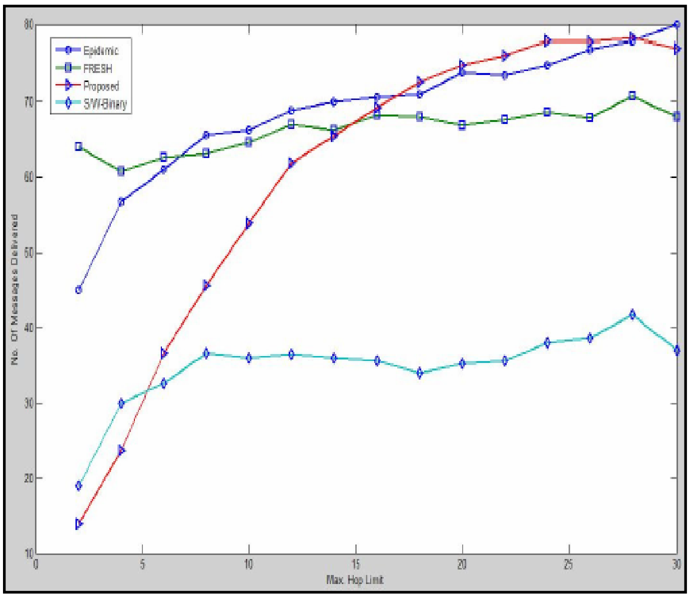


Fig. 4: Max Hop Limit Vs. No. of Messages Delivered in

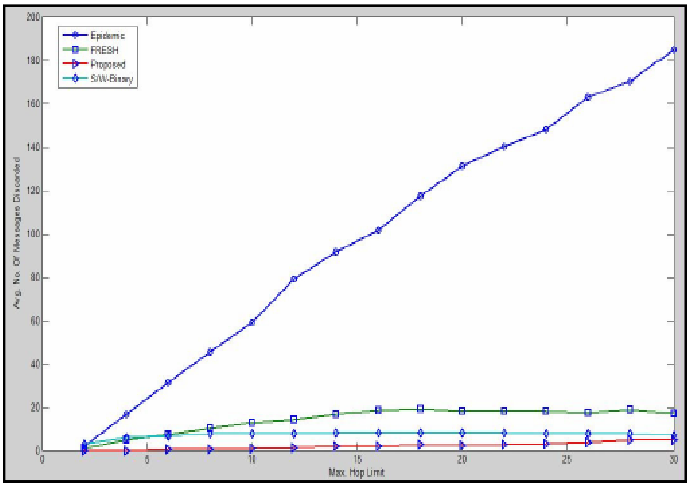


Fig. 5: Max Hop Limit vs. Average No. Messages Discarded in a 50X50 grid

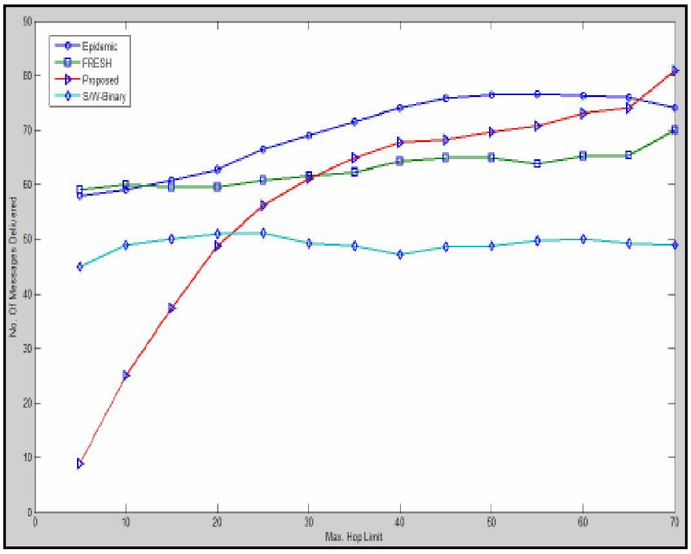


Fig. 6: Max Hop Limit vs. No. of Messages Delivered in a 50X50 grid

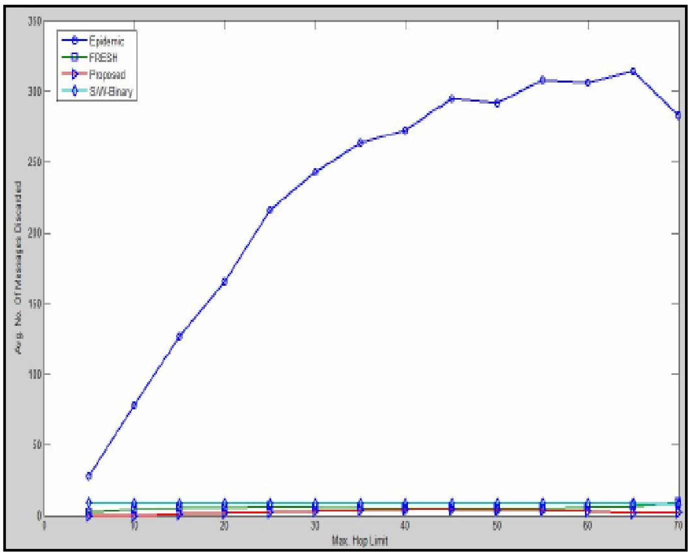


Fig. 7: Max Hop Limit vs. Average No. Messages Discarded in a 50X50 Grid

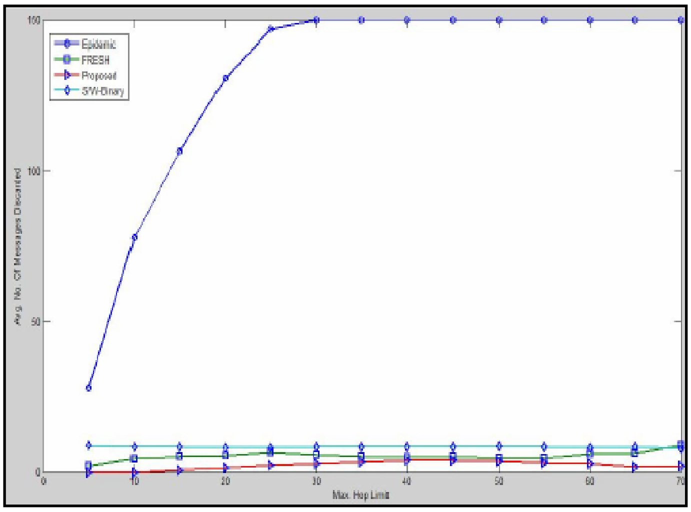


Fig. 8: Max Hop Limit vs. Average No. Messages Discarded in a 50X50 Grid (when the Avg. no. of Messages Discarded in Epidemic was Floored to 150)

### C. Analysis

As it can be observed from the graphs, the no. of messages delivered in the proposed algorithm gradually increases with the increase in the Maximum Hop limit. At the same time, the average number of messages discarded from was the least in the proposed algorithm. This has a direct impact on conserving the energy (e.g. battery power) of the mobile nodes and saving the bandwidth. However if the maximum hop limit is kept low, then the proposed algorithm does not perform well in the sphere of delivering messages to the intended destination.

## V. Conclusion

The history of encounters of a node with other nodes gives a vague but very important picture of the relative locations of the nodes in the network. In the proposed algorithm, we have tried to exploit this history along with the mobility pattern of the nodes. We have tried to strike a balance between high delivery of messages and low number of messages replication in the network. Since the nodes are generally held by rational entities like human beings, the social behavior of these nodes is also an important criterion and can play a pivotal role in improved delivery rates. In the future work, the social behavior of the nodes can be analyzed and included as a parameter in forwarding the message from one node to another

## References

[1] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, J. Jetcheva,"A performance comparison of multi-hop wireless Ad -Hoc network routing protocols", In ACM MobiCom, 1998.

[2] C. Perkins,"Ad-Hoc Networking", Addison Wesley, Reading, MA, 2001.

[3] Y. Wang, H. Wu,"DFT-MSN: The Delay Fault Tolerant Mobile Sensor Network for Pervasive Information Gathering", IEEE INFOCOM'06, Apr. 2006

