Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039697

ARTICLE

NPBMT: A Novel and Proficient Buffer Management Technique for Internet of Vehicle-Based DTNs

Sikandar Khan[1], Khalid Saeed[1], Muhammad Faran Majeed[2,*], Salman A. AlQahtani[3], Khursheed Aurangzeb[3] and Muhammad Shahid Anwar[4,*]

[1]Department of Computer Science, Shaheed Benazir Bhutto University Sheringal, Dir (U), Khyber Pakhtunkhwa, 18200, Pakistan

[2]Department of Computer Science, Kohsar University Murree, Punjab, 47150, Pakistan

[3]Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, P.O. Box 51178, Riyadh, 11543, Saudi Arabia

[4]Department of AI and Software, Gachon University, Seongnam-si, 13120, South Korea

*Corresponding Authors: Muhammad Faran Majeed. Email: m.faran.majeed@kum.edu.pk; Muhammad Shahid Anwar. Email: shahidanwar786@gachon.ac.kr

Received: 11 February 2023 Accepted: 05 June 2023 Published: 31 October 2023

ABSTRACT

Delay Tolerant Networks (DTNs) have the major problem of message delay in the network due to a lack of end-to-end connectivity between the nodes, especially when the nodes are mobile. The nodes in DTNs have limited buffer storage for storing delayed messages. This instantaneous sharing of data creates a low buffer/shortage problem. Consequently, buffer congestion would occur and there would be no more space available in the buffer for the upcoming messages. To address this problem a buffer management policy is proposed named "A Novel and Proficient Buffer Management Technique (NPBMT) for the Internet of Vehicle-Based DTNs". NPBMT combines appropriate-size messages with the lowest Time-to-Live (TTL) and then drops a combination of the appropriate messages to accommodate the newly arrived messages. To evaluate the performance of the proposed technique comparison is done with Drop Oldest (DOL), Size Aware Drop (SAD), and Drop Larges (DLA). The proposed technique is implemented in the Opportunistic Network Environment (ONE) simulator. The shortest path map-based movement model has been used as the movement path model for the nodes with the epidemic routing protocol. From the simulation results, a significant change has been observed in the delivery probability as the proposed policy delivered 380 messages, DOL delivered 186 messages, SAD delivered 190 messages, and DLA delivered only 95 messages. A significant decrease has been observed in the overhead ratio, as the SAD overhead ratio is 324.37, DLA overhead ratio is 266.74, and DOL and NPBMT overhead ratios are 141.89 and 52.85, respectively, which reveals a significant reduction of overhead ratio in NPBMT as compared to existing policies. The network latency average of DOL is 7785.5, DLA is 5898.42, and SAD is 5789.43 whereas the NPBMT latency average is 3909.4. This reveals that the proposed policy keeps the messages for a short time in the network, which reduces the overhead ratio.

KEYWORDS

Delay tolerant networks; buffer management; message drop policy; ONE simulator; NPBMT

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

DTNs means “network of regional networks”. DTNs lack instantaneous and continuous end-to-end connectivity [1]. DTNs operate where a very long delay occurs between nodes. It works where the network loses connectivity periodically and results in loss of data. DTNs work under a predefined mechanism called the “store and forward rule” [2]. Each message/data is stored in a node and forwarded upon successful connection [3,4]. The sender node will keep a copy of the sent message until and unless it receives an acknowledgment from the receiver [5]. DTNs can also be called overlay on top of the traditional regional networks, including the Internet, the extranets, or the intranets. Other traditional network models possess problems of power, limited memory, linkages, etc., DTNs tried to resolve the problem due to specific and special features [6]. DTN technologies may include Ultra-Wide Band (UWB), free-space optical, and acoustic (sonar or ultrasonic) technologies [7,8]. Some research work has been done on the network delay and stalling of packets, but their work is limited to the quality of experience of a virtual environment and virtual reality video delivery in a congested network without any buffer management technique [9]. A proper and workable buffer management technique is the core requirement of DTN, such the network may be able to decide its onward step for the accommodation of newly arrived messages up on the buffer full situation. We have tried to find a reasonable solution for the buffer management in DTNs and to introduce a mechanism for the resolution of the ‘buffer full’ problem, due to which the newly arrived messages do not get space in the buffer. Arrival and acceptance of the newly arrived messages are badly affected due to no free space in the buffer of the receiver to accommodate them.

1.1 Applications of DTNs

Some applications of DTNs are as follows.

1.1.1 Communication Source in Rural Areas

It is much more difficult to provide the facility of the Internet at every point in rural areas. DTNs play a very important role in such scenarios and can be used as a mode of communication in remote villages to provide access to the Internet. This will ensure a reduction in cost and connectivity of users with the network [10].

1.1.2 Wild Zebra Networks

In 2004, researchers thoroughly observed the various activities of zebras in a wild, scattered area. For this reason, an idea called the “zebra collar” was implemented in which zebras were equipped with communication devices. Those devices were using the Global Positioning System (GPS). The mechanism was that the collars start for a few minutes to record the GPS location and every two hours initialize the radio functionality for communication between two zebras, whenever they come across contact with each other. Through this mechanism, the activities of zebras such as grazing, drinking, and even their rest time can easily be monitored [11].

1.1.3 Military and General Purpose

Due to the spreading of war zones, it is required that there may be a proper mechanism to monitor, control, and check the activities of army personnel, track the army vehicles, track the army’s cargo, track, and monitor the undersea activities of marines, do rescue control, etc. [12]. All these activities require a proper and effective communication mechanism. Due to the sensitive and far distances between the end users, it is not possible to do the communication in a traditional way [13]. DTNs play an important role and they can be used for military and intelligence purposes for search and rescue, cargo tracking, and vehicle control [14]. Moreover, the subject network can be used for agricultural activities in far and spread areas to monitor crop growth, see various stages of crop growth, vehicle tracking, underground mine communication, and data transactions [15,16].

1.1.4 Commercial Purpose

DTNs can be utilized for commercial purposes such as tracking vehicles, monitoring crops, data transactions, and underground mine communication [17].

1.1.5 Personal Use

DTNs can be utilized for monitoring personnel in remote areas and wildlife [17].

1.1.6 Agricultural Use

DTNs can be used for gathering data from remote sensors, different agricultural fields, environmental factors, etc. [18].

1.1.7 Application in Embedded Systems

DTNs play an important role in the advancement of embedded systems. Through DTN technology, a wireless local area network access point can be turned into a stand-alone DTN node for mobile applications [19].

1.2 Buffer Management in DTNs

Buffer management is an important and mandatory component of networks. DTNs also possess many specific methods and protocols for buffer management [19]. DTNs work on the store and forward rule as reflected in Fig. 1. The messages are forwarded from Node A to Node B, Node B forwards to Node C and then Node C forwards to Node D. Each node uses its storage capacity for storing messages. The storage capacity of each node and the forwarding of messages by nodes is reflected in Fig. 1. Whenever a message is forwarded from a node, the receiving node stores it in its buffer, until and unless it gets a chance to be forwarded further. Upon buffer overflow, a systematic procedure is required to drop any of the stored messages to make space for the newly arriving messages [19].

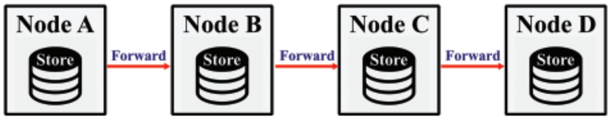

Figure 1: Movement of messages in DTNs [5]

Bundle layer Protocol over the traditional internet layers is overlapped by the DTNs architecture. The bundle layer operates lonely to send/receive data from node to node by utilizing custody transfer. Whenever a node sends a bundle in DTNs, it initiates a request. If the receiver accepts the bundle and gets its custody, it re-sends an acknowledgment packet to the sender. But if the 'time-to acknowledge' expires and the receiver does send any acknowledgment to the sender, it re-transmits the packet. Bundles in the bundle layer operate independently and none of the bundles wait for the other bundle delivery, response [7]. The layer distribution of DTNs is shown in Fig. 2.

