Performance Analysis of Buffer Management Policies in Delay Tolerant Network

Pradeep Kaush Ajay Prasad

Computer Science Engineering, Sikkim Manipal University Sikkimal

Abstract— **Delay Tolerant Network is a subset of Mobile Adhoc Networks, which deal with wireless communication and data delivery with intermittent connection or partitioned networks. Nodes in this network discover the relay nodes for very short intervals as soon as they encounter each other. As it works on intermittent connection, end-to-end path establishment is not possible, it works on store carry and forward paradigm. If nodes are not able to forward messages due to the absence of connectivity, messages got buffered based on different queue policies. The scheduling and drop policies comprise the buffer management policy. Scheduling determines which messages are forwarded first, while drop policy determines which messages get dropped in the occurrence of a buffer overflow. This paper provides an overview of several buffer management strategies utilized in DTN and assesses their benefits and drawbacks. Buffer management is divided according to the type of information used, whether it is local knowledge about messages available at the node or global knowledge about messages in DTN network.**

Index Terms: **Buffer management, Drop policy, Priority, Sorting.**

INTRODUCTION

As the population is increasing worldwide, the use of mobile gadgets like smartphones, laptops, wearables, or any device which required a connection between two devices also growing. While cities frequently have a solid mobile web system or infrastructure, there are nevertheless some locations with poor network coverage or no connectivity at all. Especially in disaster-hit areas where the network goes completely down, but communication becomes very vital at that point of time for relief works. Relief workers need to communicate with each other or to the relief center for any medical assistance or relief material distribution or sharing location of any casualties etc. But due to the no connection or intermittent networks communication setup is not possible, like traditional Mobile Adhoc Network (MANET) or TCP/IP, which require complete path setup before initiation of communication.

Delay tolerant networks can be defined as situations with intermittent or no connectivity, long average delay, uneven data rates, and high error rates [1-3].One of the important features of DTN is to store carry and forward messages because the chances of a destination coming into the encountering range of a relay or source node are very low in a DTN. It means if node wants to transmit a packet to relay node but is not able to do so, due to the absence of path or range, then the message will get forwarded to the node that gets encountered, it then stores and carry the message to the buffer until the next node or destination node encountered.

As DTN works on store carry and forward policy, it arises two main problems that turn out to be two different research areas in the DTN. The first issue is that it is not feasible to forward messages to encountered nodes. The fundamental reason is for forwarding a message to numerous relay nodes places a strain on the node's energy and buffer, as well as wastes limited network bandwidth. It can lead to network congestion due to the flooding of messages in a network. Hence, which node to select for forwarding must be chosen very carefully. One of the most important problems which open several research areas is the management of space in the buffer. Because nodes in DTN use a store carry and forward technique, their buffer fills up very fast. It is possible that every node in the network gets exhausted from a message generated from one node. In such a case, whenever two nodes come in encounter range, the node must either discard a message from buffer based on their respective drop policy or reject the source node for a new message transfer.

Managing buffer space becomes very crucial in DTN because it particularly includes message storage and message replicas, which consume a large amount of bandwidth and space on nodes [4].Whenever the space of buffer gets exhausted, to accommodate a incoming message nodes has to drop a few messages which can be important message and impact the delivery of a correct message to the destination. Buffer management in DTN is defined not just by dropping policies, but also by scheduling policies. Because DTN has to suffer from partial and intermittent connection, sorting messages in order of their relevancy are critical for successful delivery. It is possible that while transferring message connection may be lost or power shut down due to limited energy, hence it is important to choose the correct drop and scheduling policy.

So, the goal should be to provide efficient buffer management which can increase the delivery probability or ratio of the network. Despite the fact that DTN is built to permit considerable delays. So, decreasing average delay becomes the most important issue to be considered while developing an effective buffer management policy. It should not be the case that upon arrival of a message to a destination with a considerable delay, its significance is gone. This paper surveys and reviews several policies published for DTN buffer management. These policies have been categorized mostly based on whether they are using local or global available information about all messages. Various algorithms are briefly explained, along with their merits and demerits in an intermittent network. The rest of the paper is organized as mentioned below: Section 2 explains different buffer management policies, section 3 explains various sorting techniques, section 4 metrics used to measure performance, section 5 includes simulation and result and section 6 concludes the paper.

