ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 2, February 2014

Delay Tolerant Networks An Emerging Communication Paradigm

J.Dhivya[1], M.Vanitha Lakshmi[2]

PG Scholar, Department of PG Studies, S.A Engineering College, Chennai, India[1]

Assistant Professor, Department of PG Studies, S.A Engineering College, Chennai, India[2]

Abstract: The Delay Tolerant Networks (DTN) are networks where the end-to-end paths between source and destination are unstable or unlikely. In such networks, conventional path-discovery-based MANET routing techniques like AODV and DSR are not possible because the network may not form a single connected partition at any time, and thus a full path may never exist between the source and the destination. In such cases a store and forward algorithm is required, where the data packets can be stored in the network and be delivered before its deadline is over. Humans are not only users of this network, they are also service providers. It's because DTNs have been adopted in various fields ranging from mobile networks to exotic media networks such as satellite communication. This paper is brief study of algorithms related to DTNs and analysis of an efficient algorithm Backpressure from its origin to its developed versions at present. It also explains the methods that can improve the working of backpressure in high traffic as well as low traffic with reduced overloads at the nodes maintaining queues.

Key words: delay tolerant networks, store and forward, backpressure, queues

I. INTRODUCTION

A Delay Tolerant Network (DTN) is a network designed to operate effectively over extreme distances such as those in space communications or on an interplanetary scale. In such networks, long latency sometimes measured in hours or days is unacceptable. However, similar problems can also occur over more modest distances when interference is extreme or network resources are severely overburdened. Some of the remote areas which cannot be supported by current network technology can be well supported by DTN[2]. It is because they make use of temporary connections instead of depending upon end to end network conditions. These networks could be useful in scenarios ranging from interconnecting sensors to connecting remote regions of the world. Message transferred to a DTN node classified as persistent, has large amounts of non-volatile storage and can hold the message until the next communication opportunity, referred as custody transfer.

Delay tolerant networks are characterized by their lack of connectivity, high latency, long queuing delay, short range contact, disconnection where it is hard to find an instantaneous end-to-end. Routing protocol such as AODV and DSR first establishes a proper route and then transmits the data to its destination, hence such protocols fails in the challenging situations as that of faced by a DTN. At this point of time DTN's enter into store and forward approach, where the data is stored in the network until it reaches its final destination. In most infotainment and urban sensing applications, there exists a deadline, beyond which packets are considered obsolete. For example, in real time urban environment monitoring applications, data generated many hours ago may be considered useless since the environment may have changed greatly during that time. Therefore, such a deadline restriction should be taken into consideration when evaluating the DTN performance. Such a DTN finds its application in various fields such as satellite communication, ZebraNet, DAKnet, telemedicine, etc., where the data as to be stored and delivered to the destination. DTN's have various algorithms [4], [5], [6] to support store and forwarding nature of their network such as Epidermis, Spray and Wait, Maxprop, Prophet etc., which have their own advantages and disadvantages. Apart from this algorithms Backpressure based scheduling algorithms have been proved to be highly efficient even in the case of high traffic and loss of connectivity.

II. STORE AND FORWARDING DTN ALGORITHMS

DTN routing focuses on opportunistic approaches i.e., where no contact information is known priori, neighbor nodes can exchange some information before sending packets of data, and no network infrastructure exists to provide connectivity. DTN's have some special properties which makes it different from other networks. Properties of DTN's include

- High Latency - Any two nodes may never meet each other.
- Low Data Rate - Due to the long latency of data delivery.
- Disconnection - It is hard to find an end-to-end path.
- Long Queuing Delay - Because of the disconnection.
- Short Range Contact –Only one-hop communication is guaranteed.
- Dynamic Network Topology - Different types of user behavior will result in dramatically different network conditions.
- Interoperability considerations
- Low power requirements

When instantaneous end-to-end paths are difficult or impossible to establish, routing protocols must take to a "store and forward" approach, where data is incrementally

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 2, February 2014

moved and stored throughout the network in hopes that it will eventually reach its destination.