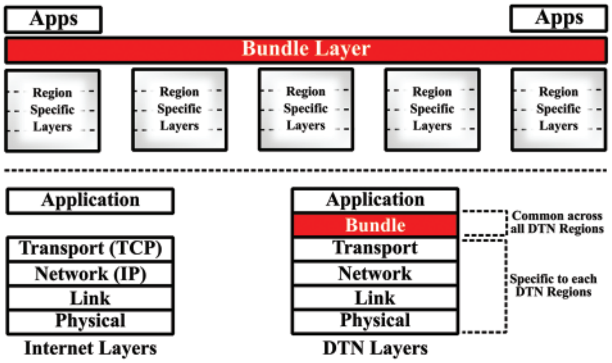

Figure 2: Bundle layer of DTNs over TCP-IP [8]

1.3 Buffer Management Policies in DTNs

When it comes to Delay Tolerant Networks (DTNs), buffer overflow is a major issue that results in message loss [19]. To tackle this problem, various buffer management policies have been proposed. It is essential to have an efficient policy in place for the DTN environment to function properly [20]. These policies can be divided into two categories such as policies having no network knowledge and policies having network knowledge.

1.3.1 Policies Having No Network Knowledge

Some buffer management policies in DTNs do not keep the knowledge of the network. These policies are useful because they work independently. These policies do not require the implementation of a specific routing scheme. Some examples of such policies are as follows:

Drop Front (DF)

The message drop policy that can drop messages from the memory front is called Drop Front. The Drop Front policy acts in a way that as the buffer overflow occurs, it starts dropping the messages from the front of the memory. This can be referenced as the First in First out (FIFO) mechanism [21].

Drop Last (DL)

Drop Last policy drops or discards messages from the end of the buffer. The freshly arrived message is removed first from the last end of the list. It may be stated that the message with the least arrival time will be deleted first from the memory [21].

Drop Random (DR)

As the name shows, this buffer management policy of DTNs is deleting random messages from the overflowed buffer. No deletion priority parameter has been set in it, and the mechanism has no such knowledge of the messages that which one must be deleted. It just selects random messages for deletion. This random deletion policy ensures that for every message, i.e., a freshly arrived message or an old one, small size or large or moderate size, all messages have equal chances of deletion. The system can select anyone randomly for deletion [22].

Drop Youngest (DY)

In DTNs, each message has a lifetime known as Time-to-Live (TTL). When a message travels and spends time in the network, its TTL reduces. This message drop policy deletes messages having the highest TTL value. As cleared, a new and freshly arrived message has consumed very few resources of the network, and dropping such a message which is the youngest among all messages may create the minimum overhead ratio. As the deletion of freshly arriving messages is occurring, there is a possibility of losing many important and informative messages [23].

Drop Oldest (DOL)

When a message travels and spends time in the network, its TTL reduces. The message having the lowest TTL value is the oldest. This policy drops the message having the lowest TTL value. It does not consider the freshly arrived messages for dropping or deletion [21].

Drop Largest (DLA)

It is understood that every message traveling in the network will have a specific size, some will be a small size, some will be a large size, and some will be a moderate size. Based on and considering the size of a message for its dropping, in [24], Rashid et al. introduced a buffer management scheme called DLA. DLA considers the size of a message as a parameter to select it for deletion. This scheme drops messages having the largest size from the buffer to accommodate the newly arriving message [25].

1.3.2 Policies Having Network Knowledge

Some buffer management policies keep some information regarding the network such as the number of nodes, message copies, etc. Such buffer management policies utilize available information while deciding on a message drop.

Evict Most Forwarded First (MOFO)

DTN buffer management can keep a message sender counter for the sent messages. Through how many hops or nodes a message has been passed, it keeps this information. MOFO a buffer management scheme, considers or measures this parameter and considers it as an input property for the deletion of a message from an overflowed buffer. It will set the message-sending counter value, and the message with the maximum value will be dropped. The message with the highest counter or highest hop value also guarantees that there may exist many replicas of the message in the network. Hence, a message that has already created many replicas in the network may not be required anymore. It will be better to get rid of such redundant messages. Deletion of such replicated messages will create room for the newly arrived messages to travel freely in the network and propagate themselves [26].

Global Knowledge-Based Drop (GBD)

In [27], Krifa et al. presented a buffer management technique that considers some statistical information to originate the value of each message based on node contact history in the network. This is quite valuable to maximize the average value of the delivery ratio and minimize the average value of the delivery delay. Obviously, along with its usefulness, the knowledge of the mobility model used in the network is mandatory for it. It also requires an infinite transfer bandwidth and all messages in the network may be the same size. Such harsh and impractical requirements of GBD make it difficult for it to survive in real-world networks. All these features are much more difficult to acquire, and these cannot be reached realistically [27].

Evict Least Probable First (LEPR)

Evict Least Probable First (LEPR) calculates an extra variable to a message called its FP value. LEPR measures the lowest FP value of a message. It is considering messages for dropping having the lowest FP values. The main drawback of LEPR is that those messages that were not yet passed from several hops and chosen to be deleted first. This is an anonymous loss of useful information and loss of such messages that have not yet created their replicas in the network [28–30].

1.4 Problem Statement

Due to the non-availability of all-time end-to-end connectivity, DTNs store messages in their available or provided memory/buffer, but nodes have the limitation of storing messages due to non-availability or shortage of resources (i.e., more memory to accommodate the newly arrived messages). As a result, when messages are received and the node does not have any more space to accommodate the received messages, congestion occurs. To accommodate the received messages, the node must drop some existing messages from the buffer. SAD and E-Drop policies drop messages according to their sizes. These policies drop equal or nearly equal size messages from the buffer and do not have the mechanism for selecting small-size messages to be dropped from the buffer in case of having all small messages from the buffer. To solve this limitation in the mentioned drop policies, a new buffer management policy called a Novel and Proficient Buffer Management Technique (NPBMT) is proposed in this research, and to properly evaluate the performance of NPBMT, a comparison is done with SAD, DOL, and DLA policies.

1.5 Objectives of the Research

The main objectives of this research are as follows:

- Select the appropriate smallest size message with the lowest TTL.
- To decrease the probability of unnecessary message dropping by appropriate size message selection.
- To increase the forwarding probability of large-size messages.
- To evaluate the performance of the proposed policy by comparing it with existing policies using ONE simulator.

1.6 Significance of the Research

Due to the limited buffer size, accommodating new messages in DTNs is a challenging task, when the buffer is full. A proper, effective, and efficient policy for buffer management is needed in the DTN environment. The message drop policy presented in this research has the advantage that initially it will consider the equal size message to be dropped. In case of the non-availability of an equal-size message, the proposed system will consider the appropriate size message with the lowest TTL for dropping, which will cause an overall improvement in performance.

The rest of the paper is arranged as follows: Section 2 contains a literature review, Section 3 contains the methodology adopted for this research, Section 4 includes results and detailed discussion, and Section 5 concludes the research along with some future research direction.

2 Review of Literature

DTNs are established upon requirements and have totally an ad-hoc type of network, having no such permanent deployment model or phenomenon. Due to this non-regular and mobile nature, DTN architecture has low bandwidth, buffer space, and energy issues. As two nodes are in the range of each other, they communicate successfully with each other. As a result, at once many messages interchange is expected, which will certainly require a large enough bandwidth. Provision of such a huge bandwidth is impossible, and buffer overflow must be observed. Hence, certain messages will be dropped to free up space for the newly arriving messages.

