International Journal of Communication Networks and Information Security
2024, 16(1), 6520
ISSN: 2073-607X,2076-0930
https://ijcnis.org/
Research Article

Enhancement of Buffer Management and Data Transmission in Delay Tolerant Network Using Secant Osprey Optimization

Ajay Prasad [1, 2*], Sandeep Gurung [3], Kalpana Sharma [4]

[1.] Research Scholar, Sikkim Manipal Institute of Technology, SMU.
[2.] Assistant Professor, Centre for Distance and Online Education(CDOE), SMU.
[3.] Associate Professor, Department of Computer Science and Engineering, Sikkim Manipal Institute of Technology, SMIT..
[4.] Professor, Department of Computer Science and Engineering, Sikkim Manipal Institute of Technology, SMIT.
***Corresponding Author:** ajay.prsd1@gmail.com

Citation: A. Prasad, S. Gurung, and K. Sharma, "Enhancement of Buffer Management and Data Transmission in Delay Tolerant Network Using Secant Osprey Optimization," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 16, no. 1, pp. 241-257, Apr. 2024.

ARTICLE INFO

Received: 22 May 2024
Accepted: 11 June 2024

ABSTRACT

In both wired and wireless networks, message forwarding takes place after establishing a route between source and destination, even in the infrastructure-less network. It stores the messages in buffer nodes and carries them until a destination or appropriate relay node is encountered. However, the storing and carrying approach for a long time causes buffer congestion. Therefore, an optimized technique is required for buffer management and efficient communication. In Delay Tolerant Network (DTN), the chances of disconnections are higher due to propagation issues, node mobility, and power disruptions. In order to solve the problem, this paper proposed a Secant Osprey Optimization (SOO)-based efficient buffer management scheme. First, the message data is collected from the Twitter dataset for buffer analysis, and then the repeated messages are removed using SHA-512. Then, messages are clustered using Lorentz Distributed K-Means (LDKM). Further, the messages are scheduled using a Probabilistic Function-based Adaptive Neuro Fuzzy Inference System (PF-ANFIS) for forwarding. For forwarding, the optimal path is selected using the SOO algorithm. Thus, the heavy load data are forwarded via the selected path without disruptions. The experimental analysis is carried out using the PYTHON software tool by comparing the proposed model with the existing methods. The simulated outcome showed that the proposed methodology attains a higher delivery ratio of 0.815 with a lower delay rate of 1217s. Also, the proposed technique schedules the messages with 98.2% accuracy, 96% precision, and 96.5% recall. Hence, the presented approaches are more highly performed in the buffer management of DTN than the existing techniques.

Keywords: Probabilistic Function-based Adaptive Neuro Fuzzy Inference System (PF-ANFIS), Buffer Management, Secant Osprey Optimization (SOO), Secure Hashing Algorithm-512 (SHA-512), Lorentz Distributed K-Means (LDKM).

INTRODUCTION

DTNs operated in extreme conditions and over very large distances are characterized as intermittently connected ad-hoc networks, where the connection between nodes is not guaranteed, and causes a long delay in the message's transmission [1], [2]. For this reason, DTNs follow a store, carry, and forward communication strategy that enables successful delivery of data to the destination. [3], thereby obtaining high data delivery and managing long delays and data losses efficiently in opportunistic or challenging circumstances and environments [4].

The routing mechanisms in the network come across a major problem called Buffer Management (BM) as the nodes in DTN store and carry messages in their local buffer and replicate numerous copies once they encounter any other node [5]. A DTN requires buffer space for holding incoming messages during periods without connectivity. If the receiving node buffer doesn't have enough buffer space in the transmission of data, it leads to considerable data loss due to partial transmission of data [6].

There are numerous works in the existing literature using different message-dropping strategies, First Input First Output (FIFO), Random, and QoI-based BM algorithms [7], [8]. Since the dropping techniques are the independent routing protocol, they may drop a message having a higher probability of being delivered to the destination than the messages in the buffer [9], [10]. Although several BM algorithms have been developed, there is no comprehensive study aware of the data size that aims to maximize buffer utilization while avoiding unnecessary message drops.ConventionalBM rules, namely drop front, drop last, and drop ran, lack an efficient way of determining which messages to drop. Alternatively, such approaches depend on the sequence, where messages arrive and reside in the buffer. Many techniques rely heavily on message prioritization. These techniques are not ideal for DTNs as they perform better in often linked networks vs frequently disconnected ones. Therefore, in this proposed work, to eliminate the buffer occurrence among the messages, the repeated messages are discarded using the SHA-512 algorithm. It checks the hash value of messages and removes the messages having the same values to eliminate repeated messages. Further, the message dropping in the buffer is prevented by clustering the messages using the LDKM method. This method can cluster both the linear and non-linear structure of the message data. So, it is utilized to reduce the message dropping and analyze the buffer occupancy. For forwarding messages without congestion, the messages are scheduled in this work using the PF-ANFIS technique. It schedules the messages based on the predefined rule and the probabilistic function, thus forwarding messages without delay. Moreover, it is essential to deliver messages along a reliable path to ensure secure transmission. Hence, the optimal path for forwarding messages is chosen in this work using the SOO algorithm. It is utilized for their hunting strategy of acquiring the food, which selects the optimal path by assuming minimum delay and maximum delivery rate as the best fitness. With regard to these presented techniques, the proposed work manages the buffer and transmits the message efficiently in DTN.

The communication network is developed for achieving real-time data transmission between the source and destination. On the contrary, the DTN manages unreliable network connectivity, especially in remote areas. The communication network easily failed to transfer data in case of poor network connectivity. Meanwhile, the DTN can store and forward the data through intermittent connectivity. Thus, data communication is improved with the help of DTN, even in challenging conditions.

BUFFER MANAGEMENT IN DTN

BM is crucial for maintaining control over buffer resources. BM is responsible for scheduling messages. Effective BM is essential for making decisions about dropping messages. When the buffer overflows, it drops messages to create a place for new ones [11].

One important difficulty with DTN is buffer overflow, which leads to frequent message drops. Several BM strategies were employed and updated over time to control the buffer overflow. These strategies fall into two categories: those without network knowledge and those with network knowledge [12].

Policies Having No Network Knowledge

Many BM strategies in DTNs don’t store information about the network. These policies can be beneficial since they function independently. These policies don't necessitate the use of a specific routing system. When storage is limited, and network circumstances are extremely changeable, these policies specify which packets to reject when the buffer is full, affecting data delivery efficiency and overall network performance. Some examples of such policies are given below:

Drop Oldest (DO): When fresh packets come, the oldest one in the buffer is discarded. This is simple, but it may result in the loss of key packets.

