Jurnal Teknik Informatika (JUTIF) Vol. 6, No. 4, August 2025, Page. 2239-2252
P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

Utility-Based Buffer Management for Enhancing DTN Emergency Alert Dissemination in Jakarta's Urban Rail Systems

Agussalim*[1], Nguyen Viet Ha[2], Handie Pramana Putra[3], Ma'ratul Adila[4], I Gede Susrama Mas Diyasa[5], Basuki Rahmat[6]

[1,3,4,5,6]Master of Information Technology, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Indonesia
[2]Faculty of Electronics-Telecommunications, VNUHCM-University of Science, Ho Chi Minh City, Viet Nam, Vietnam

Email: [1]agussalim.si@upnjatim.ac.id

Received : Jul 5, 2025; Revised : Aug 16, 2025; Accepted : Aug 17, 2025; Published : Aug 19, 2025

Abstract

The efficiency of emergency alert dissemination in highly populated and densely urban transport networks, such as Jakarta's integrated rail system, is undermined by sporadic connectivity and limited network resources. In this environment, an initial comparison of baseline Delay-Tolerant Network (DTN) routing protocols revealed that flooding-based routers, such as Epidemic, while achieving above-average delivery rates, suffered from high overhead and poor buffer utilization. This paper fills this gap by proposing the Combined Utility Router, a novel buffer management policy that overcomes the limitations of naive strategies, such as Drop-Oldest. Our approach holistically evaluates a message's value by assigning a weighted utility function based on its Time-To-Live (TTL), estimated total replicas, message size, and a user-defined priority. The router maintains high-value messages by discarding the message deemed the lowest utility score under the buffer constraint. Utility-based simulations in The ONE simulator demonstrate that applying our approach to Epidemic routing improves delivery probability, reduces average latency in high network congestion scenarios, while maintaining overhead rates. This work confirms that, in the context of developing reliable and efficient emergency communication systems for challenging urban topographies, optimizing buffer management extends beyond simply selecting the appropriate protocol.

Keywords : *Buffer Management, DTN, Drop Policy, Emergency Alert, Utility Based.*

1. INTRODUCTION

Effective communication is crucial for safety and security in public and densely populated urban areas, as exemplified by Jakarta's integrated public transport system [1, 2]. Timely alerts during emergencies such as natural disasters and accidents can minimize destruction and loss of life. However, urban and metropolitan transportation systems with high node mobility, signal obstruction from tall buildings, and congestion suffer from sparse intermittent connectivity, a feature of mobile urban environments [3]. Delay-Tolerant Networks (DTNs) with a Store-Carry-Forward (SCF) strategy provide a robust solution under such conditions [4-7].

In our previous paper, an initial investigation of the topic analysed the use of Delay Tolerant Networks (DTNs) for emergency alert dissemination throughout Jakarta's railway system, evaluating the standard routing protocols Prophet, Spray and Wait, ProphetV2, and Epidemic [8]. The results showed a severe compromise: with flooding-based protocols Epidemic, while message delivery rates were high, the system suffered from significant overhead and inefficiency, high latency, and waste of scarce buffer resources. This initial study indicated that for DTNs to be practically implemented, there was a need to either refine standard protocols with sophisticated routing logic or, more fundamentally,

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

with more innovative buffer management techniques. This finding supports a broad consensus in the DTN research community that, in situations where a node's buffer is complete, the buffer management policy becomes critical. The policy for selecting which message to drop, especially in a resource-limited and congested scenario, is as important as the policy for forwarding the message.

Default policies such as Drop-Oldest (FIFO) and Drop-Random are described as "context-unaware" [9-11]. These policies often result in suboptimal performance, for example, removing a message that has spent considerable time in transit and is close to its destination, to make space for a newer and less important message. It has led to the development of more intelligent and context-aware policies of dropping messages. A dominant paradigm in this area is utility-based buffer management, which treats the problem of determining which message to drop as an optimization problem [12, 13]. This method assigns a "utility" score to each message, and the message with the lowest score is dropped. Balasubramanian et al. [14] in their RAPID protocol pioneered the work of routing and buffer management as a resource allocation problem based on utility, which provided significant performance improvements.

The methods of calculating this utility have been researched extensively. Common heuristics involve dropping messages with the shortest Time-To-Live (TTL) because they will expire shortly. [15-19]. Another approach is to drop messages with the largest hop count, which represents the degree of replication of a message in the network, as a rough estimate of the message's replication [20]. More advanced works have suggested the use of multiple metrics, such as combining size with TTL [21], or conflicting parameters with Fuzzy Logic in order to account for the uncertainty of DTN environments [22-25].

This policy explores an identified gap in existing interdisciplinary research. Motivated by the inefficient buffer policy in the Jakarta rail system study, we propose the Combined Utility Router. Instead of the naive drop policy used in the robust but inefficient Epidemic protocol, we propose enhancing it via a multi-metric utility function that computes the value of a message based on its TTL, hop count, size, and application-level priority. We argue that the delivery performance and latency of the network suffer in the presence of intelligent drop decisions when the underlying protocol's robust forwarding mechanism is maintained. The goal of this research is to verify that enhanced policy approaches, specifically focused on the drop policy, can lead to improvements in system reliability and efficiency in DTN-based emergency communications.