In [30], Rashid et al. explained in their research the importance of buffer management and message-dropping policies in DTNs. Replication and storage of messages result in congestion. A significant and useful policy for dropping the messages in case of congestion from the buffer was the intense need for time. Their work declared how a dropping message can be important. They have shown the importance of the dropped messages.

In [31], Elwhishi et al. practiced a delay-tolerant network with a uniform delay mechanism. They have included epidemic routing protocol and two-hop forwarding routing mechanism for uniform delay tolerant networks. They worked hard to evaluate the message delivery ratio and tried to find a way to control message delivery delay. They have implemented a mechanism named GHP, which showed much better results than the existing buffer management policies. Their simulations included two mobility models, a synthetic random waypoint, and a real trace model zebra net.

In [32], Rani et al. presented a policy that includes the performance comparison of different buffer management policies like MOFO, Epidemic, PRoPHET, and MaxProp. The policies were assembled with different message buffer sizes. They have calculated the delivery probability, overhead ratio, and latency time average of MOFO with the MaxProp routing protocol. Their research work gave better results for the three routing protocols (i.e., Epidemic, Prophet, and MAxProp).

Silva et al. [29] worked on congestion control in DTNs. Their research showed that a large quantity of information in the shape of messages is wasted due to congestion. Not only is message dropping a waste of information, but congestion also plays a very vital role in the waste of valuable information. Whenever congestion occurs, it wastes valuable resources. They have proposed a comprehensive congestion control mechanism that combines the two mechanisms (Proactive and Reactive).

In [33], Iranmanesh introduced the idea of dropping messages in the shape of bundles. He suggested a mechanism that used encounter-based quota protocols for forwarding the messages. According to his phenomenon, bundles of messages are made during the forwarding process and priority is assigned to each bundle. As the nodes contact each other, the bundles start to transmit based on their priority assigned. The proposed idea got a score because of its multi-objective characteristics. It reduced delivery delays as well as enhanced the delivery ratio.

In [34], Moetesum et al. presented a mechanism called Size Aware Drop (SAD), calculating exactly the required space of the newly arrived message, and selecting a message of the same size to drop as per the required space. SAD shows better results over Epidemic (Drop Oldest and Drop Largest) and PRoPHET (Evict Shortest Lifetime First and Evict Most Forwarded First).

In [35], Ayub et al. introduced a policy that purely worked based on Priority-based Reactive Buffer Management (PQB-R). This buffer management policy divided the buffer messages into three subcategories. These categories are named source, relay, and destination queues. Another mechanism of a separate drop metric is also deployed to each separate queue for dropping messages. After various tests, the results have shown that PQB-R gave much better results than the policies in comparison with, and has increased the number of messages transmission, delivery ratio, and reduced messages drop.

In [36], Rashid et al. presented a new buffer management policy called "Best Message-Size Selection buffer management policy" (BMSS) [36]. This policy utilizes the message header's local information, such as message size, for controlling the message drop mechanism. They evaluated the effectiveness of their proposed policy against existing policies DOA, DLA, LIFO, MOFO, N-Drop, and SHLI using ONE simulator. Their designed mechanism worked with an adaptive approach having the feature of selecting equal or large-size messages for dropping. Their proposed policy, BMSS, has three main procedures. Best Message-Size Selection, Drop Large Size Message and Drop Equal Size Message. The mechanism used its built-in provided intelligence to use which one policy for message drop. As the message arrives and congestion occurs, its size is checked and upon its size, the system will decide which procedure to call. If the arrived message size is less than the free buffer space then it will be accommodated, else if, drop equal size message drop function will be called to check the equal size message for dropping, on the zero replies of the equal size message drop function drop the largest message size function will be called and that will drop the largest size message for accommodating the arrived message. Their simulation results proved that BMSS has reduced message drop, overhead ratio, end-end delay, latency and resulting in an increase in the delivery ratio.

In [26], Samyal et al. presented a Proposed Buffer Management Policy (PBMP). The PBMP focused on the number of forwards NoF and TTL. It records the number of forwards of a message as well as its TTL. If a message resides in the buffer, it may increase the probability of high forwarding as well and its TTL may become lower. PBMP observed these two features and decided to drop a message on these two attributes. This mechanism has been checked in comparison with another message-dropping policy 'MOFO'. MOFO drops messages having the highest value of the sent counter. While PBMP combines two attributes of a message, NoF, and TTL, and then decides to drop a message. The proposed policy PBMP works with a 13.18% higher delivery probability than MOFO.

In [37], Abbasi et al. have produced and diagnosed some statistical calculations and numerical analyses about buffer management in DTNs. They showed that obvious buffer management is a serious challenge faced by DTN researchers. They have presented an idea of determining the relay nodes using fuzzy logic in DTNs. They have proved that if the sender node determines the destination node it will send data, otherwise it will wait for meaningful contact. Their work showed that by using fuzzy logic the determination of relay nodes can increase the efficiency and delivery probability of the messages. Their work elaborates that by increasing the number of encounters, the packet delivery ratio will increase too. If any sender node has more information about the destination node, this information will enhance the message delivery probability. It has been observed that as a node has higher energy, its delivery probability ratio will be higher as compared to low-energy nodes.

In [38], authors devised a mechanism for the load balancing and fair delivery of messages in DTNs. As obvious that DTNs mostly adopt the mechanism of SPRAY and WAIT protocol through which messages are sent anonymously to all nodes. There is not any such load balancing mechanism to check the sent and received messages. A few responsible relay nodes that are efficient in forwarding and receiving messages observed/accommodated all the load of data delivery and fairness has been much harder to maintain in such networks. Such efficient and workable relay nodes may face the problem of power failure due to a very high rate of usage. They may drain quickly because of their overuse. Moreover, their chance of being congested is much greater due to heavy load. Their research specifically proposed the solution for such reasons. The research was conducted in a disaster-affected area. They have identified and analyzed the role of important parameters for delivering messages fairly in DTNs. They have presented a metric called Fairness Aware Message Forwarding (FAMF) metric for message forwarding in the network. In their presented work, whenever the buffer of a relay node becomes full, the message having the lowest contact with the destination will be dropped from the buffer. Results have shown that FAMF has a better message delivery probability. Their presented scheme maintained fairness in message delivery in case of congestion in the relay nodes.

In [19], Rehman et al. proposed an inception size-based message drop policy SS-Drop for DTNs. SS Drop selects an appropriate message to be dropped when the buffer becomes full. The main advantage of the suggested scheme is that it avoids unnecessary deleting of messages from the buffer. The implementation of SS-Drop has been done using ONE simulator. The comparison of SS-Drop has been done with a drop front, drop oldest, and drop largest. SS-Drop produced better results as compared to drop oldest, drop front, and drop largest in terms of different performance evaluation parameters.

Artificial Intelligence can be utilized in DTNs to improve their efficiency. AI can also be utilized for the implementation of efficient secure solutions in DTNs. The authors in [39] have employed a combination of AI and Machine Learning techniques for gathering multimedia streaming data. They developed a technique called IAIVS-WMSN for Video Surveillance in IoT-enabled WMSN, which integrates AI. Their main objective was to create a practical method for object detection and data transmission in WMSN, and the proposed IAIVS-WMSN approach comprises three stages: object detection, image compression, and clustering. Object detection in the target region primarily relies on the Mask Regional Convolutional Neural Network (RCNN) technique. Additionally, they use the Neighborhood Correlation Sequence-based Image Compression (NCSIC) technique to minimize data transmission.

The authors in [40] developed an event-driven architecture (EDA) to enable the efficient implementation of IoT in various environments such as mobile, edge, fog, or cloud. The communication between the different components of the EDA was established through event brokers that facilitate communication-based on messages or events. Moreover, the components created were environment-independent, allowing them to be deployed in any environment. The authors introduced the SCIFI-II system in their paper, which is an event mesh that enables event distribution among event brokers. Using this system, components can be designed independently of event brokers, simplifying their deployment across various environments.