Drop Newest (DN): Discards the newest packet when the buffer is full, seeking to preserve earlier, potentially more vital data.

Drop Largest (DLA): Removes the largest packet to make room for other packets, which may be more useful in resource-constrained scenarios.

Drop Random (DR): This policy randomly selects messages to drop as the buffer overflows. The random deletion property provides unbiased message selection. This strategy assigns equal priority to each message for deletion.

Drop Front (DF): Messages drop from the front of the queue as the buffer overflows, this approach follows the first-in-first-out (FIFO) principle.

Drop Last (DL): The last incoming message dropped from the queue; this approach follows the Last in Last out principle.

Policies Having Network Knowledge

In DTNs, including network information in BM policies can dramatically improve performance. These rules use network status information, such as node connections, mobility patterns, and historical data, to drive packet storage and forwarding decisions. Here are a few policies that use network knowledge:

Probabilistic Drop: Historical encounter data is used to predict the chance of successful packet delivery to a destination. When the buffer fills up, packets with lower delivery probabilities are dropped first [10].

Contact History-based Drop: Prioritizes packets based on nodes' interaction history. Nodes that maintain frequent and reliable communication with the destination are more likely to keep their packets.

Energy-aware Buffer Management: Considers node energy levels while determining which packets to drop. Nodes with limited energy may prioritize discarding packets to save power for critical tasks.

Storage-aware Drop Policy: Adapts to the storage capacity of the nodes. When buffer capacity has been limited, the policy dynamically adjusts packet priority to maximize storage utilization.

Mobility Pattern-based Drop: Analyses node mobility patterns for selecting drop decisions. Packets are preserved or dropped according to the chance of future contact with the destination [13].

Community-based Drop Policy: Uses the idea of social communities in the network. Packets are prioritized according to the community framework, with packets intended for nodes in the same community being kept longer [14].

Adaptive Drop Policy: Several factors are considered, including delivery probability, level of energy, buffer status, and patterns of movement. The policy responds dynamically to current network conditions [15].

Priority-based Forwarding and Drop: Integrates priority-based forwarding and drop policies. Packets are prioritized based on network knowledge, and these priorities are taken into account when making forwarding and discarding decisions.

PROBLEM STATEMENT

The existing BM methods perform poorly due to the following shortcomings,

The data transmission in DTN regardless of data size leads to considerable data loss due to the reason of the heavy load.

The epidemic routing protocol used in the literature consumes a large amount of network resources, which includes buffer space and bandwidth.

Dropping messages randomly may lead to dropping information that has a higher probability of traveling a lot in the future in the network.

Checking the buffer occupancy was complex for large data and repeated data storage makes buffer overflow.

These limitations motivated the proposed model to have the following contributions:

To improve buffer utilization, the SOO-based path selection, which is conscious of the data size, is introduced.

To reduce the complexity of checking buffer occupancy, LDKM-based data clustering is employed.

To solve the problem of repeated data storage, SHA-512-based hashing is used for data removal.

The remaining paper is structured as follows: the related work is reviewed in Section 4, the proposed methodology is explained in Section 5, the developed model is evaluated in Section 6, and the paper is concluded in Section 7.

LITERATURE REVIEW

[16] suggested congestion control metric for BM, instead of considering only unilateral features of nodes or messages resulting in a blind spot. Congestion control metric considers nodes or message properties like the remaining lifetime of messages, delivery probability, and buffer overhead. The proposed research reduces message loss and improves network speed by analyzing the congestion control metrics of every message in the buffer and dropping the message with the lowest value. However, the calculation of joint considerations such as

delivery probability, time measurement, and buffer overhead ratio is a complex issue, and for every message repeated process was carried out.

[11] suggested a Spontaneous Size Drop (SS-Drop) method for calculating the inception size required to drop a message when a buffer overflows. The research also looked into the shortest map-based mobility paradigm. In an experimental assessment, the presented research's performance was analyzed. However, the scheduling strategy of the model wasn't an automated process, and every process scheduling was carried out.

[17] presented a Catorasystem as a multi-copy message transmission and BM solution to assist in ensuring the delivery of priority messages while minimizing congestion and degradation. In experimental analysis, the method had higher delivery ratios and shorter latencies while decreasing the consumption of resources. The complex network is made by the epidemic routing process.

[18] recommended a method based on Artificial Neural Network (ANN) to forward messages in DTN. The method used was based on the Remaining Time to Encounter Nodes with Destination Node (RTD) mechanism. As per the simulation outcomes, the algorithm makes the most appropriate decision than the top-notch. The research was highly suitable for the large number of nodes in the network because the performance was only proved with the low number of buffer nodes.

[19] presented a routing algorithm, DTN-Balance utilizes the relaying nodes' forwarding capacity and forwarding queue into account to provide an appropriate distribution of load in the network. In experimental analysis, the performance of the methodology was analyzed and it attained better results. The buffer may have a heavy load because it maintains the same data for a longer time.

[20] developed a fair credit-centric incentive system for forwarding in DTN-centric sensor networks. Thus, the incentive mechanism encourages the selfish node to cooperate with the network. The result evaluation proves that the mechanism motivates more nodes to participate in forwarding. But, the research was only proved with the specific behavior of the node, so it is not suitable for all nodes in the network.

[21] proposed a Q-learning-based adaptive multi copy message relaying in a Vehicular Delay Tolerant Network (VDTN), Q- The learning reward factor is based on the encounter rate of the node to assess the node activity and node density. Therefore, it controls the message delivery and minimizes the local copy density. In experimental analysis, the performance of the proposed research was analyzed, and it outperforms related methodology in terms of delivery rate, message latency, and network overhead across various node speeds, density, and load circumstances.

[22] suggested multi-step double Q-learning routing (DQLR) algorithm for DTN, it adopts a dynamic reward mechanism that takes routing hops and node centrality into account. By balancing these two parameters, the K-DQLR algorithm finds out best relay node. In experimental analysis, the performance of the proposed algorithm was significantly increased.

[23] developed an effective application-based DTN Position-based Opportunity Routing (POR), with increased catching to guarantee node series from source to destination enhanced cache capacity. The approach was based on a combination of three mechanisms such as application variety, network downtime, and context-aware relay selection. Simulation results show that the approach attains high performance when compared to state-of-the-art protocols. However, compromising overhead to achieve a better delivery ratio.

[12] provided an innovative and effective BM strategy for vehicle-based DTNs, combining appropriate-size messages with the shortest TTL. Therefore, increases the forwarding probability of large-size messages. However, small-size messages get dropped leading to the unnecessary dropping of a large number of small-size messages to accommodate large-size messages.