A. Epidemic Routing

A large number of routing protocols have been proposed based on various assumptions regarding the connectivity and mobility patterns. The first proposed routing protocol for DTN is Epidemic Routing Protocol. Its idea is almost the same as the flooding routing scheme in traditional ad hoc networks. When an intermediate node receives a message, it will broadcast to all its neighbors, and thus information disseminates in a flooding manner. It is intuitive to consider the Epidemic routing as an unconstrained optimization to maximize the performance of the throughput. However, in real world deployments, especially in large scale, resource constraint scenarios like the one we are facing in the urban DTN, careful consideration of the complicated tradeoff between the performance and the resource cost, and adaptation to the heterogeneity of the network is a must for sophisticated protocols.

B. MAXPROP

MaxProp is flooding-based routing protocol. In this algorithm packets that are not held by a node will be replicated and transferred. The intelligence of MaxProp lies in deciding which messages should be transmitted first and which packets should be dropped first. MaxProp maintains an ordered-queue based on the destination of each message, ordered by the estimated likelihood of a future transitive path to that destination. When two nodes meet, they first exchange their estimated node-meeting likelihood vectors. Ideally, every node will have an up-to-date vector from every other node. MaxProp uses several mechanisms to define the order in which packets are transmitted and deleted in order to better utilize the network resource. However, it is not explicitly addressed how to optimize a specific routing metric using MaxProp..

C. RAPID

RAPID, Resource Allocation Protocol for Intentional DTN routing. RAPID is designed to explicitly optimize an administrator specific routing metric, such as minimizing average delay, minimizing missed deadline, or minimizing maximum delay. RAPID translates the routing metric to per-packet utility and determines at every transfer opportunity if the marginal utility of replicating a packet justifies the resource used. However, RAPID requires the flooding of information about all the replicas of a given message in the queues of all nodes in the networks in order to derive the utility. In scalable networks such as the urban DTN, such information is difficult to achieve and information flooding also requires tremendous network resource.

D. SPRAY AND WAIT

Spray and Wait is a routing protocol that attempts to gain the delivery ratio benefits of replication-based routing as well as the low resource utilization benefits of forwarding-based routing. Spray and Wait achieves resource efficiency by setting a strict upper bound on the number of copies per message allowed in the network. The Spray and Wait protocol is composed of two phases: the spray phase and the wait phase. When a new message is created in the system, a number **L** is attached to that message indicating the maximum allowable copies of the message in the network. During the spray phase, the source of the message is responsible for "spraying", or delivery, one copy to **L** distinct "relays". When a relay receives the copy, it enters the wait phase, where the relay simply holds that particular message until the destination is encountered directly**.**

III. BACKPRESSURE SCHEDULING

Backpressure [7] is an efficient algorithm for DTN where, each node maintains a queue for each destination. A differential backlog is calculated between two nodes and the packets are transmitted in the direction of maximum weight. Hence in this algorithm links are scheduled rather than packet scheduling. Since the packet flows in the order of maximum weight, it looks as though if packets are pulled towards the destination. A key feature of the back-pressure algorithm is that packets may not be transferred over a link unless the back-pressure over a link is non-negative and the link is included in the picked schedule. This prevents the transmission of data packets to the nodes that are already congested, thus providing the adaptivity of the algorithm.

This algorithm defines the capacity region of the network as the set of all end-to-end traffic load matrices that can be stably supported under some network control policy. The stability means that all queues in the network have finite backlog. The larger the capacity region the better the performance will be, since the network will be stable for a wider range of traffic loads. The capacity region of the network is the union of the individual policy capacity regions over all possible control policies. A network policy is called throughput-optimal if its capacity region coincides with the network capacity region. In other words, a throughput optimal policy can stably support every end-to-end traffic load matrix in the network capacity region. The throughput would never increase if a node injects traffic at a rate that is outside the capacity region. In that case, the queues at intermediate nodes will overflow and the packets will be dropped before they reach the destination.