The authors of [41] aimed to tackle the challenge of managing the massive amount of multimedia data generated by Internet of Things (IoT) devices, which cannot be handled efficiently by cloud computing alone. To address this, they proposed using fog computing, which operates in a distributed environment and can provide an intelligent solution. The authors specifically focused on minimizing latency in e-healthcare through fog computing and proposed a novel scheme called Intelligent Multimedia Data Segregation (IMDS). This scheme employs machine learning techniques (k-fold random forest) to segregate the multimedia data, and a model was used to calculate total latency (transmission, computation, and network). The simulated results showed that the proposed scheme achieved 92% classification accuracy, reduced latency by approximately 95% compared to the pre-existing model, and improved the quality of services in e-healthcare.

3 Research Methodology

This section includes the methodology used to carry out this research. It includes the flow chart and detailed explanation of NPBMT, simulation parameters that are utilized in carrying out the simulation, and performance evaluation parameters for evaluating the performance of NPBMT.

Recent studies show that whenever a new message arrives and its size is greater than the available free space in the receiving node, some messages must drop to make room for the newly arrived messages. To overcome these problems different mechanisms have been made and tried. Some researchers agreed to drop messages that traveled for a long time in the network, while some were dropping messages of big size.

3.1 Flow Chart of NPBMT

The Flow chart of the proposed scheme NPBMT is reflected in Fig. 3.

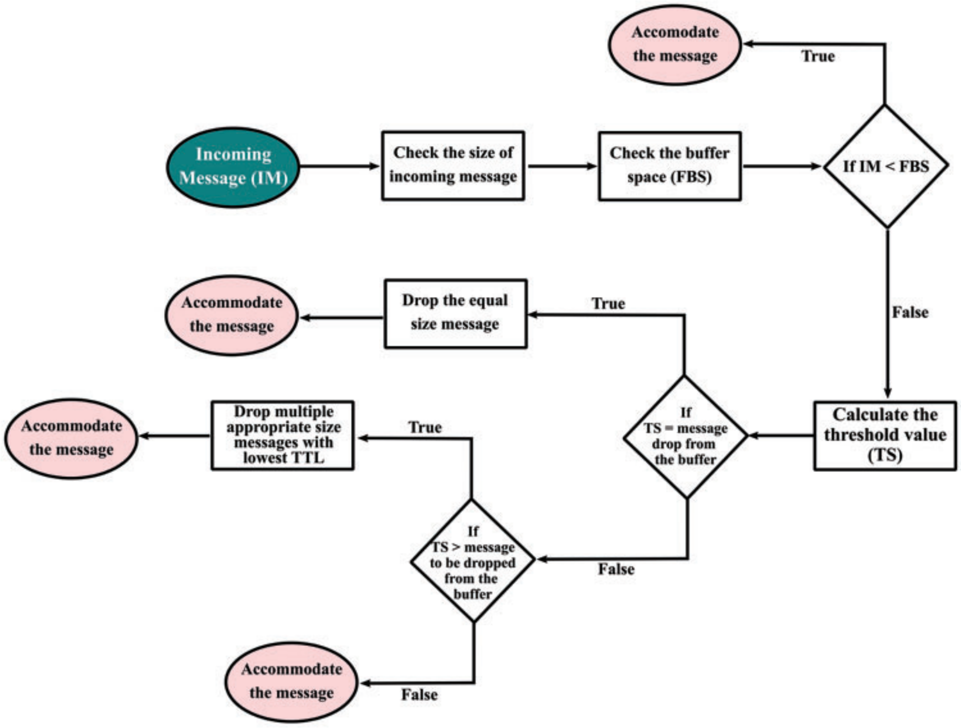
Accomodate the message
True
Incoming Message (IM)
Check the size of incoming message
Check the buffer space (FBS)
If IM < FBS
False
Accommodate the message
Drop the equal size message
True
Accommodate the message
Drop multiple appropriate size messages with lowest TTL
True
If TS = message drop from the buffer
Calculate the threshold value (TS)
If TS > message to be dropped from the buffer
False
Accommodate the message
False

Figure 3: Flow chart of NPBMT

An Incoming Message (IM) will enter the buffer and the proposed system will observe the size of the IM as well as Free Buffer Space (FBS). If the size of IM is less than that of FBS, then the entered message will be accommodated directly in the buffer. If the proposed system finds that FBS is less than the size of the IM, then the actual space for the IM will be calculated, and it is known as Threshold Size (TS). The formula for calculating the threshold size is given in Eq. (1).

$$TS = IM - FBS \tag{1}$$

All the enlisted messages will be compared with the TS value and a message with an equal size of the Ts value will be dropped. In case the proposed system does not find any equal size match with the Ts value, then the proposed policy will select the appropriate size messages with the lowest TTL to accommodate the newly arrived message.

3.2 Simulation Parameters

The simulation parameters used to carry out the simulation using ONE simulator are reflected in Table 1. The simulation parameters used in this research are common parameters and these have been utilized by the relevant research carried out in this area.

Table 1: Simulation parameters

Parameter	Value		
	Set 1	Set 2	Set 3
Simulator	ONE		
No. of nodes	20	40	60
Movement model	Shortest path map-based movement model		
Routing protocols	Epidemic		
Buffer size	3 MB	6 MB	9 MB
Group interface	Bluetooth		
Transmission speed	300 kbps		
Transmission range	15 m		
Nodes speed	1–3 m/s		
TTL	3600 s		
Message creation interval	15–25 s		
Message size	500 KB–1 MB		
Simulation area	4500 × 3400		
Simulation time	43200 s		

3.3 Performance Evaluation Parameters

The following performance evaluation parameters are utilized for evaluating NPBMT performance.

3.3.1 Messages Drop (MD)

The difference between the messages created and delivered is known as messages drop. In DTNs, the number of messages dropped should be minimized. The formula to calculate the messages dropped is given in Eq. (2).

$$MD = Messages\ Received - Messages\ Delivered \tag{2}$$

3.3.2 Delivery Probability (DP)

The ratio of messages transferred successfully to the receiver is called delivery probability (DP). The delivery probability should be maximized to improve performance. The formula for the delivery probability is given in Eq. (3).

$$DP = Messages\ Delivered - Messages\ Created \tag{3}$$

3.3.3 Overhead Ratio (OR)

The ratio between the messages relayed and delivered messages. The overhead ratio in the network should be minimized. The formula for the overhead ratio is given in Eq. (4).

$$OR = \frac{Messages\ Relayed - Messages\ Delivered}{Messages\ Delivered} \tag{4}$$

3.3.4 Buffer Time Average (BTA)

Messages always reside in the nodes. Over there it stays and takes some time. The time spent by a message in the buffer can be called the buffer time average. The buffer time average should be maximized to keep the messages for a long time in the buffer, which increases the delivery probability of the messages. The buffer time average can be calculated as per Eq. (5).

$$BTA = \sum\nolimits_{i=1}^{n} Message(i) \frac{(Stay\ Time)}{Message\ in\ Buffer} \tag{5}$$

3.3.5 Latency Time Average (LTA)

The latency time average is the delay of messages in the network. LTA should be minimized to ensure the timely delivery of messages to its target [21].

3.3.6 Messages Relayed (MR)

Messages relayed are the ratio of all those messages sent to the other nodes and still on the way while reaching their destination. To create more duplicates of the traveling message in the network, the messages relayed ratio may be maximized [21].

3.3.7 Hop Count Average (HCA)

The message passed from several hops to reach the destination is known as the hop count average. The low hop count average value indicates that the message has traveled through fewer hops. The high value of the hop count average declares the maximum number of hops through which the message has been passed while reaching its destination [34].

3.4 Pseudo Code of NPBMT

The pseudo-code for NPBMT is shown in Algorithm 1.