2. METHOD

This section explains the design of the utility-based algorithm along with the simulation setup and the evaluation metrics.

2.1. Utility Score Algorithm Design

The fundamental advancement of *CombinedUtilityRouter* is the supersession of the *getNextMessageToRemove* function. In this instance, the router determines the utility score of every message and then drops the one with the lowest utility score. For any message m, the total utility score (*U_Total*) is calculated as a weighted sum of four normalized sub-utilities, as illustrated in Equation (1).

$$U\_Total(m) = (w\_ttlcdotU\_ttl) + (w\_hopscdotU\_hops) + (w\_sizecdotU\_size) + (w\_priocdotU\_prio) \quad (1)$$

The weights (w) are configurable, allowing the policy to be adapted to different network scenarios or operational goals.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

1. **TTL Utility** (U_ttl)**:** Prioritizes messages with a longer remaining lifetime, giving them more opportunity to be delivered.

$$U\_ttl = fractextTTL * textremainingtextTTL * textinitial \quad (2)$$

2. **Hops Utility (**U_hops)**:** Uses hop count as an inverse proxy for replica count. Messages with fewer hops are considered rarer and thus more valuable.

$$U\_hops = 1 - minleft(1, fractextHopCounttextMaxHopsright) \quad (3)$$

3. **Size Utility** (U_size)**:** Retaining smaller messages is generally more efficient for buffer space utilization. This metric assigns higher utility to smaller messages.

$$U\_size = 1 - fractextMessageSizetextBufferSize \quad (4)$$

4. **Priority Utility** (U_prio)**:** Allows applications to assign an explicit importance level to messages, ensuring critical alerts are preserved.

$$U\_prio = fractextPriorityValuetextMaxPriority \quad (5)$$

The proposed pseudocode implements a message selection strategy for buffer management when the buffer is full. The algorithm selects the message with the lowest utility value for removal according to a combined utility metric.

```
// Message selection for dropping
    METHOD getNextMessageToRemove(excludeMsgBeingSent)
        messages = getMessageCollection()
        messageToDrop = NULL
        minUtility = MAX_DOUBLE_VALUE

        FOR EACH message IN messages DO
            IF excludeMsgBeingSent AND isSending(message.id) THEN
                CONTINUE
            END IF

            currentUtility = calculateUtility(message)

            IF currentUtility < minUtility THEN
                minUtility = currentUtility
                messageToDrop = message
            END IF
        END FOR

        RETURN messageToDrop
    END METHOD
```

The utility calculation pseudocode employs a multifactor utility function to assess the significance of a particular message for message drop. The utility calculation algorithm evaluates messages discriminatively through a cumulative score, which uses four primary components. Each of the components that define the diagram of the importance of messages attempts to achieve a balanced optimal value in terms of the probability of delivery, resources spent, and degree of handling. The utility calculation algorithm evaluates messages by integrating four primary components to produce a single normalized score. Each component is crafted to represent various elements of the importance of messages, ensuring balance in delivery probability, resources, and efficiencies in handling.

```
// Utility calculation
    METHOD calculateUtility(message)
        // 1. TTL component (higher is better)
        normTtl = message.ttl / this.msgTtl
```

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

```
        // 2. Hops component (lower is better)
        normHops = message.hops.size() / MAX_HOPS_NORMALIZATION
        utilityHops = 1.0 - MIN(1.0, normHops)

        // 3. Size component (smaller is better)
        utilitySize = 1.0 - (message.size / this.getBufferSize())

        // 4. Priority component (higher is better)
        priority = getMessagePriority(message)
        utilityPrio = priority / MAX_PRIORITY_NORMALIZATION

        // Weighted sum
        totalUtility = (wTtl * normTtl) +
                       (wHops * utilityHops) +
                       (wSize * utilitySize) +
                       (wPrio * utilityPrio)

        RETURN totalUtility
    END METHOD
```