A. Maximum Weighted Backpressure

Consider a multi-hop network with N nodes. The network operates in slotted time t Є {0, 1, 2, . .}., when a New data enters into the network routing, transmission and scheduling decisions are made to deliver all data to its original destination. Let data that is destined for node k Є {1, 2, . . N} be labeled as commodity k data. Data in each node is stored according to its commodity. For n Є {1, 2, . . N} and k Є {1, 2, . . N}, let $Q_n^{(k)}t$ be the current amount of commodity k data in node n, also called the queue backlog. Figure.2 shows queue backlogs inside a node. The units of $Q_n^{(k)}t$ can be a integer units of packets, if the data is a segmented fixed length packet. It can also take real valued units of bits. It is assumed that $Q_n^{(k)}t = 0$ for all k Є {1, 2, . . N} , because no node stores data that is

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 2, February 2014

destined for itself. When the Data is transmitted from one node to another node, it is removed from the queue of the first node and added to the queue of the second. If the data had reached its final destination, it is removed from the network.

1. Node Selection

Node selection plays an important role in backpressure. A neighbor of node a is node b such that the transmission rate of the data that is to be sent over the link (a,b) is choosed from the available transmission rates of $^{\Gamma}$s(t). Probably a node can have all N − 1 other nodes as neighbors. Neighbor nodes can also be classified based upon geographical distances, by link connections, or nodes can be classified as neighbors that would have propagated signal strength below a certain threshold.

2. Optimal Commodity

The differential backlog quantity calculated between two nodes a and b is given by

$Q_a^{(k)}(t) - Q_b^{(k)}(t)$ (1)

Where c is defined as the commodity that is to be transferred over the link (a,b). Let $k_{ab}^{opt}(t)$be the optimal commodity transferred over the link, which is the commodity of the maximum of all the calculated differential backlogs.

After determining the optimal commodities, the network controller calculates the weight $W_{ab}(t)$ Where,

$W_{ab}(t) = \max[Q_a^{\left(k_{ab}^{opt}(t)\right)}(t) - Q_b^{\left(k_{ab}^{opt}(t)\right)}(t)]$ (2)

The weight $W_{ab}(t)$ is the value of the differential backlog associated with the optimal commodity for link (a,b).

On time slot t, let the rate at which packets are transferred over the link (a,b) be $\mu_{ab}(t)$. At every time slot t, the topological state of the network varies due to mobility of nodes and time varying channel conditions. Hence there must be choices for the selection of transmission rates depending upon the time varying network conditions. Let the topological state of the network be captured asS(t). Let $^{\Gamma}$s(t) contain different transmission rates upon varying network conditions , such that $^{\Gamma}s(t) = \{ \mu_a, \mu_b, \mu_c, \}$. Hence the optimal commodity that is to be transferred over the link is generalized as

$\text{Max}(\sum_{a=1}^{N}\sum_{b=1}^{N} W_{ab}(t)\mu_{ab}(t))$. (3)

In figure 2, there are three nodes n, k1 and k2. Node n maintains queues for different destinations such as k1 and k2. Similarly nodes k1 and k2 have their own destination queues.

The commodity that is to scheduled over the link (n,k1) is given by

$Q_n^{(1)}(t) - Q_{k1}^{(1)}(t) = 4$ (4)

$Q_n^{(2)}(t) - Q_{k1}^{(2)}(t) = 5$ (5)

$Q_n^{(3)}(t) - Q_{k1}^{(3)}(t) = -3$ (6)

From the above equation the maximum weight commodity is given by

$W_{(n,\,k1)}^{\max} = (\max\{5\text{-}1, 7\text{-}2, 2\text{-}5\})^{+} = 5$ (7)