Literature Summary

From the related study, it is observed that the appropriate path for transmitting the message was not concentrated in most of the works, including [16], [17], [19], [20], and so on. But, the data congestion was attempted to reduce in [16]. Hence, the proposed study improved the data transmission by selecting the optimal path using the SOO algorithm. Due to the non-scheduling of messages, the dropping of messages occurred in [12], which affected the transmission efficiency. The proposed work resolves such issues by clustering the messages using the LDKM method and scheduling them by using the PF-ANFIS algorithm before forwarding. Thus, the delivery rate of message transmission is improved by the proposed approach. Also, the repeated messages were not focused in [21] and [23], which increased the buffer and caused data collision during transmission. Such repetition among messages is recognized through the hash value and neglected using the SHA-512 technique. Hence, the proposed work develops an efficient buffer management system in DTN by tackling the existing downsides in the prevalent works.

METHODOLOGY

BM is severely needed for reducing data loss. When the buffer is full, it can be difficult to accommodate fresh messages in DTNs due to their limited size. The DTN environment requires an appropriate BM policy. This paper presents the advanced approach for BM. The repeated data is removed, and the clustering proceeds to drop the message during overflow. Then, the message is scheduled and forwarded via the optimal path. The block diagram for the proposed methodology is exhibited in Figure 1.

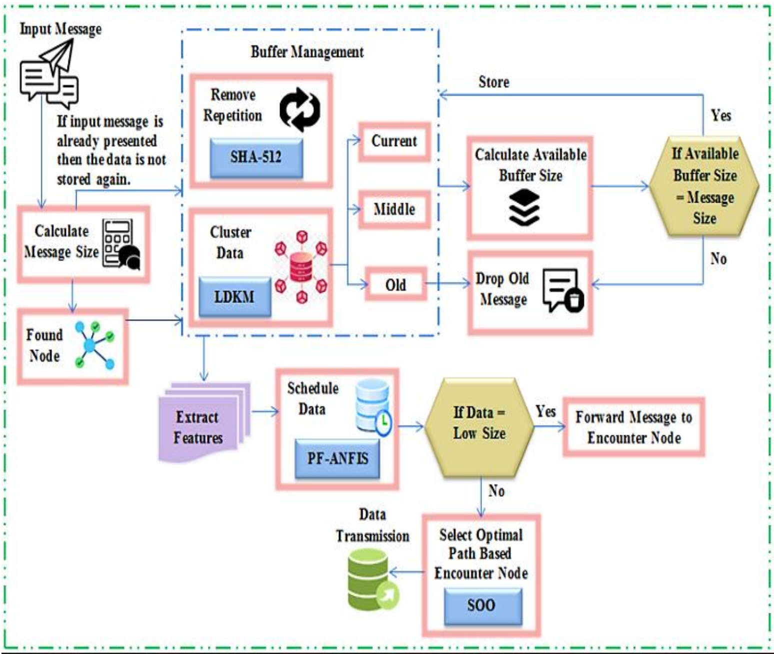

Figure 1. Block Diagram for the Proposed Methodology

Input Data

Primarily, the size of the entered message is calculated, and the message is presented to the found node. The message is denoted as,

$$A_M = \{A_1, A_2, \dots, A_N\} \tag{1}$$

Here, A_M defines the entered messages.

Buffer Management

For the storage of message buffer space availability is checked, and repeated data are removed. Initially, check whether the input message is already present. If it is present, then the data is not stored again. For repeated data removal, the SHA-512 algorithm is used, and the messages in the buffer are clustered for dropping.

Repetition Removal

For the repetition, the SHA-512 process has three steps. Thus, the hash value is checked with the buffer hash values. If the hash value is matched, then the repeated hash-related messages are deleted.

$$BB_M = \begin{cases} discard & if\left(A_M == BB_M\right) \\ store & else \end{cases} \tag{2}$$

Where, BB_M denotes the stored buffer message.

Data Clustering

After that, the messages in the buffer are clustered to reduce the complexity of message dropping using the LDKM algorithm. The convergence process is better than the other grouping methodologies, so this research chose this algorithm. However, it isn't clustered non-linear data correctly. To support the non-linear data, this research methodology uses the Lorentz distribution function to fit arbitrarily shaped data. The clustering process is based on the Time To Live (TTL) metric. The TTL of buffer messages B_M is denoted as,

$$B_M = \{B_1, B_2, ..., B_N\} \tag{3}$$

The input data are shaped W_x by using the Lorentz distribution, which is expressed in equation (4),

$$W_x = \frac{1}{\pi\left(1 + \left(B_M^2\right)\right)} \tag{4}$$

From the input values, the centroid values are randomly selected. The selected centroid C_M is given as,

$$C_M = \{C_1, C_2, ..., C_N\} \tag{5}$$

For clustering, the distance between each TTL of the buffered message and all centroids dis is calculated.

$$dis = \sqrt{\sum_{n=1}^{N} \left(B_M - C_M\right)^2} \tag{6}$$

The obtained cluster is denoted as,

$$U_c = \{u_1, u_2, u_3, ..., u_N\} \tag{7}$$

Where, U_c defines the clustered message. The pseudo-code of LDKM is shown below,

Input: Buffer messages BB_M

Output: clustered messages $U_c = \{u_1, u_2, u_3, ..., u_N\}$

Begin

Initialize TTL of messages $B_M = \{B_1, B_2, ..., B_N\}$, centroids C_M

For messages

  Elect the cluster center

  $C_M = \{C_1, C_2, ..., C_N\}$

End for

For each remaining data

Calculate distance $dis = \sqrt{\sum_{n=1}^{N} \left(B_M - C_M\right)^2}$

Assign the node to $C_M = \{C_1, C_2, ..., C_N\}$

End for

Return the number of clusters $U_c = \{u_1, u_2, u_3, ..., u_N\}$

End

Here, the output of the cluster message is old, current, and middle with respect to TTL. If the buffer space is available, then store the message otherwise drop the message from the old, clustered part and store the entered message into the buffer.