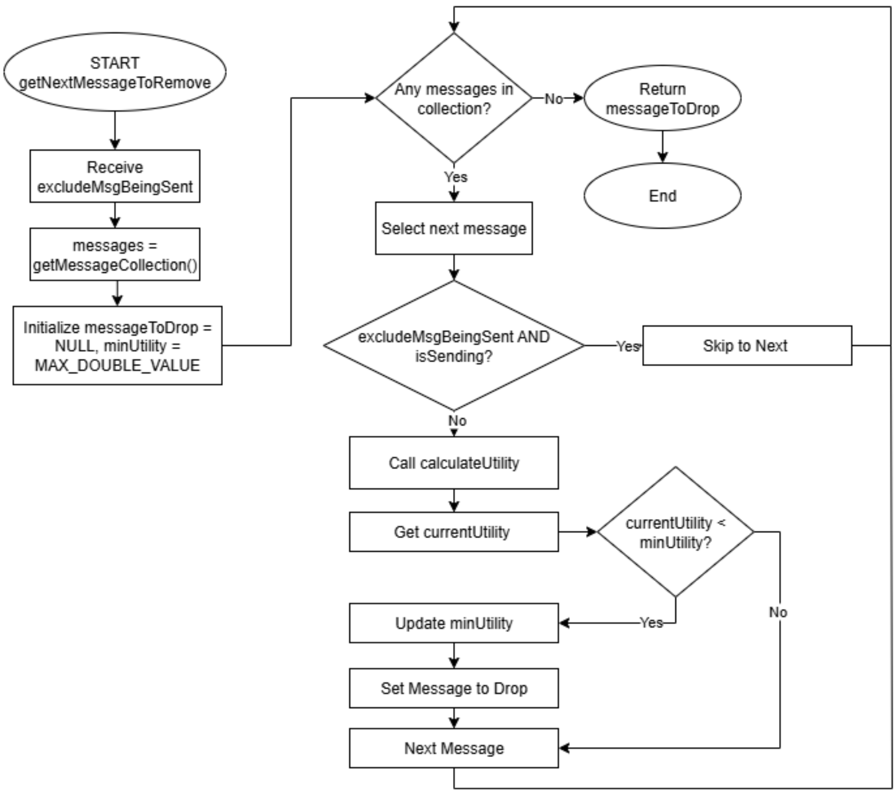

Figure 1. Flowchart of Message selection for *getNextMessageToRemove(excludeMsgBeingSent)*

The flowchart in Figure 1 illustrates the *getNextMessageToRemove(excludeMsgBeingSent)* method, which selects a message from a collection for removal based on the lowest utility value, with the option to exclude messages currently being transmitted. The process begins by retrieving the message collection and initializing variables *messageToDrop set to NULL* and *minUtility* set to a maximum value). It iterates through each message, skipping those being sent if *excludeMsgBeingSent* is true, and computes the utility of each message using the *calculateUtility* subprocess. The message with the lowest utility is selected for removal, ensuring efficient resource management in constrained systems, such as network buffers.

The *calculateUtility* subprocess, shown as a subgraph, calculates a message's utility based on four components: time-to-live (TTL), hop count, message size, and priority. These are normalized and combined into a weighted sum, reflecting the message's relative importance. The flowchart's modular design, with clear decision points and iterative loops, effectively illustrates the algorithm's logic, making it suitable for applications requiring optimized message prioritization and buffer management.

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

2.2. Simulation Environment and Scenario

This study builds upon our previous research [8] on the application of Delay Tolerant Network (DTN) technology for the emergency alert information dissemination system within the Jakarta Integrated Rail Systems. As previously illustrated in Figure 2, the focus of this paper is to develop a sophisticated buffer management technique to enhance DTN functionality within the specified operational area.

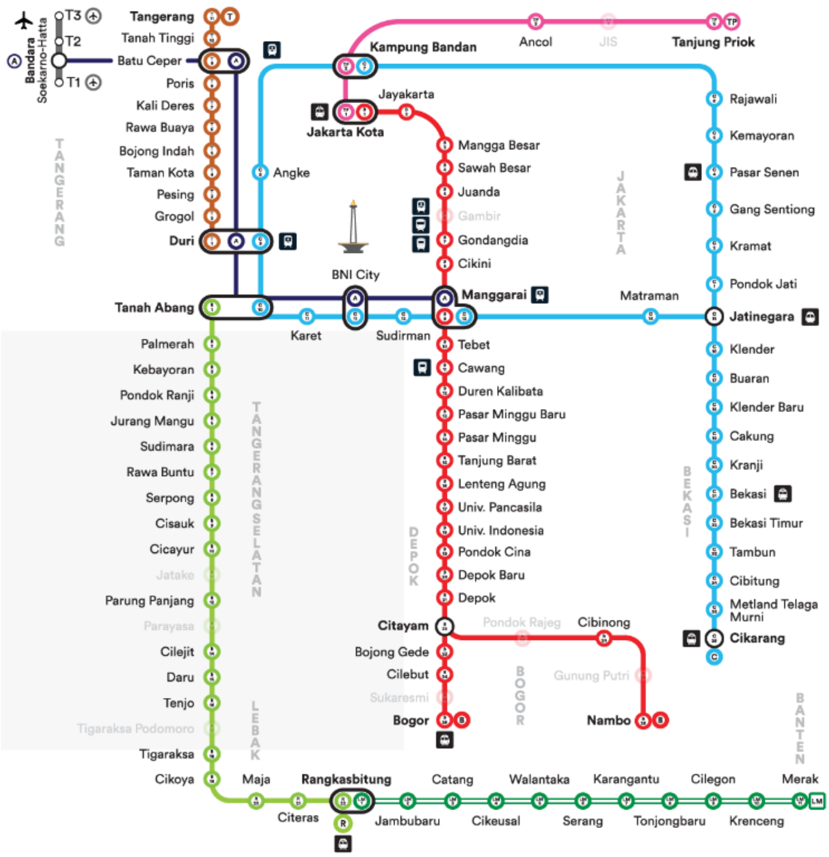

Figure 2. Jakarta Integrated Rail System Map (source: https://commuterline.id/perjalanan-krl/peta-rute)

The entire integrated transportation system of Jakarta is illustrated in Figure 2. For Jakarta, these include the Mass Rapid Transit (MRT), Light Rail Transit (LRT), and commuter rail services. According to the ridership statistics for June 2024, the MRT enabled the movement of 3.47 million passengers, representing a 9.41% increase from the previous month. Meanwhile, the LRT recorded 102,707 boardings, a 6.38% increase. Enhanced public transport systems in Jakarta are essential for improving metropolitan area mobility, alleviating traffic congestion, and increasing accessibility for the metropolitan region. Moreover, to protect public health, respond to natural disasters, or provide essential services, there is a need for timely and effective emergency information dissemination within the extensive network of public transportation systems.