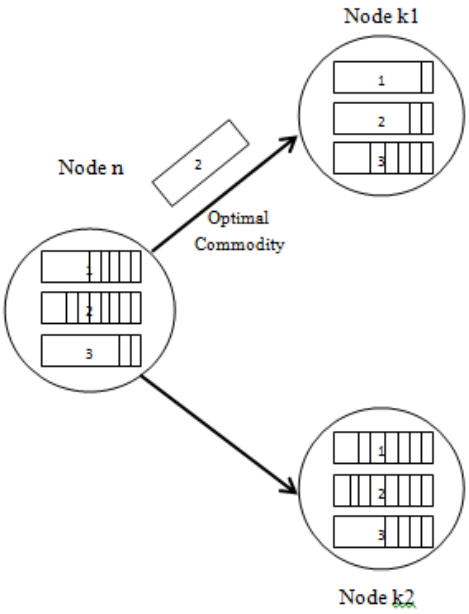

Fig1.Queues maintained at each nodes.

Let the transmission rates of the commodity 1,2,3 be given as $\mu_{1=2,}$ $\mu_{2=2,}$ $\mu_{3=1}$. From equation 4 the weights of the optimal quantity is calculated as

$\sum_{n=1}^{N}\sum_{k1=1}^{N} W_{nk1}(t)\mu_{nk1}(t) = 8$ (8)

$\sum_{n=2}^{N}\sum_{k1=2}^{N} W_{nk1}(t)\mu_{nk1}(t) = 10$ (9)

$\sum_{n=3}^{N}\sum_{k1=3}^{N} W_{nk1}(t)\mu_{nk1}(t) = -3$ (10)

Finally the optimal commodity that is to be transferred over the link is commodity 2 of node n. This commodity will be transmitted at a rate given by

$\mu_{ab}^{k}(t) = \mu_{ab}(t)$ if $k = k_{ab}^{opt}(t)$and (11)

$Q_a^{\left(k_{ab}^{opt}(t)\right)}(t) - Q_b^{\left(k_{ab}^{opt}(t)\right)}(t) \geq 0$

B. Greedy Maximal Scheduling (GMS) Backpressure Algorithm

GMS algorithm [8] requires the same queue structure as that of the traditional back-pressure algorithm. The difference between these two algorithms lies in the methods used for picking a schedule. Here the weights are queue lengths and not the backpressures. Let *S* denote the set of all links initially. Let *Nb*(*l*) be the set of links within the interference range of link *l* including *l* itself. At each time slot, the GMS algorithm picks a link *l* with the maximum weight first, and removes links within the interference range of link *l* from *S*, i.e., *S* = *S**Nb*(*l*); then it picks the link with the maximum weight in the updated set *S*, and so forth. This method reduces the computational complexity to a great extent but with the penalty of reduced network region capacity. Study of GMS indicate that there may be reduction in throughput in certain network topologies, it seems to perform well in practice.

The back-pressure algorithm has several disadvantages that prohibit practical implementation. The back-pressure algorithm requires maintaining queues for each potential destination at each node. This queue management requirement could be a prohibitive overhead for a large network. The back-pressure algorithm is an adaptive

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 2, February 2014

routing algorithm which explores the network resources and adapts to different levels of traffic intensity. However it might also lead to high delays because it may choose long paths unnecessarily. High delays are also a result of maintaining a large number of queues at each node, and each of those queues being large. The queues can be large because, under back-pressure algorithm, average size of a per destination queue at a node can grow with the distance from the node to the destination. Furthermore, large number of queues takes away statistical multiplexing advantage: since only one queue can be scheduled at a time, some of the allocated transmission capacity can be left unused if the scheduled queue is too short this can contribute to high latency as well.

C. Back-Pressure with Shadow Queues

In shadow-queue based back-pressure routing [9], each node maintains a fictitious queue called a shadow queue, which is just a counter, for each flow. Scheduling and forwarding decisions are made based on shadow queue sizes $q\hat{}_n^f$ instead of the real queue sizes $q_n^f(t)$. The shadow queues are updated in a similar manner to the real queues but with a shadow packet arrival rate that is slightly higher than the real packet arrival rate. For example, if the real packet arrival rate of flow f, λ^f is then the shadow packet arrival rate is $(1+\varepsilon)\lambda^f$, $\varepsilon > 0$. In order to ensure stability of the shadow queues, the shadow arrival rate $(1+\varepsilon)\lambda^f$ must lie in the interior of the network capacity region. Since the shadow queue size always upper bounds the real queue size, it follows that the real queue is also guaranteed to be stable. The advantage of this approach is that buildup of the shadow queues can take place to provide a routing "gradient" for the back-pressure algorithm without corresponding build up (and so packet delay) of the real queues, but at the cost of reduced network capacity.