Algorithm 1: NPBMT (X, New Message)/Here X is the free space where a new incoming message will be accommodated

```
if X == Size of the new message then
    X = insert (new message)
else
    Create Space (X, New message)
end if
Make room/Create space (X, New message)
if (X != New message) then
    Loc = Find equal size message for dropping and creating space
    Loc == New Message
else
```

(Continued)

Algorithm 1 (continued)

```
11: Loc = Drop Multiple appropriate size messages with the lowest TTL
12: Loc == New Message
13: end if
```

4 Results and Discussion

The proposed message drop policy NPBMT is tested and implemented through ONE simulator. It has been tested in comparison with Size Aware Drop (SAD), Drop Oldest (DOL), and Drop Largest (DLA) buffer management policies. DOL policy is doing the deletion process in a manner that drops/deletes the message with the lowest TTL, regardless of its size. This policy does not care about the size of a message, and it is a clear indication that the drop oldest policy may drop such an utmost message required by the users. Also, there is a possibility that even though the message was left with the least TTL, it still did not get a route or chance to be propagated in the network and there it has no replicas in the network. This deletion act performed by the DOL will result in serious information loss consequently. As obvious that DLA is a message-dropping scheme that drops only the message by considering its size. This policy drops the message as it counts its size to be the largest of all. It does not measure how much a message is old in a buffer, whether it has generated its replicas or not, or maybe this message did not yet get a chance to travel to its destination. DLA will directly hit the biggest size message in the buffer, regardless of its importance. Such deletion will create a scenario of very important information loss. Much required and extensively needed message/information may be deleted due to its large size. Also, a fresh message that just newly arrived in the buffer may be the victim of the Drop Largest policy due to its largest size among all. At the same time, if any message, i.e., important, or not important replicated, or not replicated can be deleted by the DLA or DOL Schemes.

SAD can drop an equal size or nearly equal size of messages from the memory of the node to store the new incoming message. In case all the available messages in the buffer are of a smaller size than the size of the incoming message, then SAD combines small messages and makes a bundle for deletion. Furthermore, it calculates a threshold size (Ts) which is the difference between the size of the message and available free buffer space. This Ts is the exact free buffer space required for the accommodation of the incoming message. The system proceeds onwards to random deletion of the existing messages, keeping this Ts as a parameter. The limitation of SAD is that it does not look for the appropriate size message to be deleted, nor does it have any priority-assigning mechanism for the deletion of messages. It is deleting randomly amongst all equal-sized messages.

The proposed policy in this research has the mechanism to select the appropriate size messages, considering firstly those messages that have the oldest TTL, along with their sizes in consideration for deletion. The reason for this is that messages with the appropriate or equal size with the oldest TTL may have many copies of it in the network. Also, they may have very little time remaining in the network, and they must vanish/die soon. It is very suitable to select such messages for deletion that have spent a very long time in the network and are near dead.

4.1 NPBMT Scenario

The performance of NPBMT is evaluated using ONE simulator. The scenario is created for NPBMT. The parameters used in the scenario are 20, 40, and 60 nodes having a buffer memory of 3, 6, and 9 MB, respectively. Among these nodes, 20 are nominated as the sender, 40 are declared as a receiver and the remaining 60 nodes are put in the category of stationary or relayed nodes. The epidemic routing protocol is utilized as a routing protocol for the scenario because of its flooding-based nature. Shortest Path Map Based has been chosen as the movement model for this message drop policy, as it contains the property of random movement based on Point of Interest (PoI). The shortest path map-based model provides the opportunity that the nodes to move randomly and easily toward the targeted point. Bluetooth served the purpose of media in the tested policy.

The range of transmission is 15 m. The message creation interval has been kept small (i.e., 15–25 s) due to the reason that creates a greater number of messages in the network. The TTL of a message in the network is set to 60 min and the scenario run time is 43200 s. The simulation area for this research work has been kept as 4500 × 3400.

4.2 Delivery Probability

Delivery probability is the term used to show the number of successfully transferred messages to their destinations. The main purpose of this research is to increase the number of delivered messages through the network in such a manner that delivery probability may increase. A comparative analysis of the delivery probability of the proposed scheme NPBMT with DOL, SAD, and DLA is given in Fig. 4.

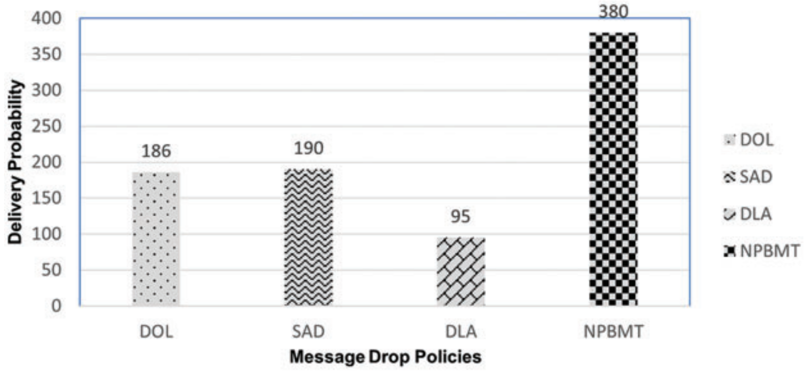

Figure 4: Delivery probability of DOL, SAD, DLA, & NPBMT

It has been proved that the delivery probability of the tested schemes goes in such a way that NPBMT got the most delivery probability value, i.e., 380, secondly SAD produced a value of 190, thirdly DOL showed its delivery probability value of 186 and DLA has proved to be the worst case amongst, which is only 95. The Delivery probability of the NPBMT is better than that of DOL, SAD, and DLA because the proposed message drop policy focuses on dropping the multiple appropriate size messages with the lowest TTL. The selection of multiple appropriate-size messages is for minimizing the number of messages to be dropped. The reason behind considering the lowest TTL is that the message has already spent much time in the network and may possess numerous replicas in the network, which can be a certain cause of the increase in the delivery probability.

4.3 Overhead Ratio

The ratio between messages relayed and messages delivered in a network can be called an overhead ratio. A comparative analysis of the overhead ratio of this research-proposed scheme NPBMT with DOL, SAD, and DLA is reflected in Fig. 5.

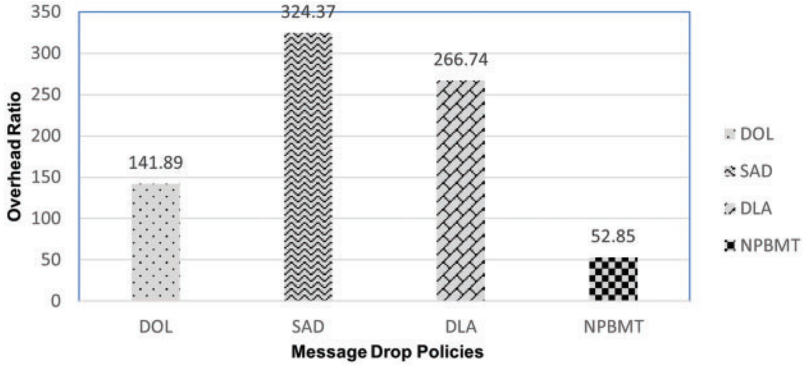

Figure 5: Overhead ratio of DOL, SAD, DLA, & NPBMT

As shown in Fig. 5, the overhead ratio of the proposed schemes is much lower, i.e., only 52.85 as compared to the other accompanying tested policies. The others DOL, SAD, and DLA in the competition got the value of overhead ratios 141.89, 324.37, and 266.74, respectively. The overhead ratio of the proposed scheme NPBMT is much lower as compared to the others. The reason is that the proposed scheme intelligently selects the message to be dropped from the buffer to accommodate the incoming message and efficiently utilize the available resources. NPBMT got a tremendous increase in the delivery probability and produced many effective results in terms of the delivery of messages. An increase in the delivery of messages would certainly reduce the overhead ratio and produce better results.