Feature Extraction

From the stored messages, the features are extracted for transmitting the messages regarding their important features and neglecting the irrelevant characteristics of the data. This also aids in reducing the time consumption and complexity of the proposed communication system. Here, features, such as speed of data, message cost, weight, number of requests, data size, type of data, and arrival time are extracted to schedule the messages for data forwarding. The extracted feature ψ_f is denoted as,

$$\psi_f = \{\psi_1, \psi_2, \ldots\ldots, \psi_n\} \tag{8}$$

Schedule Data

Here, based on the extracted features, the messages are scheduled to avoiding congestion and delay during data transmission using a Probabilistic Function-based Adaptive Neuro-Fuzzy Inference System (PF-ANFIS). The reason for selecting the algorithm is that it has a rule-based process, so it is highly helpful for the scheduling process. However, it has a computational burden problem due to the gradient learning process. To tackle that problem, this research approach utilizes the probabilistic function. The ANFIS is presented by rules, which are specified in below equations,

$$Rs_i = o_i * \psi_1 + p_i * \psi_2 + v_i * \psi_n + x_i \tag{9}$$

Where, ψ_1, ψ_2, ψ_n are the inputs, o_i, p_i, v_i are the consequent parameters, and x_i is the output fuzzy membership functions. The structure of PF-ANFIS is shown in Figure 2,

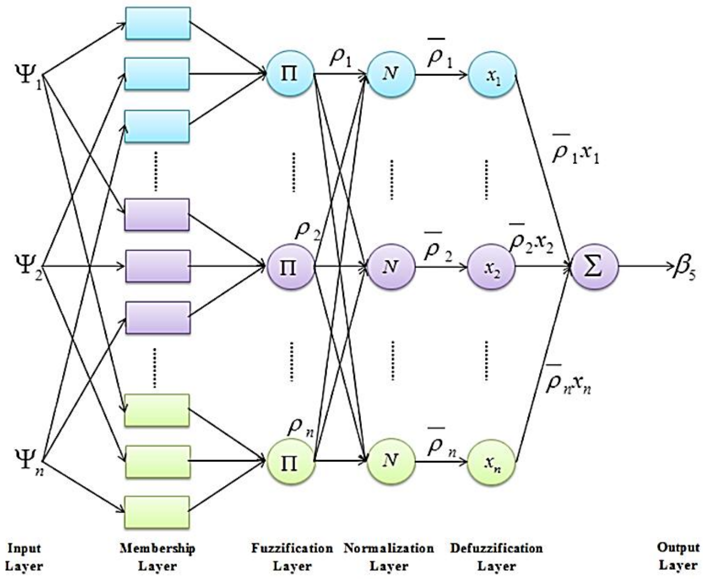

Figure 2. Structure of PF-ANFIS

The first layer is the Fuzzification Layer β_1, which is derived based on the Gaussian membership function GG, and is expressed as,

$$GG = aa.\exp\left\{-\frac{(\psi_n - bb)}{2cc^2}\right\} \tag{10}$$

Where, aa, bb, cc are the parameter set,where aa represents the shape of the Gaussian curve, bb denotes the mean or peak of the Gaussian membership curve, and cc implies the standard deviation for determine the curve distribution of the Gaussian curve. The 2nd layer is the implication layer,

$$\beta_2 = \rho_i = \psi_1 \times \psi_2 \times \psi_n \tag{11}$$

Here, ρ_i defines the weight of the neuron. The 3rd layer is the normalization layer,

$$\beta_3 = \overline{\rho}_i = \frac{\rho_i}{\sum \rho_i} \tag{12}$$

The fourth layer is the defuzzification layer. The node output is derived as follows,

$$\beta_4 = \overline{\rho}_i.Rs_i \tag{13}$$

The final layer is the output layer, which is the sum of all inputs. The final output is derived as follows,

$$\beta_5 = \tau + \frac{\sum \rho_i . Rs_i}{\sum \rho_i} \tag{14}$$

Where, τ denotes the probabilistic function, which is the multiplication of hyper-parameter and random value. Equation (14) represents the final output retrieved from the PF-ANFIS technique, which is attained by the processing of message features through the various layers of the proposed PF-ANFIS.

Based on the PF-ANFIS outcome, the messages are scheduled for transmission.

Optimal Path Selection

After scheduling, if the message is low, then it is forwarded to the receiver via all the encounter nodes otherwise the message is forwarded via the optimal path-based encounter node to reduce the complexity and data loss of the network by the SOO algorithm. The data having minimum size has some characteristics, including lower corruption possibility, lesser transmission time, congestion resistance, traffic scalability, and so on. Thus, the low-sized data are directly forwarded to the encountered node without considering the transmission path. On the other hand, optimal path selection is essential for forwarding large-sized data with improved data transmission rates. Thus, the SOO algorithm is used, which selects the food from the sea, and the proposed research also searches paths from many paths of the DTN, so here, the osprey algorithm is chosen. The osprey selects the position randomly, which may tend to the local optimum problem. So, this research methodology uses the secant method in the position updating process.

For selecting the optimal path, the SOO algorithm utilizes the scheduled messages as input, which is initialized similarly to the initialization of ospreys in the search space. Then, the fitness for choosing the optimal path is determined regarding minimum delay and maximum delivery rate. As the same as the ospreys search for prey and update their position, accordingly, the best path is investigated based on the fitness function. The position where the ospreys attain the prey produces the best fitness at the maximum iteration. Likewise, the best path is achieved for forwarding messages when the fitness function is satisfied. The position of the osprey that reaches the prey in SOO is related to the optimal path, where the messages are made forward. These are the correlation processes between the SOO and the optimal path for forwarding messages.

Here, the ospreys are considered as the encounter nodes path. Then, The ospreys' position in the search space is randomly initialized using the following equation:

$$Q = \begin{bmatrix} Q_1 \\ . \\ Q_i \\ . \\ Q_N \end{bmatrix}_{N\times m} = \begin{bmatrix} q_{1,1} \cdots q_{1,j} \cdots q_{1.m} \\ . \quad\quad . \quad\quad . \\ q_{i,1} \cdots q_{i,j} \cdots q_{i,m} \\ . \quad\quad . \quad\quad . \\ q_{N,1} \cdots q_{N,j} \cdots q_{n,m} \end{bmatrix}_{N\times m} \tag{15}$$

$$q_{ij} = \varepsilon_j + \lambda_{ij} . \left(c_j - w_j\right), \ i = 1,2,....,N, \ \ j = 1,2,......,m \tag{16}$$

Where, m is the number of problem variables, λ_{ij} are random numbers, Q specifies the population matrix of ospreys' locations, $q_{i,j}$ is its j^{th} dimension, N denotes the number of ospreys, ε_j and c_j are the lower bound and upper bound of the j^{th} problem variable, correspondingly. Here, minimum delay and maximum delivery are considered as the fitness function f_{fun}.