Min-resource routing is also proposed in this paper, where the utility function is modified as

$$Wab(t) = \max(Q_a(t) - Q_b(t) - M)\,\mu_{ab}(t) \qquad (12)$$

Where $M \geq 0$, is a design parameter. With this change, the differential in queue backlog at node n and neighbor m must exceed M packets before packets for flow f will be forwarded to m, since the utility function must be positive before a link will be used. M = 0 recovers the basic back-pressure algorithm. Larger values of M make forwarding decisions less sensitive to small changes in queue occupancy, but this comes at the cost of large queue sizes.

D. LIFO Backpressure

In this paper [10] Backpressure algorithm is combined with Last in First out (LIFO) queuing discipline. LIFO performs like original Backpressure but only the queuing discipline is changed from FIFO to LIFO. It shows that this algorithm achieves utility within O(1/V) of the optimal value, for V≥1, where Vis a scalar, by maintaining an average delay of $O([\log(V)]^2)$for few fraction of the network traffic. In order to improve delay for majority of traffic, some of the packets has to wait in the queue for long time. Hence this algorithm is further developed by interleaving between FIFO and LIFO.

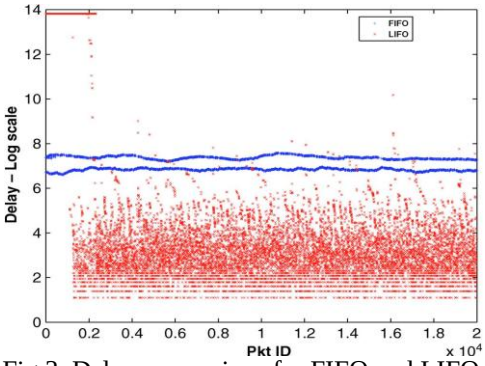

Fig 3. Delay comparison for FIFO and LIFO

Figure 3 [10] shows the calculated delay for the first 20 000 packets that enter into the network when V=500. It shows that Backpressure algorithm along with LIFO queuing discipline, most packets experience very small delay, while under Backpressure with FIFO, each packet under goes an average amount of delay.

E. Adaptive Redundancy

Backpressure works well under high traffic conditions, by utilizing the available network resources in a highly dynamic fashion. However in low traffic conditions, when packets don't enter the network, many other nodes may have a small or 0 queue size. This leads to inefficiency, by increasing delay. This is because when there are no packets in the queue, it is difficult to build up gradients and so the packets may enter looping or take long time to reach the destination. Such delays cannot be tolerated in already delay tolerant network. In this case redundant transmission can be used to improve delay when the traffic is low. In [11] adaptive redundancy technique is included along with backpressure where data packets are replicated when the traffic becomes low.

In order to preserve the performance of backpressure, as well as make use of replication to improve the throughput under low load condition, adaptive redundancy technique can be used. This technique creates copies of packets in a new duplicate buffer upon an encounter, when the transmitter's queue occupancy is low. These duplicate packets are transmitted only when the original queue is empty. Since copies of same packet exist in multiple nodes, the destination will have more chance to encounter the message intended for it. By this way the packet reaches the destination faster reducing looping. It also describes the methods of maintaining, utilization and removal of data packets from the network.

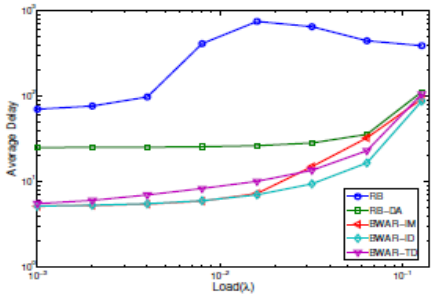

Fig.3. Comparison of delay performance of backpressure algorithms.