4.4 Hop-Count Average

To successfully reach its destination, a message must travel along the network and must pass from different nodes/hops. Hop-count average is the number that shows how many hops/nodes a message has passed while reaching and traveling to its destination. As it shows the number of hops through which the message has been passed during its traveling time in the network, so hop-count average must be low for a better buffer management policy. A comparative analysis of the Hop-count average of this research-proposed scheme NPBMT with DOL, SAD, and DLA is shown in Fig. 6.

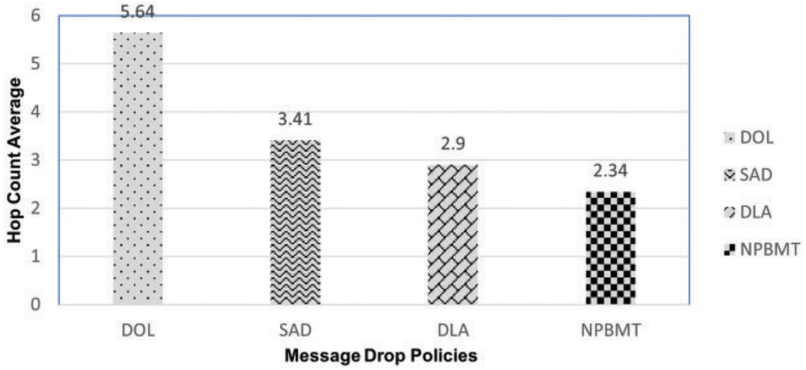

Figure 6: Hop count average of DOL, SAD, DLA, & NPBMT

It is clear from Fig. 6 that the Hop count of DOL is the largest among all, i.e., 5.64. Onward, SAD got second place and produced a 3.41 value in terms of hop count average. DLA produced a 2.9

hop count average while the proposed mechanism NPBMT produced the lowest value 2.34 in terms of the hop count average, which is the main purpose of this research. The hop count average means the number of hops through which a message has been passed, it shall be as low as possible. A low or least hop count average guarantees that the message-dropping policy is efficient. As here in this scenario, NPBMT showed the lowest result, so it means that NPBMT is the best amongst its competitors in terms of hop count average. This showed the better intention of the proposed scheme towards best performance.

As a result, the messages must pass from a small number or few numbers of hops, so the consumption of the network resources in terms of memory, power, etc. will be much lower as compared with the other schemes DOL, SAD, and DLA. This enhances the overall performance of the network as well as the delivery probability.

4.5 Buffer Time Average

While traveling to its destination, a message must spend time in the buffer. This time spent by a message in the buffer is called the buffer time average. A higher buffer time average indicates a good forwarding probability of a message in the network. A comparative analysis of the buffer time average of this research-proposed scheme NPBMT with DOL, SAD, and DLA is given in Fig. 7.

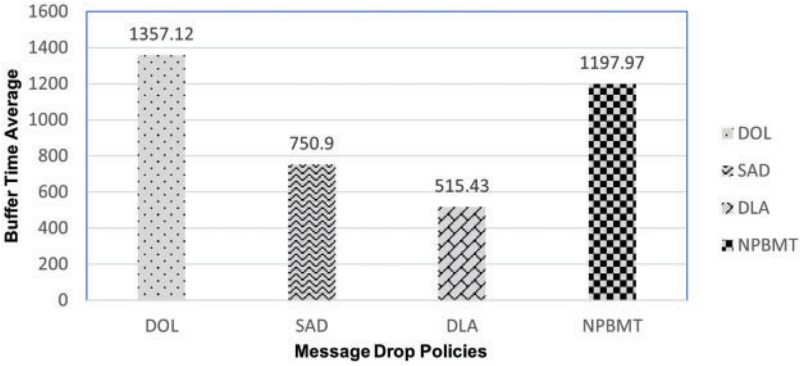

Figure 7: Buffer time average of DOL, SAD, DLA, & NPBMT

As buffer time average is the traveling time of a message in the buffer. The proposed scheme got an in-between position in terms of buffer time average amongst the other competitor schemes, i.e., DOL, SAD, and DLA. NPBMT's buffer time average is 1197.97 lower than that of DOL 1357.12. It is quite reasonable compared to DOL. While the SAD buffer time average is 750.9 and the DLA buffer time average is much better, i.e., 515.43. Spending more time in the network is better in a way that, as the message spends more time in the network, there is a chance that it may generate multiple replicas in the network due to which its delivery probability may increase.

4.6 Messages Dropped

There are several purposes of the already presented buffer management policies for DTNs which are to reduce the ratio of message dropping. An increase in the number of message drops means that overall performance is affected. Also, an increase in message drop shows and indicates the wastage of valuable resources. An effective message-drop policy must have a low message-drop ratio as compared to the existing ones. A comparative analysis of the message drop of this research-proposed scheme NPBMT with DOL, SAD, and DLA is reflected in Fig. 8.

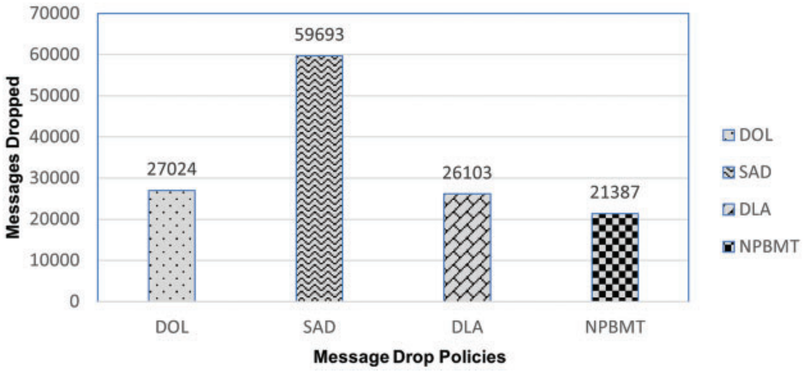

Figure 8: Messages dropped ratio of DOL, SAD, DLA, & NPBMT

Fig. 8 is self-explanatory and shows the performance of the buffer management schemes, i.e., DOL, SAD, DLA, and NPBMT in terms of messages dropped. It is observed that SAD dropped the maximum number of messages in the tested scenario and produced a very high dropped messages value i.e., 59693. Secondly, the highest messages-dropping policy is DOL which dropped 27024 messages. DLA stands in the third position, and it dropped 26103 messages. NPBMT proved to be the most proficient and workable buffer management policy which only dropped 21387 messages in the tested scenario. Its message-dropping ratio is much lower as compared to the other buffer management policies. These results declare the proposed message drop policy NPBMT a better buffer management policy among the four tested. NPBMT certainly reduced the number of dropped messages, which created a very positive and efficient performance impact on the network. The reason is that the system focuses on the deletion of very appropriately sized messages with the lowest TTL to accommodate the newly arrived message. These features of the proposed scheme make it much different from the existing schemes in terms of messages dropped. The proposed scheme is tested in comparison with DOL, SAD, and DLA which are a few most popular buffer management schemes. Significant improved results have been obtained as shown in Fig. 8. Accordingly, SAD dropped the most messages, DOL stood second in terms of dropping messages, DLA proved to be third in the most dropping messages scheme and NPBMT proved to be the least dropping messages scheme.

4.7 Latency Average

As data travels from hop to hop in the network, it takes time to travel. Sometimes the network has very low resources in terms of memory, power, etc., due to which the processing of the data takes a long time in the network. This delay in the processing of data in the network can be termed as Latency average. Greater latency averages guarantee that a message will stay for a long time in the network and will have more chances of delivery. Comparison analysis in terms of the Latency average of the proposed scheme NPBMT with DOL, SAD, and DLA is reflected in Fig. 9. Latency is the traveling time taken by a message in the network. Low latency means that a message passes from a smaller number of hops in the network and, therefore, consumes much fewer resources. In Fig. 9, DOL achieved the highest latency, i.e., 7785.5 among all. DLA remained second achieving a 5898.42 markup, SAD reached a 5789.43 value and NPBMT achieved a 3909.4 value. As much as a message will take less time to reach its destination, it will reduce the burden on the network, and it will certainly enhance the performance of the network. It will also save energy and memory consumption.