$$f_{fun} = \left(\min\left(d_y\right), \max\left(d_r\right)\right) \tag{17}$$

$$d_y = K\frac{LT}{TR} \tag{18}$$

$$d_r = \frac{T_d}{T_g} \tag{19}$$

Here, d_y and d_r indicates the delay and delivery rate, T_d and T_g specifies the total message delivered and total message generated, and LT denotes the message length over K links of transmission rate TR. The derived fitness Z is represented as per the given equation (20):

$$Z = \begin{bmatrix} Z(Q_1) \\ . \\ Z(Q_i) \\ . \\ Z(Q_N) \end{bmatrix}_{N\times 1} \tag{20}$$

Where, $Z(Q_N)$ specifies the fitness function of the N^{th} osprey. Before updation, the fish position FP_i is considered objective for osprey and is specified in below equation (21),

$$FP_i = \{Q_l \,|\, l \in \{1,2,....,N\} \wedge Z_l < Z_i\} \cup \{Q_{best}\} \tag{21}$$

Where, Q_{best} indicates the best candidate solution. Then, the osprey's new position is calculated as in equation (22),

$$q_{i,j}^{e1} = q_{i,j} + \zeta_{i,j} .\left(SS_{i,j} - \eta_{i,j} .q_{i,j}\right) \tag{22}$$

$$q_{i,j}^{e1} = \begin{cases} q_{i,j}^{e1} \ , \ \varepsilon_j \le q_{i,j}^{e1} \le c_j; \\ \varepsilon_j, \quad q_{i,j}^{e1} < \varepsilon_j; \\ c_j, \quad q_{i,j}^{e1} > c_j. \end{cases} \tag{23}$$

$$Q_i = \begin{cases} q_i^{e1}, & Z_i^{e1} < Z_{i,j} \\ q_i, & else \end{cases} \tag{24}$$

Here, $SS_{i,j}$ denotes the selected fish, i^{th}, $\zeta_{i,j}$ are random numbers in the interval [0, 1], and $SS_{i,j}$ represent the secant values for the population.

Input: encounter node paths

Output: selected path

Begin

Initialize population, fitness f_{fun}, iteration io_t, and maximum iteration m_t

Compute fitness

Set iteration $io_t = 1$

While $(io_t \le m_t)$ do

Calculate the position $q_{i,j}^{e1}$

If $\varepsilon_j \le q_{i,j}^{e1} \le c_j$ {

$q_{i,j}^{e1} = q_{i,j}^{e1}$

} else if {

$q_{i,j}^{e1} = \varepsilon_j$

} else {

$q_{i,j}^{e1} = c_j$

} end if

Calculate fitness

Set iteration $io_t = io_t + 1$

End while

Return selected path

End

The pseudo-code for the proposed SOO is given. Upon hunting a fish, the osprey then carries it to an appropriate place to eat it. For eating, the position is also searched according to the fitness. The heavy load data is only forwarded via the optimal path. The experimental analysis of the presented research is given in the further sections.

RESULTS AND DISCUSSION

In this section, the performance of the proposed BM model is studied by conducting numerous experiments in the working platform of PYTHON.

Simulation Setup

For evaluating the performance, the proposed work is implemented using the PYTHON software tool as mentioned above. The specific PYTHON software is a widely used high-level programming language. It was developed to emphasize the code readability and express the contents in fewer lines of the code. It also makes it work quickly and efficiently to integrate the system. Here, an isolated Python environment in the directory form is created from the virtual environment tool for the experiment. The hardware parameters used for the simulation are mentioned as follows.

Operating system-Windows 10

Processor-Intel i5/core i7

System type-64 bit

CPU speed-3.20 GHz

Random Access Memory (RAM)-4 GB

Further, the performance is assessed by utilizing the message from the Twitter dataset. This dataset contains thousands of tweets gathered from the Twitter platform. From this datasets, 80% of the data is used for training, and the rest 20% is used for training the proposed communication system.

Performance Analysis

To investigate the impact on network parameters, the proposed model was compared with the existing methods.

The accomplishments concerning the delivery ratio and delay rate of the proposed SOO and existing OO, Evolutionary structural optimization (ESO), Aquila Optimizer (AO), and Transient Search Optimization (TSO) methods are depicted in Figure 3. For the maximum buffer size of 100 MB, the proposed SOO attains a higher delivery ratio (0.815) and minimum delay rate (1217 s) than other existing models. This is profited by selecting a path to the encountered node under the consideration of scheduled data size.

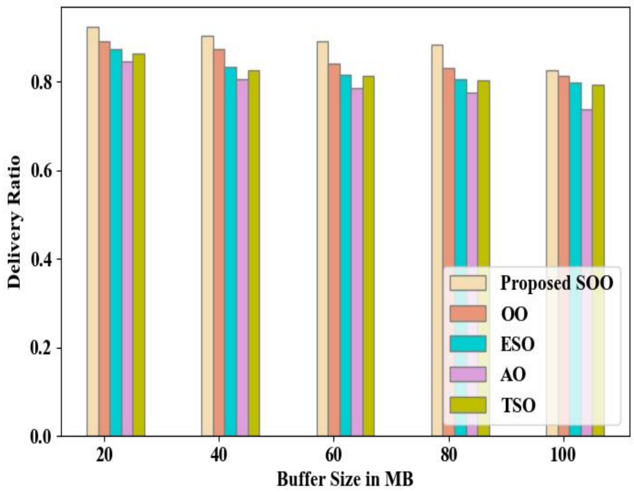

Figure 3(a). Performance Evaluation of SOO Regarding Delivery Ratio

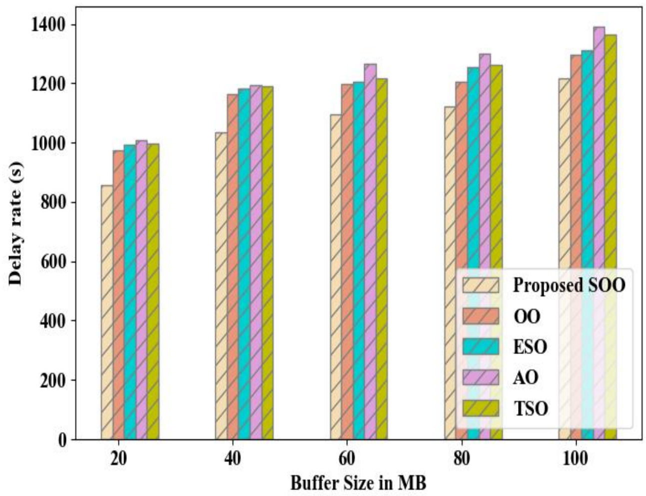

Figure 3(b). Performance Evaluation of SOO Regarding Delay Rate

The overhead ratio and the number of messages dropped outcomes of the proposed SOO and prevailing algorithms were compared in Figure 4. By varying the buffer size from 20 MB to 100 MB, the overhead and message drop of the proposed model ranges between 12 to 9.3 and 698 to 507, which are much less than the existing methods. Hence, the proposed model has excellent performance in adopting data clustering to determine the number of messages to be dropped, and it is sensitive to the data size during path selection.