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 2, February 2014

Figure3[11] is the comparison of backpressure algorithms from the regular backpressure algorithm, along with(BWAR-IM), where when duplicated original remains in the main queue and duplicates are stored in duplicate buffer. For (BWAR-ID) both original and duplicate are stored in duplicate buffer. In (BWAR-TD) duplicates are removed from duplicate buffer after a predefined timeout.

IV. CONCLUSION AND FUTURE WORK

DTNs are emerging communication paradigm, where people find various uses by such a network from telemedicine to exotic networks and remote areas. This paper is a comprehensive study DTNs and its algorithms. At the beginning of this paper overview of DTN is given. Then different algorithms that have been used to improve the performance of DTN is studied. Next back pressure based scheduling algorithm is studied from its origin to various enhancements undergone by it, such as changing the queuing discipline from FIFO to LIFO to reduce delay, shadow queue techniques to reduce the overhead of nodes maintaining queues, greedy maximal algorithm to properly utilize the available capacity, adaptive redundant algorithm in order to increase the throughput in case of low traffic has been widely studied.

Backpressure works well in high traffic, the nodes of the backpressure suffers due to overload, since each node has to maintain queues for each destination. This can be reduced by using shadow queue techniques, but at the penalty of limited network capacity. In this case adaptive redundancy can be merged with shadow queue technique to improve throughput as well as delay.

REFERENCES

[1] Stephen Farrell, Vinny Cahill, Dermot Geraghty, Ivor Humphreys & Paul McDonald 2006, 'When TCP Breaks Delay and Disruption-Tolerant Networking', Internet Computing, IEEE, vol. 10, no. 4 ,pp. 72-78.

[2] Alex McMahon & Stephen Farrell,2009, 'Delay- and Disruption-Tolerant Networking', Internet Computing, IEEE, vol. 13, no. 6, pp. 82-87.

[3] Kaimin Wei, Xiao Liang &KeXu, 2013, 'A Survey of Social-Aware Routing Protocols in Delay Tolerant Networks: Applications, Taxonomy and Design-Related Issues', IEEE Communications Surveys & Tutorials, pp1-23.

[4] Takahiro Matsuda & Tetsuya Takine, 2008, '(p,q) - Epidemic Routing for Sparsely Populated Mobile Ad Hoc Networks', IEEE Journal On Selected Areas In Communications, vol.26, no.5, pp.783-793.

[5] Pan Hui, Jon Crowcroft&EikoYoneki, 2011, 'BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks', IEEE Transactions On Mobile Computing, vol. 10, no. 11, pp. 1576-1589.

[6] Erik Kuiper &SiminNadjm-Tehrani, 2011, 'Geographical Routing With Location Service in Intermittently Connected MANETs, IEEE Transactions On Vehicular Technology, vol. 60, no. 2, pp. 592-604.

[7] L.Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," IEEE Transactions on Automatic Control, pp.1936–1948, December 1992.

[8] C. Joo, X. Lin, and N. B. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks," in Proc. IEEE INFOCOM, 2008.

[9] L. Bui, R. Srikant, and Alexander Stolyar, "Novel Architectures and Algorithms for Delay Reduction in Back-pressure Scheduling and Routing", in Proc IEEE Infocom Mini-Conference 2009.

[10] Longbo Huang, Scott Moeller, Michael J. Neely &BhaskarKrishnamachari, 2013 'LIFO-Backpressure Achieves Near-Optimal Utility-Delay Tradeoff', IEEE/ACM Transactions On Networking, vol. 21, no. 3, pp.831-884.

[11] MajedAlersaini, Maheswaran sathiamoorthy, Bhaskar Krishnamachari, Michael J.Neely, Backpressure with Adaptive redundancy(BWAR),IEEE INFOCOM 2012.