As part of evaluation procedures, we opted for The ONE (v1.6.0) simulator [26]. The simulation environment was carefully set up to accurately emulate the Jakarta metropolitan area's erratic urban mobility patterns. It included the incorporation of the most important transport facilities: MRT Jakarta, LRT Jakarta, KRL Commuter Line, and the Airport Rail Link (Kereta Bandara) as outlined in Figure 3—the created urban mobility model simulated commuter movement for the available public transport systems. The user movements, or nodes, were adjusted to simulate real transit behaviour for each system

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

based on existing operational time tables, fixed pathways, and the number of passengers per unit time. Moreover, the model incorporated Jakarta's urban mobility infrastructure peculiarities, such as high spatial mobility, spatially intermittent network outages, time-varying traffic congestion, and other urban problems.

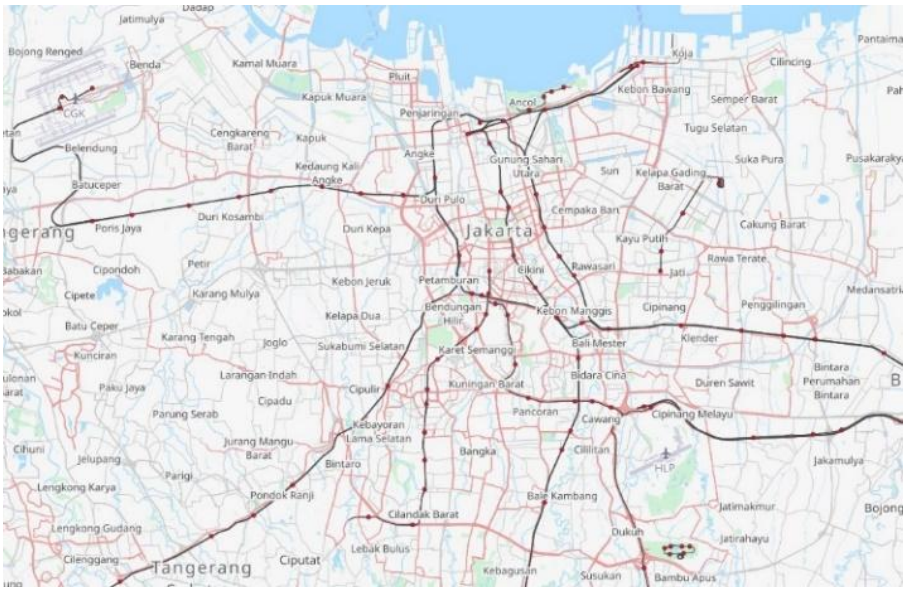

Figure 3 MRT Jakarta's Integration Map

In the urban mobility model, commuter flow and public transport modalities were interconnected through integration. Users, represented as nodes, were simulated to behave according to real travel patterns based on the system's operational schedules, fixed pathways, and passenger volumes. In addition, the model described the most significant urban problems, such as intermittent network outages and time-varying congestion, which are characteristic of Jakarta. The implementation of the Jakarta Integrated Transportation System in the context of the ONE Simulator framework is shown in Figure 4.