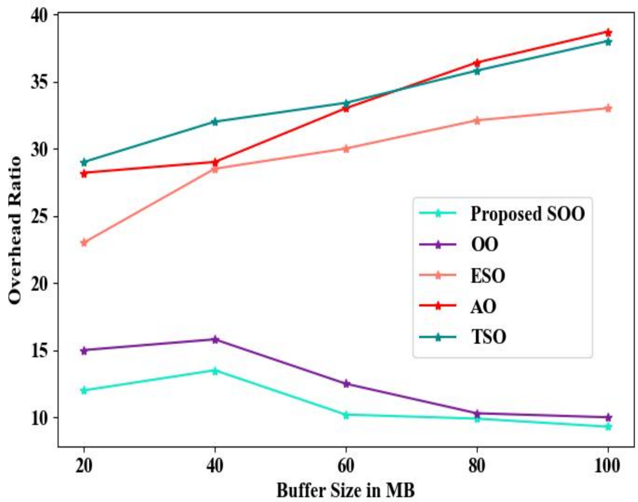

Figure 4(a). Performances of SOO Concerning Overhead Ratio

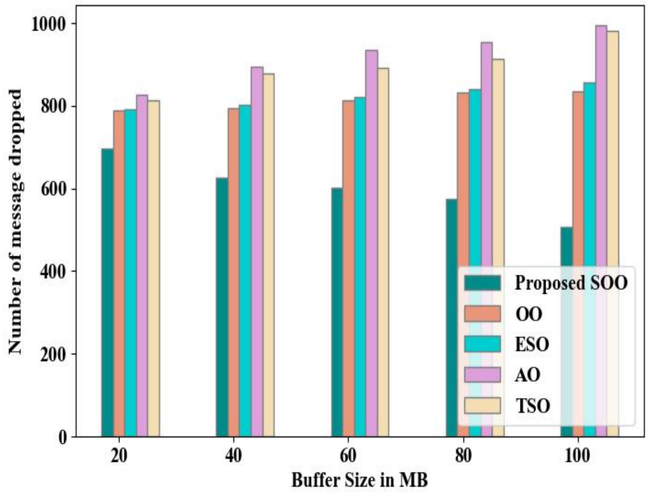

Figure 4(b). Performances of SOO Concerning the Number of Messages Dropped

As per Figure 5, the proposed model achieved better fitness values for all iterations than other schemes. For the 50th iteration, the fitness attained by the proposed model is improved by 9.4% than the existing OO algorithm. This shows that by solving the local optimum problem in the position updation process, the proposed SOO algorithm converges faster than the existing methods.

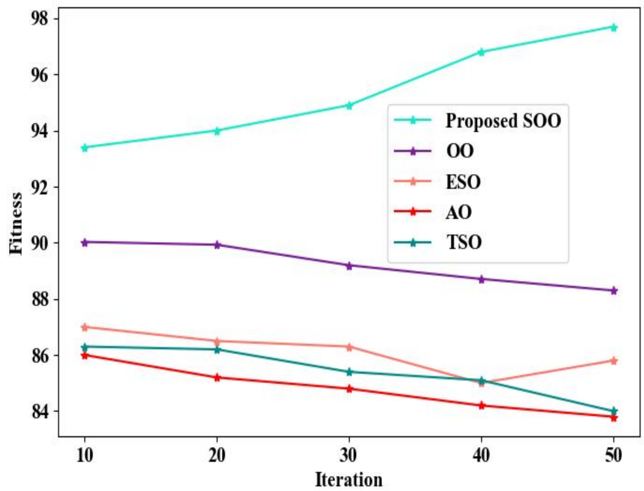

Figure 5. Fitness VS. Iteration

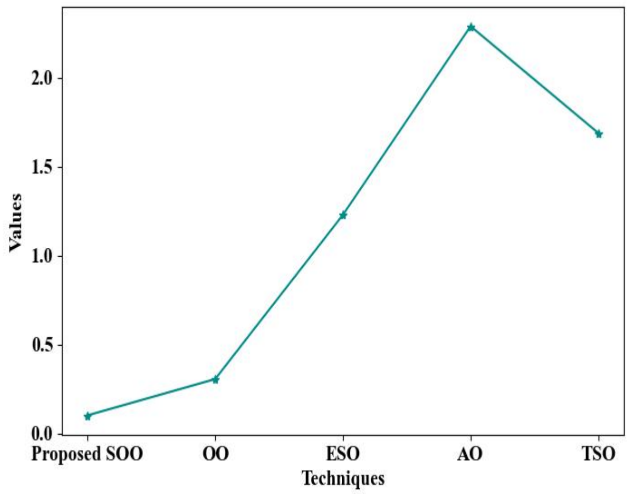

Figure 6. Analysis of Path Selection Error

Figure 6 depicts the path selection error by comparing the proposed SOO and existing OO, ESO, AO, and TSO methods. From Figure 6, it can be said that the error attained by the proposed method is lesser by 2.04% than the existing OO algorithm. Hence, the secant method used in path selection banned the proposed path selection method from being stuck in suboptimal solutions and reduced the loss in quality of its solutions.

Table 1. Performance Comparison of Proposed PF-ANFIS

Metrics/Methods	Proposed PF-ANFIS	ANFIS	DNN	RNN	ANN
Accuracy (%)	98.2	93	92.4	92	88.8
Precision (%)	96	91	89.7	89	86.8
Recall (%)	96.5	90.8	90	89.9	86.9
F-measure (%)	96.2	90.8	89.7	89.7	86.8
MCC (%)	95.6	93	92.5	90	88.3
FNR	0.032	0.045	0.08	0.12	0.15
FPR	0.042	0.075	0.09	0.13	0.18

Table 1 shows the effectiveness of data scheduling by the proposed PF-ANFIS and existing ANFIS, Deep Neural Network (DNN), Recurrent Neural Network (RNN), and Artificial Neural Network (ANN) methods. From Table 1, it can be said that the proposed model acquires better performance with higher values of accuracy, precision, recall, F-measure, and Mathews Correlation Coefficient (MCC) and minimum value of False Negative Rate (FNR) and False Positive Rate (FPR).