[4] S. Dolev et al.,"Virtual Mobile Nodes for Mobile Ad Hoc Networks", 18th Int'l. Symp. Distributed Comp., DISC 2004.

[5] S. Jiang et al.,"A Prediction-based Link Availability Estimation for Mobile Ad-Hoc Networks", IEEE INFOCOM 2001.

[6] K. Fall,"A delay–tolerant network architecture for challenged internets", In ACM SIGCOMM, 2003.

[7] R. Handorean, C. Gill, G.-C. Roman,"Accommodating transient connectivity in ad hoc and mobile settings", Lecture Notes in Computer Science, 3001, pp. 305–322, March 2004.

[8] B. Hull, K. Jamieson, H. Balakrishnan,"Mitigating mcongestion in wireless sensor networks", In ACM SenSys, 2004.

[9] S. Jain, K. Fall, R. Patra,"Routing in a delay tolerant network", In ACM SIGCOMM, 2004

[10] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, D. Rubenstein,"Energy-efficient computing for wildlife tracking: design trade-offs and early experiences with zebranet".

[11] A. Lindgren, A. Doria, O. Schelen,"Probabilistic routing in intermittently connected Networks SIGMOBILE Mobile Computing and Communications Review", 2003, In Proc ASPLOS'02, October, 2002.

[12] T. Spyropoulos, K. Psounis, C. S. Raghavendra,"Single-copy routing in intermittently connected mobilenetworks", In Proc. of IEEE Secon'04, 2004.

[13] R. C. Shah, S. Roy, S. Jain, W. Brunett E,"Data mules: Modeling and analysis of a three-tier architecture for sparse sensonetworks", Elsevier Ad-Hoc Networks Journal, September, 2003.

[14] A. Lindgren, A. Doria, O. Schelen,"Probabilistic routing in intermittently connected networks", ACMSIGMOBILE CCR, 2003.

[15] B. Chen, C. Choon,"MobiCent: A credit-based incentive system for disruption tolerant network", In Proc. IEEE INFOCOM, 2010.

[16] S. Zhong, J. Chen, Y.R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile Ad-Hoc networks", In Proc. IEEE INFOCOM, 2003.

[17] U.Shevade, H. H.Song, L.Qiu, Y. Zhang,"Incentive-Aware routing in DTNs", IEEE, 2008.

[18] Kevin Fall,"A Delay-Tolerant Network Architecture for Challenged Internets", 2003

[19] F.Warthman,"Delay Tolerant Networks: A Tutorial", 2003.

[20] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss,"Delay-Tolerant Network Architecture", DTN Research Group Internet Draft, 2003.