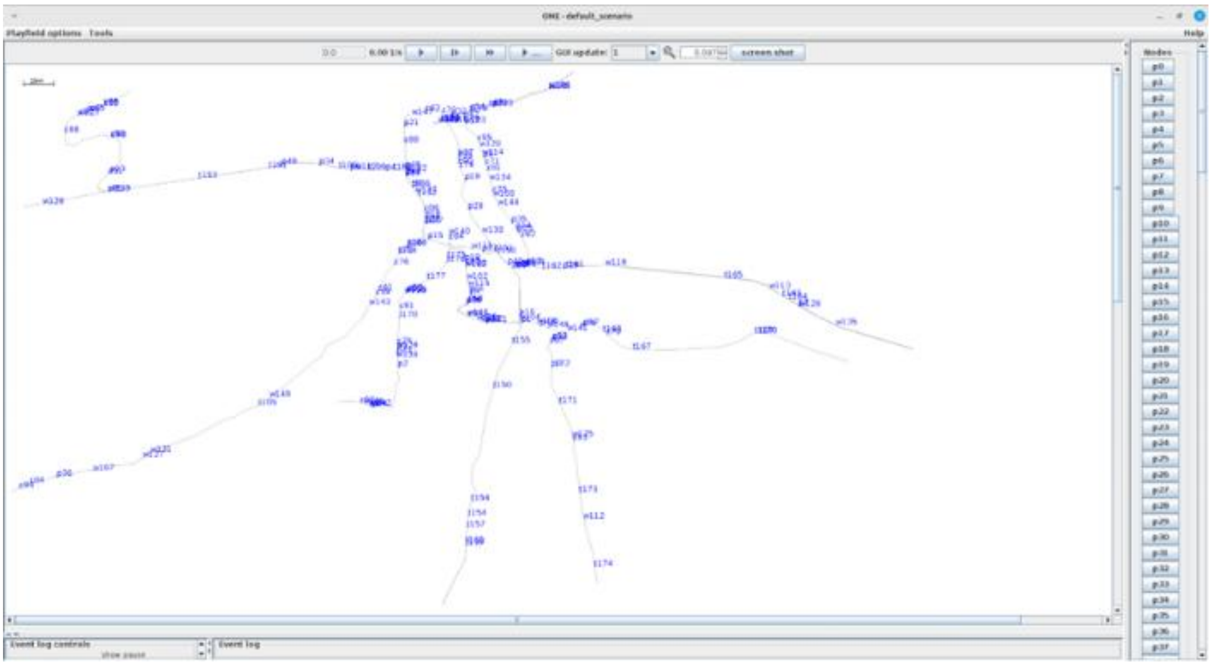

Figure 4. Modeling the Jakarta Integrated Rail System within the ONE Simulation Environment

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

Our proposed buffer management strategy was assessed against the two established methods: Last-In, First-Out (LIFO) and Drop Oldest. These buffer management techniques were embedded within the Epidemic Routing Protocol [27] to evaluate the performance gain from our buffer management strategy, even in a basic routing protocol. The simulation settings, outlined in Table 1, were selected to represent a densely populated urban DTN environment, aligning with the congestion issues described in the study of the Jakarta rail system.

Table 1. simulation parameters for the Jakarta integrated rail system scenario

Simulation Control	
Simulation Duration	12 h
Warm up time	1 h
Communication Network Attributes:	
Node Buffer Capacity	5 MB
Wireless Fidelity (Wi-Fi) Data Transmission Rate	10 Mbps
Wi-Fi Communication Proximity	15 m
Message Expiration Time (Time-to-Live)	5h
Mobility Model Components:	
Vehicle Velocities	Car 10 – 50 km/h Bus 20 – 40 km/h
Defined Rail Routes (trains)	Bandara Line route (4) Bogor Line route (7) Lingkar Cikarang route (6) LRT Bekasi route (4) LRT Cibubur route (4) MRT Utara Selatan (4) Priok Line (5) Rangkas bitung line (4) Tangerang Line (4)
Other Node	Pedestarian/passenger (50) Car (50)
Emergency Message Specifications	
Emergency Message Size	500 Kbytes
Emergency Message Generation Frequency	10s, 15s, 20s, 25s, 30s
Routing Protocol	Epidemic
utilityWeightTtl	2.0
utilityWeightHops	2.0
utilityWeightSize	0.5
utilityWeightPriority	1.0

The purpose of the simulation was to study the impact of the rate of emergency message generation on the efficiency of buffer management. In this case, the simulation was set to run for twelve hours, and five intervals of message generation were used: 10, 15, 20, 25, and 30 seconds. All the simulated devices were configured to have a 5MB buffer and communicated over Wi-Fi, which had a 10 Mbps data rate within a 15-meter radius. Messages were assigned Time-to-Live (TTL) values of 5 hours to ensure their timely delivery and prevent the network from being overloaded with outdated messages.

The mobility model was designed to align with the actual transportation geography of Jakarta. It included cars moving at 10 - 50 km/h and trains at 20 - 40 km/h. The simulation model included various modes of transport available in Jakarta, and added particular train lines such as the Bandara Line (4 trains) and Bogor Line (7 trains), as well as other lines listed in [16]. The simulated network had 50 nodes for pedestrians and 50 nodes for vehicles. Emergency messages of 500KB each were added to the

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

network at the specified intervals. To ensure network stability, a one-hour warm-up period was set aside before data collection began.

The weights for the utility function, as detailed in Table 1, were determined empirically through a series of preliminary simulations aimed at optimizing performance for emergency alert dissemination. The highest weights, *utilityWeightTtl = 2.0* and *utilityWeightHops = 2.0*, were assigned to prioritize messages with a longer remaining Time-To-Live (TTL) and those with fewer existing replicas in the network (inversely represented by hop count). This strategy is critical in emergency scenarios, as it favors the retention and propagation of newer, less-replicated messages that have a higher probability of reaching their destination before expiring. Conversely, a lower weight was assigned to message size *(utilityWeightSize = 0.5)*, as buffer space efficiency was considered a secondary objective to the timely delivery of critical alerts. While these fixed weights proved effective in our simulated scenario, a comprehensive sensitivity analysis and the development of dynamic, network-aware weighting mechanisms remain a promising direction for future research.