The accuracy of the proposed PS-ANFIS represents how accurately the messages are scheduled. It is calculated as,

$$Accuracy = \frac{TP + TN}{TN + FP + TP + FN} \tag{25}$$

Where, TP and TN denote True Positive and True Negative classes of the message, respectively,and FP and FN indicate the False Positive and False Negative classes, respectively. Here, TP denotes the correct scheduling of messages and TN indicates the true negative specification of the messages. Further, FP indicates the false positive scheduling of messages and FN represents the wrong scheduling of messages. Further, the precision of the proposed model is derived based on the estimated number of TP and FP data classes. It is defined as,

$$\Pr ecision = \frac{TP}{FP + TP} \tag{26}$$

Then, the proposed algorithm is analyzed regarding recall and is attained as 96.5. It is measured as,

$$\operatorname{Re} call = \frac{TP}{TP + FN} \tag{27}$$

Subsequently, the F-measure value achieved by the proposed model is 96.2, which indicates the scheduling performance of reliable messages. It is expressed as,

$$F - measure = 2 \times \frac{\Pr ecision \times \operatorname{Re} call}{\Pr ecision + \operatorname{Re} call} \tag{28}$$

Also, the FPR and FNR are evaluated for the proposed and existing techniques to justify the accurate message scheduling by the proposed technique. The FPR of 0.042 and FNR of 0.032 are attained by the proposed PF-ANFIS model and are derived by,

$$FPR = \frac{FP}{TN + FP} \times 100 \tag{29}$$

$$FNR = \frac{FN}{FN + TP} \times 100 \tag{30}$$

Furthermore, the MCC is assessed for the proposed and existing approaches. It is derived as,

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (31)$$

From the assessment, the proposed model achieved a higher MCC of 95.6% over the existing techniques in scheduling the messages.

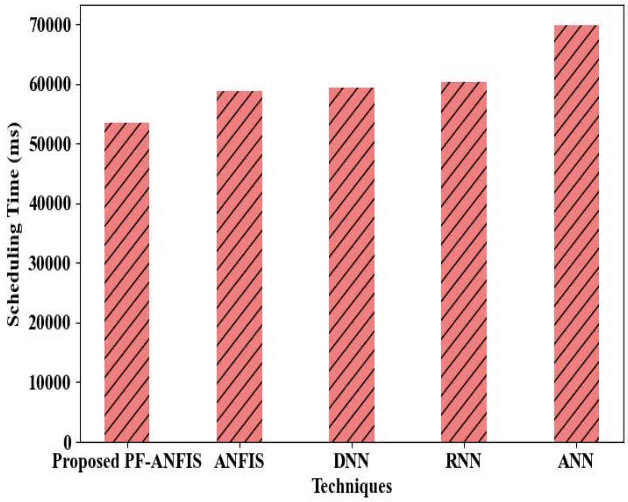

Figure 7. Performance Analysis Based on Scheduling Time

Figure 7 exhibits the performance of the proposed PF-ANFIS and prevailing data scheduling methods. The scheduling time of the proposed PF-ANFIS method is minimized by 5507 ms than the existing ANFIS. It is calculated as the difference between the beginning and ending time of scheduling. Hence, the proposed PF-ANFIS performs better than the existing methods due to the PF-based learning process that reduced the computation burden by exploiting the probabilistic relationships among the variables.

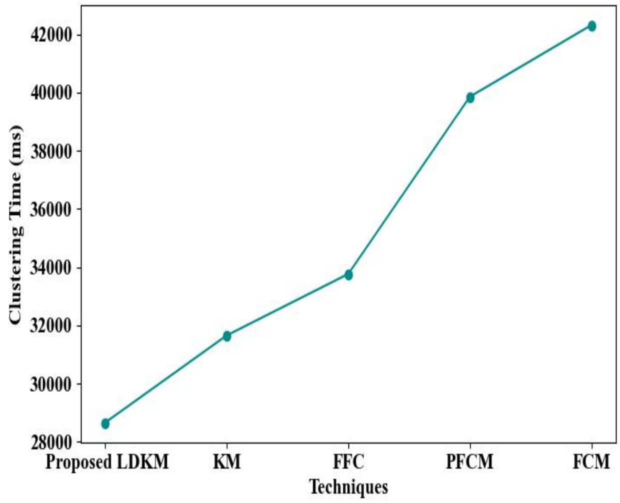

Figure 8. Performance Measure According to Clustering Time

Figure 8 compares the clustering time of the proposed LDKM and prevailing KM, Farthest First Clustering (FFC), Possibilistic Fuzzy C-Means (PFCM), and Fuzzy C-Means (FCM) methods. The proposed LDKM performs better than the existing methods due to even the clustering of non-linear data exactly with 28657ms. Hence, the analysis demonstrates that the LD-based clustering processes are well suited for supporting the non-linear data.

Comparative Analysis

Here, the proposed data size-centric routing and message drop strategy is compared with the existing BM schemes developed by [16], [24], [25], [26], [27].

Table 2 compares the proposed model's performance with the existing Inactive Node Detecting (InD), Coalition Formation-based Cooperation strategy (CF-C), Vehicular Delay Tolerant Network (VDTN), MaxDelivery, and Spectrum Aware-DTN (SA-DTN) BM schemes. In comparison, the proposed method attains better results than the existing methods. To simplify BM, the existing methods are not suitable, as they ignore the data size and type during the transmission. This makes the sharing and transmission of large data challenging in terms of processing speed and storage capacity. However, the proposed model employed an efficient optimal path selection-based routing strategy based on the data size for transmission. This enhanced the superiority of the proposed model.

Table 2. Comparative Analysis

Author Name	Method	Buffer Size= 10 MB			
		Delivery Ratio	Delay Rate	Overhead Ratio	Message Drop
[2]	InD	21%	-	-	12,345
[3]	CF-C	48.5%	3491 sec	52	-
[6]	VDTN	90%	898 ms	45	-
[15]	MaxDelivery	82%	-	-	-
[17]	SA-DTN	90%	15 min	-	-
Proposed Model	SOO	92.3%	856sec	12	698

CONCLUSION

In this paper, an efficient path selection framework for large data transmission and BM in DTN is proposed. Data clustering-based message drop and hashing-based data storage significantly enhanced the proposed model's outcome. The proposed SOO, PF-ANFIS, and LDKM methods were simulated by comparing them with the existing methods, such as ANFIS, DNN, RNN, ANN, OO, ESO, AO, TSO, InD, CF-C, VDTN, Max Delivery, and SA-DTN. The selection of optimal path and data scheduling enhanced the delivery rate of messages using the proposed SOO and PF-ANFIS techniques. As per the experimental outcomes, the proposed model is more reliable with a high delivery ratio (81.5 %) and minimum delay rate (1217 s). Therefore, an effective communication system was developed for the DTN by using the proposed technique.