II.BUFFER MANAGEMENT POLICIES

In this section, we will study various buffer management schemes for DTNs.The techniques are discussed in the order in which they were proposed in prior years, along with that we will also discuss their benefits and drawbacks.

A. Drop Policies

Davis et al. [5] proposed four policies for dropping. These are designed to improve performance in heavily partitioned or intermittent networks. The necessity of these principles was emphasized in this research considering the growing use of wearable computers where packet transport methods are used. As mentioned, this paper introduced four drop policies, namely, Drop Random (DRA), Drop Least Recently Received (DLR), Drop Oldest (DOA), and Drop Least Encountered (DLE). As the name suggests, DRA simply drops a random message when buffer space is full and requests for a new message, this process will go on until space for a new message is created and accommodated. DLR discards messages that have been sitting in a buffer for a longer period of time. The reason behind that is that DLR will assume that message stays for a long time and might be encountered with many nodes and must have been got some relay node or destination node. It is assumed that these packets have enough replicas in a network and have higher chances to get delivered to the destination compared to other packets.DOA drop packets that float in the network for a longer period. Because DOA demands network information, it is more adaptive. Messages are floating in the network for an extended amount of time are more likely to have been delivered and hence more likely to get dropped.

However, in this paper major light is given on DLE policy. Here, messages get dropped on the basis of estimated delivery probability. It is more adaptive than the other three techniques since it considers node location and movement during implementation. To forward a packet, DLE sort the packet based on the relative ability of nodes. For the calculation of relative ability, every node maintains a list that contains node addresses of other nodes in the DTN. For each timestamp, Node A must update the meeting time for the next node C in proximity to co-located node B. When node A encountered node B, the packets in the buffer are ordered depending on their relative ability to carry the packet to the destination. A technique like this assures that packets can always be delivered from nodes with a low likelihood of delivering messages to the destination nodes with a high chance of delivering a message to the destination. As a result of this, we may conclude that DLE works better than other techniques due to its adaptability to DTN settings.

B. MaxProp Buffer Management Policy

MaxProp is a protocol that is introduced very early with DTN, it is the first protocol that includes buffer management with routing. There are some protocols that are introduced in an earlier day of DTN, like Epidemic, Spray and Wait, PROPHET [7-9], all these three protocols do not specify or consider buffer management problems. To manage buffer in MaxProp, depending on the number of hops of packets, the buffer is logically divided into two halves. If the hop count is below threshold value t, the packets are then organized in ascending order of hop count. The packets for which hop counts are greater than threshold t, packets are arranged in descending order of delivery likelihood. The delivery likelihood is a new measure in MaxProp that prioritizes packets depending on the expense of reaching the target.

Whenever a node encounters another node, those packets having a hop countless compared to the threshold value are given priority and forwarded first. Upon transmission of all such packets, packets with a hop count more than threshold t are granted the next opportunity. The MaxProp prioritizes packets with a low hop count by giving preference to its first buffer segment, which means packets with a low hop count are new to the network and still haven't traveled far enough or close to their destination. As a result, the MaxProp buffer management method prioritizes such packets. To promote better packet delivery, the MaxProp employs an efficient scheduling mechanism for forwarding packets to relay nodes. When dropping packets, they are always dropped from the buffer's tail. Because tail contains less delivery likelihood packets. When compared to the other basic routing protocols in DTN, MaxProp frequently has a high delivery ratio, emphasizing the significance of buffer management. MaxProp's overhead, on the other hand, is high since the quickest path to the target is calculated to determine the 'delivery likelihood.'

C. Queuing and Scheduling Policies

Lindgren and Phanse offered several policies for dropping and scheduling [10]. As in the PROPHET routing protocol, delivery probability concepts were used in this policy as well. The authors demonstrated that taking delivery predictability into account while making routing and dropping decisions outperformed basic random selection, as utilized in the epidemic routing protocol. Authors here proposed some of the drop policies: First-in-First-out (FIFO), Most Forwarded First (MOFO), Most Favorably Forwarded First (MOPR), Shortest Life First (SHLI), and Least Probable First (LPF) are some of the drop policies

The most fundamental of the dropping policies is FIFO. The messages in the queue are arranged to discard earlier received messages. The first message in the queue gets removed, then the second, and so on. MOFO scheme forwards a message that gets relayed most. The node increments a counter every time a message is forwarded. In the event of buffer overflow, the recently sent message is dropped, This is because such packets have already reached a large number of nodes, and have a high probability of being delivered, and maybe quickly dropped.