3. RESULT

This section provides thorough insights regarding the simulations conducted. It includes a comparative study of the three buffer management policies: Drop Oldest, LIFO (Last-In, First-Out), and our proposed Utility-based strategy. The policies were evaluated under different network loads, which we simulated by varying the emergency message generation intervals from 10 seconds (high load) to 40 seconds (lower load). Each of the policies was evaluated to measure their performance in terms of delivery probability, overhead ratio, average latency, and average buffer time, which are the four key metrics of interest. These metrics provide a holistic view of the system's reliability, efficiency, and timeliness:

Delivery Probability (P_{del}) is the most critical metric for an emergency alert system. It measures the ratio of unique messages successfully delivered to their destinations to the total number of unique messages generated in the network. A higher value indicates better reliability as shown in Formula 6.

$$P_{del} = \frac{\text{Number of Unique Messages Delivered}}{\text{Number of Unique Messages Created}} \quad (6)$$

Overhead Ratio (O_{ratio}) quantifies the network's efficiency by measuring the number of redundant transmissions for each successful delivery. It represents the total cost of routing, where a lower value signifies more efficient use of network resources as shown in formula 7.

$$O_{ratio} = \frac{\text{Total Relayed Messages} - \text{Total Delivered Messages}}{\text{Total Delivered Messages}} \quad (7)$$

Average Latency (Lavg) calculates the average time delay from a message's creation to its first successful delivery at its destination. In emergency scenarios, lower latency is crucial for ensuring timely communication. It is expressed Formula 8.

$$L_{avg} = \frac{\sum(\text{Time Delivered} - \text{Time Created})}{\text{Number of Delivered Messages}} \quad (8)$$

3.1. Analysis of Delivery Probability

Figure 5 illustrates the delivery probability, which is arguably the most important metric of an emergency alert system. The data clearly shows that the Utility-Based policy consistently outperforms other policies regardless of the network conditions. Even at the highest network load (10-second message generation interval), the Utility-Based approach yielded a delivery probability of approximately 0.31, which is more than 55% greater than the Drop Oldest policy (0.20) and 72% greater

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

than the LIFO policy (0.18). The degree of this advantage is not only sustained but further enhanced as network load is reduced. At a 40-second interval, the peak performance of the Utility-Based policy is 0.51 delivery probability, which is significantly 42% higher than Drop Oldest at 0.36.

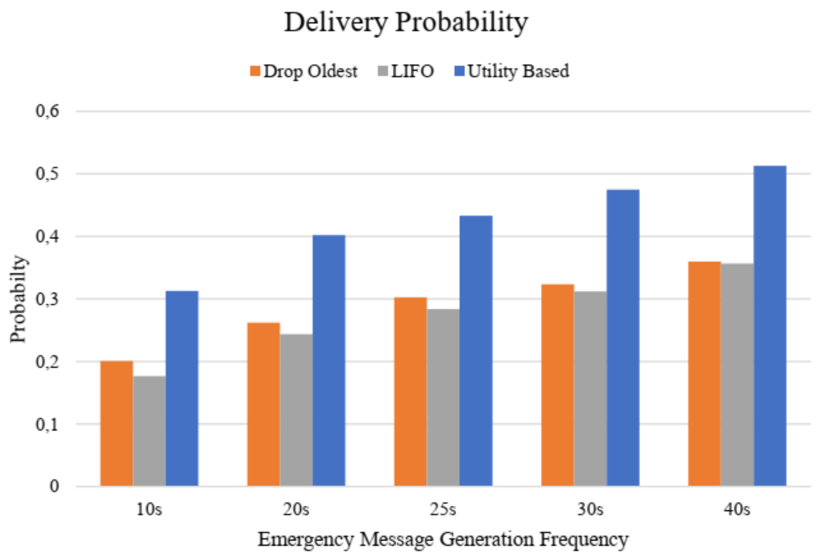

Figure 5. Delivery Probability

3.2. Analysis of Overhead Ratio

As depicted in Figure 6, each buffer management policy demonstrates a unique resource consumption profile. The LIFO policy induced a significantly high overhead ratio, surpassing 3600 relayed messages at lower network loads. In contrast, the Utility-Based policy generated around 2,650 relayed messages at the 40s interval. The Drop Oldest policy consistently produced the lowest overhead, generating just under 1,000 relayed messages in the same scenario.