Even though the message is forwarded efficiently along the optimal path of the DTN, the network challenges, including energy usage, capacity, and trust level are not concentrated in this work. Hence, in the future, the work will be extended to reduce network complexity as it takes bulky time to drop messages when the network is very large.

ETHICAL DECLARATION

Conflict of interest: No declaration required. **Financing:** No reporting required. **Peer review:** Double anonymous peer review.

REFERENCES

[1] P. Chinnasamy, S. K. Udgata, K. Lalitha, and A. Jeevanantham, "Multi-objective based deployment of throwboxes in delay tolerant networks for the Internet of Things environment," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 895-907, 2021.

[2] T. Le, "Multi-hop routing under short contact in delay tolerant networks," *Computer Communications*, vol. 165, pp. 1-8, 2021.

[3] S. Hasan, A. H. Al-Bayatti, S. Khan, Y. Harrati, and A. Algaddafi, "Performance analysis of the impact of time to leave on different drop policies for delay tolerant networks," *Preprints 2022*, 2022, doi: 10.20944/preprints202210.0076.v1.

[4] A. Verma, Savita, and S. Kumar, "Routing protocols in delay tolerant networks: Comparative and empirical analysis," *Wireless Personal Communications*, vol. 118, pp. 551-574, 2021.

[5] F. Semsarieh and N. Derakshanfard, "Relay node selection based on adaptive neuro-fuzzy inference system in delay tolerant networks," unpublished.

[6] S. Jain and V. N. Soares, "A comprehensive survey of social based routing for delay tolerant networks," *Preprints 2021*, 2021, doi: 10.20944/preprints202105.0633.v1.

[7] S. Abbas and K. Ahmad, "Opportunistic routing protocols based on fuzzy logic: Present and future directions," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, Mar. 2021, pp. 1055-1060.

[8] M. Gurmessa, Y. Negesa, and S. Nagarajan, "Comparative analysis of simbet with TTL and simbet with social based message dropping policy routing protocol," in *Proceedings of the International Conference on Innovative Computing & Communication (ICICC)*, Mar. 2022, doi: 10.2139/ssrn.4385650.

[9] S. Hasan, M. Sharifi Sani, S. Iranmanesh, A. H. Al-Bayatti, S. Khan, and R. Raad, "Enhanced message replication technique for DTN routing protocols," *Sensors*, vol. 23, no. 2, p. 922, 2023.

[10] P. Söderman, K. J. Grinnemo, M. Hidell, and P. Sjödin, "A comparative analysis of buffer management algorithms for delay tolerant wireless sensor networks," *IEEE Sensors Journal*, vol. 21, no. 7, pp. 9612-9619, 2021.

[11] O. U Rehman, I. A. Abbasi, H. Hashem, K. Saeed, M. F. Majeed, and S. Ali, "SS-drop: A novel message drop policy to enhance buffer management in delay tolerant networks," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, 2021, doi: 10.1155/2021/9773402.

[12] S. Khan, K. Saeed, M. F. Majeed, S. A. AlQahtani, K. Aurangzeb, and M. S. Anwar, "NPBMT: A novel and proficient buffer management technique for internet of vehicle-based DTNs," *Computers, Materials & Continua*, vol. 77, no. 1, pp. 1303-1323, 2023.

[13] Savita and D. K. Lobiyal, "Inter-contact delay and location information-based routing with adaptive threshold buffer management for delay tolerant networks," *International Journal of Information and Communication Technology*, vol. 17, no. 4, pp. 307-328, 2020.

[14] S. Rashid and Q. Ayub, "Social community buffer management policy for delay tolerant network," *Wireless Personal Communications*, vol. 130, no. 3, pp. 2099-2120, 2023.

[15] J. Yao, C. Ma, H. Yu, Y. Liu, and Q. Yuan, "A utility-based buffer management policy for improving data dissemination in opportunistic networks," *China Communications*, vol. 14, no. 7, pp. 1-9, 2017.

[16] Z. Du, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y, Ji, "A VDTN scheme with enhanced buffer management," *Wireless Networks*, vol. 26, pp. 1537-1548, 2020.

[17] 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.

[18] A. Karami and N. Derakhshanfard, "RPRTD: Routing protocol based on remaining time to encounter nodes with destination node in delay tolerant network using artificial neural network," *Peer-to-Peer Networking and Applications*, vol. 13, pp. 1406-1422, 2020.

[19] W. Wang, Y. Bai, P. Feng, J. Huang, M. Sha, and J. Tantai, "DTN-Balance: A forwarding-capacity and forwarding-queue aware routing for self-organizing DTNs," *Wireless Personal Communications*, vol. 118, pp. 575-598, 2021.

[20] Y. Mao, C. Zhou, J. Qi, and X. Zhu, "A fair credit-based incentive mechanism for routing in DTN-based sensor network with nodes' selfishness," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, p. 232, 2020.

[21] H. Lin, J. Qian, and B. Di, "Learning for adaptive multi-copy relaying in vehicular delay tolerant network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 3, pp. 3054-3063, 2024.

[22] J. Wu, H. Jin, S. Cai, and L. Liu, "A delay tolerant network routing algorithm based on multi-step double Q-learning," *IET Communications*, vol. 17, no. 11, pp. 1321-1333, 2023.

[23] A. Balaram, T. Sakthivel, and R. R. Chandan, "A context-aware improved POR protocol for delay tolerant networks," *Automatika: Časopis Za Automatiku, Mjerenje, Elektroniku, Računarstvo i Komunikacije*, vol. 64, no. 1, pp. 22-33, 2023.

[24] Q. Ayub and S. Rashid, "Energy efficient inactive node detection-based routing protocol for delay tolerant

network," *Wireless Personal Communications*, vol. 116, pp. 227-248, 2021.

[25] T. L. Babu and S. Chitnis, "Coalition formation based cooperation strategy for routing in delay tolerant networks," *Materials Today: Proceedings*, vol. 45, pp. 8182-8187, 2021.

[26] C. S. Nănău, "MaxDelivery: A new approach to a DTN Buffer Management," in *2020 IEEE 21st International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*, Aug. 2020, pp. 60-61.

[27] V. K. Shah, B. Luciano, S. Silvestri, S. Bhattacharjee, and S. K. Das, "A diverse band-aware dynamic spectrum access network architecture for delay-tolerant smart city applications," *IEEE Transactions on Network and Service Management*, vol. 17, no. 2, pp. 1125-1139, 2020.