SHLI policy work based on message time-to-live (TTL). The message with the shortest time to live (TTL) is the first to be dropped. This assumes a message with a low residual TTL has very little time to reach its destination node and will expire shortly, enabling it to be easily dropped.

MOFO has the best delivery ratio, whereas SHLI has the least average delay, according to the authors. This is due to MOFO's requirement that each packet is relayed once before being dropped. As in the case of SHLI, a message can also be dropped if the message stays in a buffer for a longer time and has not been forwarded.

D. Priority Queue Mechanism for Dropping Messages

Ayub et al. suggested a priority queue-based technique for differentiating source, relay, and destination communications[11]. Following that, each priority queue received its own drop metric. They have also developed a technique known as Time to Dead (TTD) for assigning a time value to messages sent to their intended recipients. As a result, the program must use TTD-supplied messages. Messages, on the other hand, will get dropped from the destination node's buffer once the TTD has expired. They also created a method for tracking messages that are in buffer but have not yet been delivered. The purpose is to get rid of these messages because the node can't locate an appropriate carrier for them. They have provided a method for calculating the number of message copies sent by the current node. The goal is to eliminate messages that have a high number of transmissions.

Rashid et al. offer Multi-Queue Buffer Management based on Dynamic Prediction (DPMQ) [12]. The approach works by separating messages in the local buffer in three different queues at the node: a low-priority queue known as the Low-Prediction Traffic list (LPTL), a high-prediction queue known as the High-Prediction Traffic list (HPTL), and a Destination Connected message traffic list (DCTL). Messages with predictability less than the threshold are queued in the LPHL queue. In case of buffer overflows, the priority-based drop event gets activated, and the initially accessible messages in the lower-priority LPTL queue are dropped. If the LPTL is empty, the High Prediction Traffic drop event is triggered; otherwise, the relay message is deleted.

III.SORTING TECHNIQUES USED IN QUEUE

The buffer management policy is a combination of queue policy and drop policy. Queue provides the sequence of a message staying in a buffer, and in the drop policy, messages are discarded if there is congestion in the order in which they are sorted in a queue. The sorting order is typically decided exclusively by local parameters. The FIFO (first-in-first-out) sorting system is well-known for sorting messages in ascending order based on their arrival times. While the majority of sorting orders have shorter names, we differentiate them by the sorting parameter and the sorting direction, FIFO used for the arrival time of a message on a node in ascending order, LIFO stands for Arrival-time in descending order, and MOFO stands for Replications in descending order [3].

IV.METRICS

The following metrics are used to assess the effectiveness of various buffer management policies. Because all metrics are scalar, they are all defined by an average single value after measuring or simulating numerous times.

1) Delivery Ratio: It refers to the ratio of messages generated from source and received to destination. A high delivery ratio suggests that more messages are delivered effectively.
2) Overhead Ratio: It shows, the number of different messages generated in order to deliver the source message to the destination.
3) Average Hop Count: Measured to count the number of nodes used to reach a message from its source to destination.

V.SIMULATION AND RESULTS

For evaluation of any routing protocol or buffer management policies, we need to use such an environment, where controlled network setups can be recreated. Despite the fact that simulation environments manage to work like real-time environments, there are different factors that cannot be controlled properly. Here, for buffer management analysis we are using Opportunistic Network Environment (ONE) simulator version 1.6.1. We compare our scheme to seven current drop policies created in the ONE simulator: SHLI, MOFO, LPR, DOA, LIFO, DPMQ, and DLA. All these techniques are mentioned in the literature.

A. Simulation Settings

For the proper validation of results, the simulation needs to run many times by considering different factors. After simulating different times, with different factors, the average of all simulations, average performance is measured in order to find out delivery ratio, average overhead ratio, and average hop count to reach a single message to the destination node. We are considering two groups or two types of node, detailed setting mentioned in below table:

Table1: Simulation Parameters

Simulation Time		13200 seconds				
Interface		Bluetooth				
Transmission Range		10m				
Transmission Speed		250 Kbps				
Buffer Size		5 Mb, 10Mb, 15Mb, 20Mb, 25Mb				
Node Speed	Pedestrians	(0.5 – 1.5) m/s				
	Vehicle	30-40km/h				
Mobility Model		**Random Way Point**				
No. of Nodes	Carrier Node	34	54	67	77	87
	Internal Node	91	96	108	123	138
	Total Nodes	125	150	175	200	225
Bundle Size		500kB - 1MB				
Bundle TTL		40m				