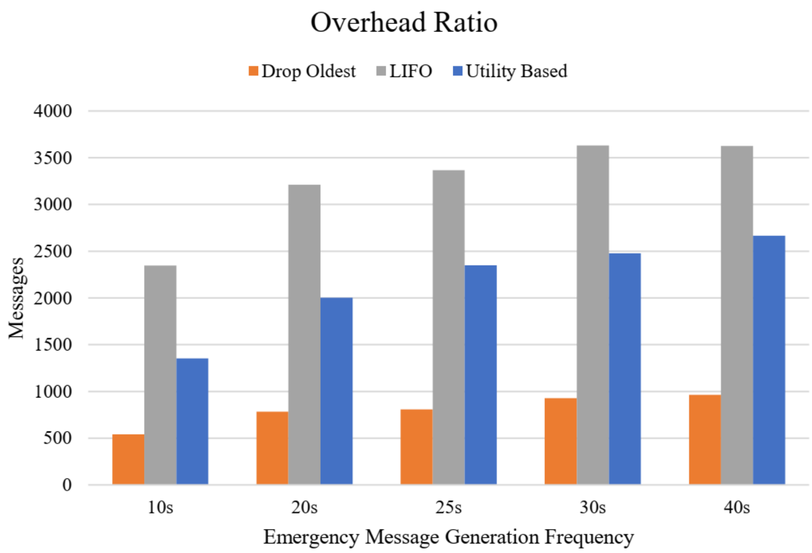

Figure 6. Overhead Ratio

3.3. Analysis of Average Latency

Figure 7 reveals the average message latency, further substantiating the effectiveness of the Utility-Based approach. It is evident that this policy significantly reduces latency, outperforming the

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

other two policies in all evaluated scenarios. Its average latency is consistently under 4,300 seconds, dropping to around 3,100 seconds at high network loads. In contrast, both the Drop Oldest and LIFO policies exhibit much higher latencies, ranging from 5,800 to over 7,300 seconds. This constitutes a statistically significant latency reduction of up to 47% compared to the Drop Oldest policy.

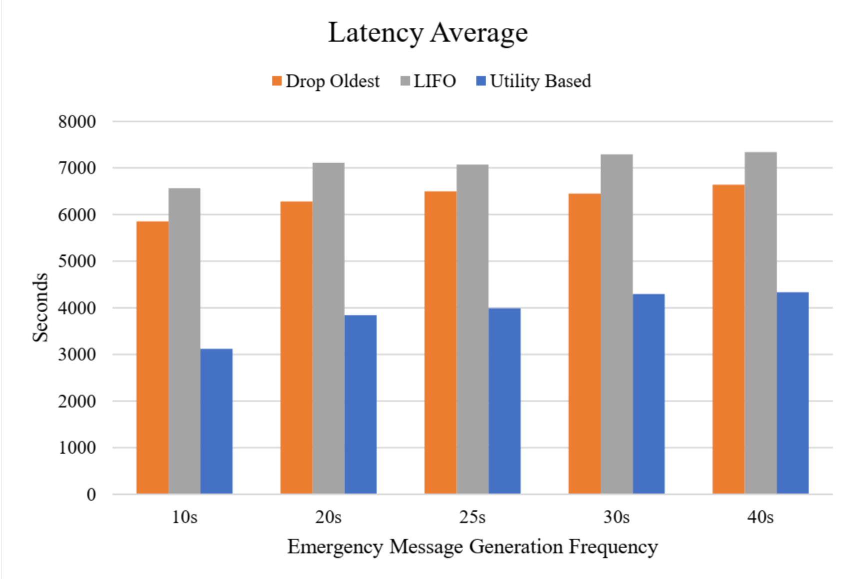

Figure 7. Latency Average

3.4. Analysis of Buffer Time Average

Figure 8 illustrates the internal state of the node buffers through the average buffer time metric. Under the Drop Oldest policy, the average buffer time exceeds 400 seconds across all scenarios. In sharp contrast, both the Utility-Based and LIFO policies exhibit a significant reduction, maintaining an average buffer time of approximately 100 seconds.

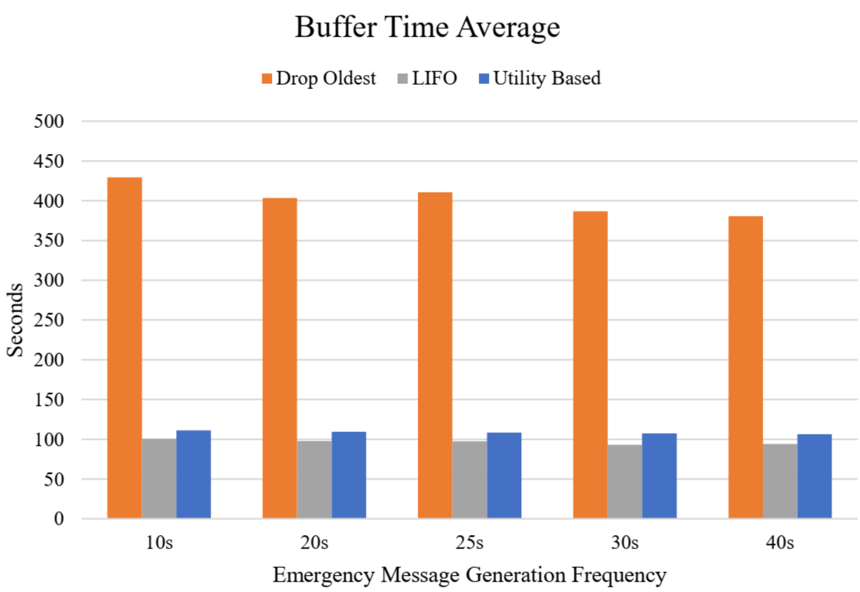

Figure 8. Buffer Time Average

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

4. DISCUSSION

This section provides a detailed interpretation of the simulation results, contextualizes the findings within the existing body of literature, and discusses the broader implications and limitations of this study. The results conclusively demonstrate that an intelligent buffer management policy is superior to naive, single-metric strategies for DTN-based emergency alert systems in congested urban environments.