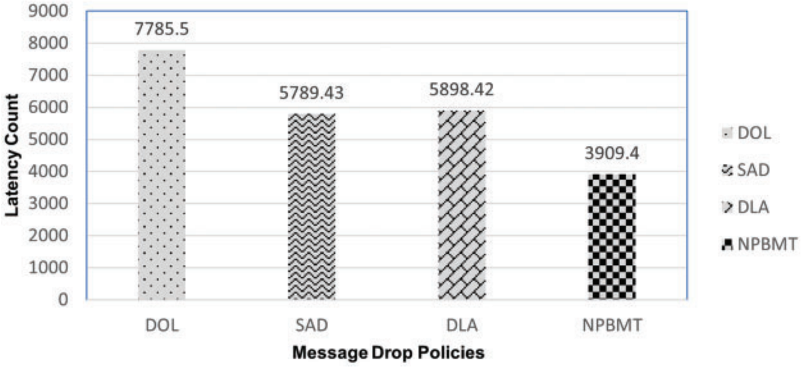

Figure 9: Latency count average of DOL, SAD, DLA, & NPBMT

4.8 Messages Relayed

Relay nodes are the stationary nodes in DTNs that hold messages that are sent to other nodes while reaching their destination or target nodes. Relay nodes store such messages for future communication and contact with other nodes. Relay nodes can also be termed stationary nodes. Comparison analysis of the relayed messages of the proposed scheme NPBMT with DOL, SAD, and DLA are reflected in Fig. 10.

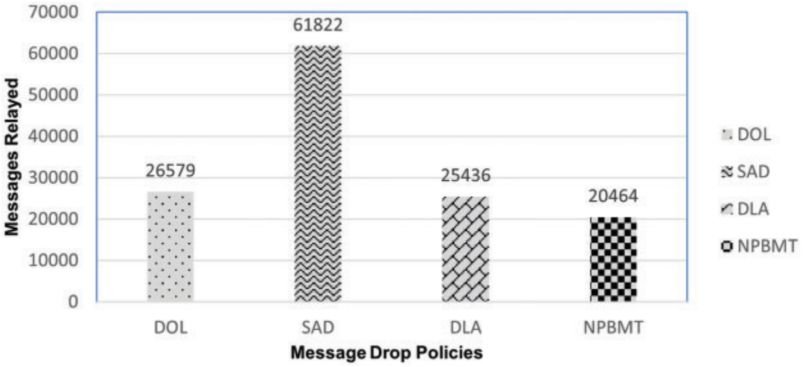

Figure 10: Messages relayed in DOL, SAD, DLA, & NPBMT

The ratio of the messages relayed in the network by the proposed policy NPBMT is much better and highly efficient as compared to DOL, SAD, and DLA. The performance of a network is inversely proportional to the number of relayed messages. As much as a buffer management policy remains with the less or least number of relay messages, it will be considered better. Specifically, in the tested scenario SAD produced a very high relayed message ratio of 61822, while DOL produced 26579 relayed messages, and DLA produced 25436 relay messages, which is much more than that of NPBMT 20464. NPBMT produced a very low ratio of relayed messages as compared to DOL, SAD, and DLA. Therefore, it beats other policies in terms of messages relayed.

5 Conclusion and Future Work

A high delivery probability buffer management policy is the key requirement of DTNs. This very important problem faced by the DTNs is specifically addressed in this research. The proposed policy, i.e., NPBMT produced better results as compared to other existing policies in terms of different performance evaluation parameters. Through several tests and analyses using the Epidemic routing protocol and shortest-way map movement model, it is concluded and proved that NPBMT produced good and effective results, i.e., a significant improvement occurred in the delivery probability and can be observed, as the proposed policy delivered 380 messages, DOL delivered 186 messages, SAD delivered 190, whereas DLA delivered only 95 messages. The overhead ratio decreased significantly, as the SAD overhead ratio is 324.37, the DLA overhead ratio is 266.74, and the DOL and NPBMT overhead ratios are 141.89 and 52.85, respectively. This shows a significantly lower overhead ratio of NPBMT as compared to other existing policies. The network latency average of DOL is 7785.5, DLA is 5898.42, and SAD is 5789.43 whereas the NPBMT latency average is 3909.4. This reveals that the proposed policy keeps the messages in for a short time in the network, which reduces the overhead ratio, as compared to existing message drop policies.

In the future, we have a plan to evaluate the performance of NPBMT with other buffer management policies using the PROPHET routing protocol. Moreover, we have a plan to work on security in DTNs.

Acknowledgement: We are thankful to King Saud University, Riyadh, Saudi Arabia for the project funding.

Funding Statement: This research is funded by Researchers Supporting Project Number (RSPD2023R947), King Saud University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm their contribution to the paper as follows: Conceptualization, Sikandar Khan, Khalid Saeed, and Muhammad Faran Majeed; Methodology, Khalid Saeed, Muhammad Faran Majeed, Khursheed Aurangzeb; Software and Simulation, Sikandar Khan, Khalid Saeed, Salman A. Qahtani; Formal Analysis, Muhammad Faran Majeed and Muhammad Shahid Anwar; Investigation, Sikandar Khan, Khalid Saeed, and Muhammad Faran Majeed; Resources, Sikandar Khan, Khursheed Aurangzeb; Writing—Original Draft, Sikandar Khan, Khalid Saeed, Muhammad Faran Majeed, and Muhammad Shahid Anwar; Writing—Review and Editing, Khalid Saeed, Muhammad Faran Majeed, Khursheed Aurangzeb; Supervision, Khalid Saeed and Muhammad Faran Majeed; Funding Acquisition, Khursheed Aurangzeb.

Availability of Data and Materials: The data collected during the data collection phase will be provided upon request to the authors.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] F. D. Raverta, J. A. Fraire, P. G. Madoery, R. A. Demasi, J. M. Finochietto *et al.,* "Routing in delay-tolerant networks under uncertain contact plans," *Ad Hoc Networks*, vol. 123, pp. 102663, 2021.

[2] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proc. of the 2003 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Karlsruhe, Germany, pp. 27–34, 2003.

[3] D. McGeehan and S. K. Madria, "Catora: Congestion avoidance through transmission ordering and resource awareness in delay tolerant networks," *Wireless Networks*, vol. 26, no. 8, pp. 5919–5937, 2020.

[4] A. Sharma, N. Goyal and K. Guleria, "Performance optimization in delay tolerant networks using backtracking algorithm for fully credits distribution to contrast selfish nodes," *The Journal of Supercomputing*, vol. 77, no. 6, pp. 6036–6055, 2021.

[5] V. Goncharov, *Delay-tolerant networks*. Denmark: Institut fur Informatik, Albert Ludwigs Universitat Freiburg, Lehrstuhl fur Rechnernetze und Telematik, 2010.

[6] M. F. Majeed, S. H. Ahmed, S. Muhammad, H. Song and D. B. Rawat, "Multimedia streaming in information-centric networking: A survey and future perspectives," *Computer Networks*, vol. 125, pp. 103–121, 2017.

[7] F. Warthman and A. Warthman, "Delay-and disruption-tolerant networks (DTNS) a tutorial," Based on Technology Developed by the DTN Research Group (DTN-RG), Version 3.2, pp. 1–35, 2015. [Online]. Available: https://www.nasa.gov/sites/default/files/atoms/files/dtn_tutorial_v3.2_0.pdf

[8] M. F. Majeed, M. N. Dailey, R. Khan and A. Tunpan, "Pre-caching: A proactive scheme for caching video traffic in named data mesh networks," *Journal of Network and Computer Applications*, vol. 87, pp. 116–130, 2017.