B. Results and Discussion

To check the impact of buffer space on drop policies, we consider various buffer sizes, starting from 5MB to 25MB by 225 nodes with two groups namely, pedestrian and vehicle. Simulation shows a direct impact on different buffer management policies by varying buffer sizes. DTN works on store carry and forward approach, most of the time it transmits by multi-copy routing. By doing so, they mostly transmit more than one copy, so that it can move forward using the different paths and reach to destination quickly. One of the major disadvantages with this is that even though the message reached the destination other copy of messages remains flooding in the network, which creates congestion and nodes become full. Hence no place for new incoming messages and generates higher overhead.

Figure 1 shows the delivery ratio of seven schemes mentioned in the literature, with different buffer sizes. By simulation, we found that when the buffer size is as low as 5MB, the delivery ratio in the DPMQ scheme is 52% greater than the other six policies. As the buffer size increase from 5 MB to 25 MB, all policies delivery ratio tends to increase, because of more packets flooded in a network, but in DPMQ in the increase of buffer size delivery predictability also gets increased. Hence, DPMQ is better than all other six policies because itforwards and drops a lesser message, as a result, it saves bandwidth and more messages can accommodate on given bandwidth.

We have also analyzed the overhead ratio to estimate the resources consumption and to find the average message relayed in order to deliver one message. Figure 2 shows the overhead ratio of all seven buffer management techniques with varying buffer sizes. Out of all policies, LPR, MOFO and LIFO generate higher overhead. Because, It drops messages that have spent less time in the buffer and generally drops source messages. In finding overhead ratio, DPMQ again perform well as compared to other policies.

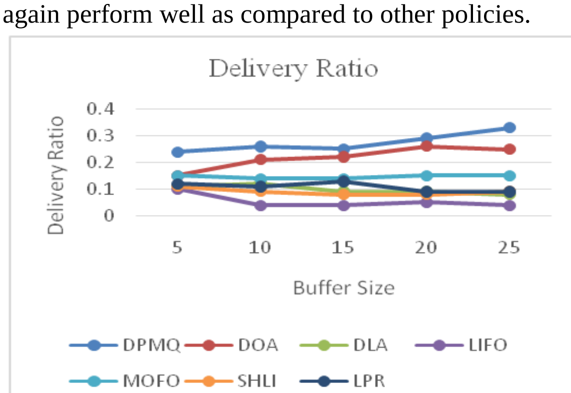

Figure 1: Delivery Ratio by varying buffer sizes

One of the most important metrics in DTN, is to measure the average number of hops per message takes to reach the destination. Lesser hop count is necessary because DTN works on infrastructure-lessnetworks, where resources always remain in scarcity, so the aim is to travel a message less number of nodes to reach the destination. Figure 3 shows the average number of hops used by different policies to reach the destination. On average nodes used by DOA policy is 3, DPMQ 1.5, DLA 2.8, LIFO 2, MOFO 2.8,SHLI 2.4, and LPR uses 2.5. Thiscontrast arises from the use of dynamic prediction multiple queues saved messages from a redundant drop, that has a low overhead on routing protocols, which tends to avoid messages from being largely generated message copies on the PROPHET.