[9] S. Khalid, A. Alam, M. Fayaz, F. Din, S. Ullah *et al.,* "Investigating the effect of network latency on users' performance in collaborative virtual environments using navigation aids," *Future Generation Computer Systems*, vol. 145, pp. 68–76, 2023.

[10] S. Passeri and F. Risso, *Application-gateway in a DTN environment*. Italy: Politecnico di Torino, 2021.

[11] A. Upadhyay and A. K. Mishra, "Routing issues & performance of different opportunistic routing protocols in delay tolerant network," *MIT International Journal of Computer Science and Information Technology*, vol. 6, no. 2, pp. 72–76, 2016.

[12] M. F. Majeed, S. H. Ahmed, S. Muhammad and M. N. Dailey, "PDF: Push-based data forwarding in vehicular NDN," in *Proc. of the 14th Annual Int. Conf. on Mobile Systems, Applications, and Services Companion*, Singapore, pp. 54, 2016.

[13] R. Abubakar, A. Aldegheishem, M. F. Majeed, A. Mehmood, H. Maryam *et al.,* "An effective mechanism to mitigate real-time DDoS attack," *IEEE Access*, vol. 8, pp. 126215–126227, 2020.

[14] M. F. Majeed, S. H. Ahmed and M. N. Dailey, "Enabling push-based critical data forwarding in vehicular named data networks," *IEEE Communications Letters*, vol. 21, no. 4, pp. 873–876, 2016.

[15] H. S. Modi and N. K. Singh, "Survey of routing in delay tolerant networks," *International Journal of Computer Applications*, vol. 158, no. 5, pp. 8–11, 2017.

[16] M. W. Kang and Y. W. Chung, "An efficient routing protocol with overload control for group mobility in delay-tolerant networking," *Electronics*, vol. 10, no. 4, pp. 521, 2021.

[17] P. Gantayat and S. Jena, "Delay tolerant network–A survey," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 7, pp. 477–480, 2015.

[18] H. Ochiai, H. Ishizuka, Y. Kawakami and H. Esaki, "A DTN-based sensor data gathering for agricultural applications," *IEEE Sensors Journal*, vol. 11, no. 11, pp. 2861–2868, 2011.

[19] O. ur Rehman, I. A. Abbasi, H. Hashem, K. Saeed, M. F. Majeed *et al.,* "SS-Drop: A novel message drop policy to enhance buffer management in delay tolerant networks," *Wireless Communications & Mobile Computing*, vol. 2021, pp. 1–12, 2021.

[20] Y. Li, M. Qian, D. Jin, L. Su and L. Zeng, "Adaptive optimal buffer management policies for realistic DTN," in *GLOBECOM-IEEE Global Telecommunications Conf.*, Honolulu, Hawaii, USA, pp. 1–5, 2009.

[21] S. Rashid, Q. Ayub, M. S. M. Zahid and A. H. Abdullah, "E-DROP: An effective drop buffer management policy for DTN routing protocols," *International Journal of Computer Applications*, vol. 13, no. 7, pp. 118–121, 2011.

[22] S. Jain and M. Chawla, "Survey of buffer management policies for delay tolerant networks," *The Journal of Engineering*, vol. 2014, no. 3, pp. 117–123, 2014.

[23] S. Rashid, Q. Ayub and A. H. Abdullah, "Reactive weight-based buffer management policy for DTN routing protocols," *Wireless Personal Communications*, vol. 80, no. 3, pp. 993–1010, 2015.

[24] S. Rashid and Q. Ayub, "Efficient buffer management policy DLA for DTN routing protocols under congestion," *International Journal of Computer and Network Security*, vol. 2, no. 9, pp. 118–121, 2010.

[25] S. Rashid, Q. Ayub, M. S. M. Zahid and A. H. Abdullah, “Impact of mobility models on DLA (drop largest) optimized DTN epidemic routing protocol,” *International Journal of Computer Applications*, vol. 18, no. 5, pp. 1–7, 2011.

[26] V. K. Samyal and N. Gupta, “Comparison of MOFO drop policy with new efficient buffer management policy,” *International Journal of Innovative Research & Studies*, vol. 8, no. 4, pp. 456–460, 2018.

[27] A. Krifa, C. Barakat and T. Spyropoulos, “Optimal buffer management policies for delay tolerant networks,” in *5th Annual IEEE Communications Society Conf., on Sensor, Mesh and Ad Hoc Communications and Networks*, San Francisco, California, USA, pp. 260–268, 2008.

[28] A. Lindgren and K. S. Phanse, “Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks,” in *1st Int. Conf. on Communication Systems Software & Middleware*, New Delhi, India, IEEE, pp. 1–10, 2006.

[29] A. P. Silva, S. Burleigh, C. M. Hirata and K. Obraczka, “A survey on congestion control for delay and disruption tolerant networks,” *Ad Hoc Networks*, vol. 25, pp. 480–494, 2015.

[30] S. Rashid, A. H. Abdullah, M. S. M. Zahid and Q. Ayub, “Mean drop an effectual buffer management policy for delay tolerant network,” *European Journal of Scientific Research*, vol. 70, no. 3, pp. 396–407, 2012.

[31] A. Elwhishi, P. H. Ho, K. Naik and B. Shihada, “A novel message scheduling framework for delay tolerant networks routing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 5, pp. 871–880, 2012.

[32] A. Rani, S. Rani and H. S. Bindra, “Performance evaluation of MOFO buffer management technique with different routing protocols in DTN under variable message buffer size,” *International Journal of Research in Engineering and Technology*, vol. 3, no. 3, pp. 82–86, 2014.

[33] S. Iranmanesh, “A novel queue management policy for delay-tolerant networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, pp. 1–23, 2016.

[34] M. Moetesum, F. Hadi, M. Imran, A. A. Minhas and A. V. Vasilakos, “An adaptive and efficient buffer management scheme for resource-constrained delay tolerant networks,” *Wireless Networks*, vol. 22, no. 7, pp. 2189–2201, 2016.

[35] Q. Ayub, M. A. Ngadi, S. Rashid and H. A. Habib, “Priority queue based reactive buffer management policy for delay tolerant network under city-based environments,” *PLoS One*, vol. 14, no. 10, pp. e0224826, 2019.

[36] S. Rashid and Q. Ayub, “Integrated sized-based buffer management policy for resource-constrained delay tolerant network,” *Wireless Personal Communications*, vol. 103, no. 2, pp. 1421–1441, 2018.

[37] A. Abbasi and N. Derakhshanfard, “Determination of relay node based on fuzzy logic in delay tolerant network,” *Wireless Personal Communications*, vol. 104, no. 3, pp. 1023–1036, 2019.

[38] A. Roy, T. Acharya and S. DasBit, “Fairness in message delivery in delay tolerant networks,” *Wireless Networks*, vol. 25, no. 4, pp. 2129–2142, 2019.

[39] F. Romany, C. Mansour, R. Soto, J. E. Soto-Díaz, D. Gutierrez *et al.,* “Design of integrated artificial intelligence techniques for video surveillance on IoT enabled wireless multimedia sensor networks,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 7, pp. 14–22, 2022.

[40] R. Berjón, M. E. Montserrat Mateos and A. F. Beato, “An event mesh for event driven IoT applications,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 7, pp. 54–59, 2022.

[41] A. Kishor, C. Chakraborty and W. Jeberson, “A novel fog computing approach for minimization of latency in healthcare using machine learning,” *International Journal of Interactional Multimedia and Artificial Intelligence*, vol. 6, no. 7, pp. 7, 2021.