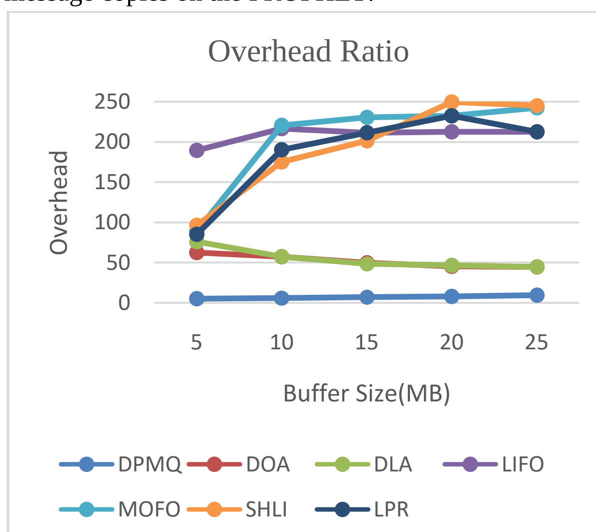

Figure 2: Overhead Ratio by varying buffer sizes

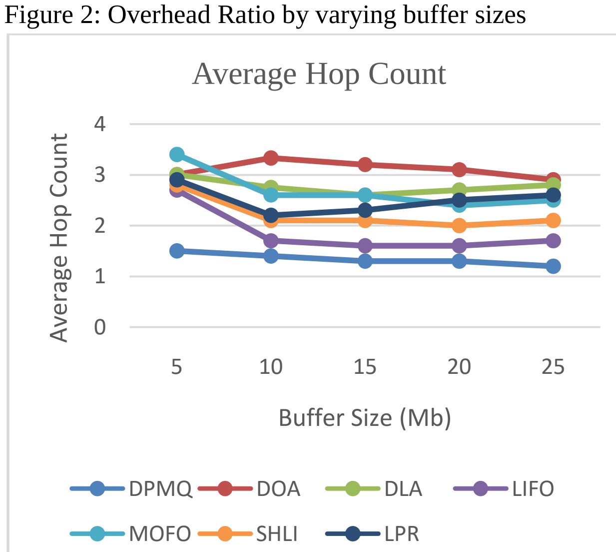

Figure 3: Average hop count by varying buffer sizes

VI.CONCLUSION

DTN will be more useful in the next-generation internet topology. Buffer management and routing are two significant study topics in DTN. We investigated seven buffer management policies in this study: DPMQ, DOA, DLA, LIFO, MOFO, SHLI, and LPR. In our investigation, we found that DPMQ performs well in terms of delivery ratio, overhead ratio, and average hop count compared to all other six policies. All these policies basically work ona number of replicas of packets and the sorting a message in a queue according to their utility, so that drop event can occur based on the order messages are stored in buffer. In this study, we tried to show various buffer policies that are generally used with protocol and specific for some protocols. In the future, we can use soft computing or machine learning to make decisions for buffer management.

REFERENCES

[1] Ayub Q, Ngadi MA, Rashid S, Habib HA (2019) Priority Queue Based Reactive Buffer Management Policy for Delay Tolerant Network under City Based Environments. PLOS ONE 14(10): e0224826.

[2] Burgess J., Gallagher B., Jensen D. (2006), MaxProp: routing for vehiclebased disruption-tolerant networks. INFOCOM– The 25th IEEE Int. Conf. Computer Communications, Barcelona, Catalunya, Spain, 23–29, pp. 1–11.

[3] Cao Y., Sun Z. (2012), Routing in delay/disruption tolerant networks: a taxonomy, survey and challenges, IEEE Commun. Tutor.

[4] Davis J.A., Fagg A.H., Levine B.N. (2001) ‘Wearable computers as packet transport mechanisms in highly partitioned ad-hoc networks’. Int. Symp. Wearable Computing, , pp. 141–148

[5] Delay-Tolerant Networking Research Group. Retrieved from http://www. dtnrg.org

[6] Delay Tolerant Network: Challenges and Application by Farid Farahmand (AITISl)

[7] Krifa A., Barakat C., Spyropoulos T.(2008) Optimal buffer management policies for delay tolerant networks. IEEE Conf. Sensor, Mesh and Ad Hoc Communications and Networks (SECON), pp. 1–9

[8] Lindgren A., Doria A., Schelen O.(2003) Probabilistic routing in intermittently connected networks, SIGMOBILE Mob. Comput. Commun. Rev, 7, (3), pp. 19–20

[9] Lindgren A., Phanse K.S. (2006) Evaluation of queuing policies and forwarding strategies for routing in intermittently connected networks. Proc. IEEE COMSWARE, pp. 1–10

[10] Ayub Q. Hanan A.Rashid S. Zahid M. (2013) Dynamic Prediction based Multi Queue (DPMQ) drop policy for probabilistic routing protocols of delay tolerant network. Elsevier, Journal of Network and Computer Applications.

[11] Spyropoulos T., Psounis K., Raghavendra C.S. (2005) Spray and wait: an efficient routing scheme for intermittently connected mobile networks. ACM SIGCOMM 2005 – Workshop on Delay-Tolerant Networking and Related Networks (WDTN-05), Philadelphia, PA, USA, 22–26, pp. 252–259

[12] Vahdat A., Becker D.: (2000) Epidemic routing for partially connected ad hoc networks. Technical Report, CS-200006, Duke University.