4.1. Interpretation of Results

The superior performance of the Utility-Based policy across all key metrics is a direct consequence of its intelligent, multi-metric decision-making process. As shown by the delivery probability results in Figure 5, this efficient performance hinges on thoughtful decision-making stratified within the utility function. Unlike naive policies that depend on a single, arbitrary measure like message age, the utility-based paradigm determines the worth of each message by holistically assessing its Time-To-Live (TTL), estimated hop-based replica count, size, and priority. This capability helps retain messages that have a higher chance of successful delivery, such as newer messages that have not yet been extensively disseminated. In contrast, the Drop Oldest policy often discards messages that may still be relevant, while the LIFO policy's logic of dropping the newest messages is detrimental to dissemination, as it removes messages before they have a chance to propagate.

Furthermore, the remarkable reduction in average latency, shown in Figure 7, is attributable to the policy's proactive buffer management, which preserves a 'fresh' and 'valuable' state in the buffer. The utility function actively removes messages that are unlikely to aid in prompt delivery, such as those with a low remaining TTL or those that are over-replicated (high hop count). This prevents buffer stagnation and frees up resources to explore less congested routes. This proactive clearing mechanism is also reflected in the average buffer time analysis in Figure 8. While both Utility-Based and LIFO policies show low buffer times, the causes are opposed. LIFO's low time is an artificial result of prematurely dropping new messages, which harms delivery. In contrast, the Utility-Based policy's low buffer time represents a "healthy churn," treating the buffer as an active transit area rather than long-term storage. This ensures a fresh and relevant set of messages, which supports rapid forwarding and contributes to the impressive low-latency performance.

While the Utility-Based policy exhibits a moderately higher overhead ratio than Drop Oldest (as shown in Figure 6), this is a justifiable trade-off for enhanced reliability and speed. This relatively high overhead is indicative of higher delivery success; to deliver more messages, more forwarding and replication actions are necessary. The Utility-Based policy ensures these actions are meaningful by propagating high-value messages. The low overhead of the Drop Oldest policy is a misleading indicator of efficiency, as it stems from its ineffectiveness at propagating messages in the first place. In the context of an emergency alert network, this increased overhead is a preferable cost for achieving superior delivery probability and latency.

4.2. Comparison with Existing Literature

The findings of this study empirically confirm the sub-optimal nature of context-unaware buffer management policies critiqued in prior research. Our results, which show the poor performance of Drop Oldest and LIFO, align with the conclusions of other researchers who have also identified the weaknesses of such naive approaches. This work reinforces the consensus that in resource-constrained DTNs, the message dropping policy is as critical as the forwarding protocol itself.

Moreover, the success of our multi-metric utility function is consistent with the broader paradigm of intelligent, context-aware DTN management explored in other studies. The performance of our policy is in line with the findings of researchers who have utilized fuzzy logic and other multi-parameter

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

systems to enhance routing and buffer decisions. For instance, the work by Wu et al. on fuzzy-logic-based Q-learning also demonstrates the benefits of integrating multiple network parameters. While our approach uses a deterministic weighted sum tailored for the specific needs of an urban emergency alert system, it validates the same core principle: that considering a richer set of message attributes leads to demonstrably better network performance.

4.3. Implications and Limitations

In considering these findings, their relevance for emergency communications cannot be overstated. An efficient emergency notification system must be, first and foremost, dependable. The practical implication of this research is significant for developing resilient public safety communication systems in smart cities, particularly for a dense and disaster-prone metropolitan area like Jakarta. The proposed Utility-Based policy offers a viable and more reliable alternative to default DTN configurations, ensuring that critical alerts are disseminated with higher probability and lower delay. During an emergency, alert latency and delivery probability are the factors that matter most, and this policy ensures recipients receive information promptly for strategic planning and evacuations.

However, this study has several limitations that must be acknowledged. First, the findings are based entirely on simulations conducted within The ONE simulator. While the mobility model was designed to be realistic, real-world network conditions—including unpredictable signal interference, hardware constraints, and power limitations—were not fully captured. Second, the utility function relies on static, empirically chosen weights. These weights, while effective for our simulated scenarios, may not be optimal for all possible network conditions or traffic patterns.

These limitations highlight clear avenues for future research. The logical next steps include validating the proposed policy on a physical testbed of hardware nodes and developing adaptive weighting mechanisms, potentially using machine learning techniques like Reinforcement Learning, to allow the router to dynamically adjust its priorities based on real-time network feedback.

5. CONCLUSION

This study provides conclusive evidence that a strategic, utility-based buffer management policy is distinctly superior for Delay-Tolerant Networks applied to emergency alert systems in congested urban environments. Moreover, concludes that the proposed utility-based buffer management policy significantly enhances DTN performance for emergency alerts, achieving an improvement of up to 42% in delivery probability and a reduction of up to 47% in average latency compared to the standard Drop Oldest policy. The success of this approach is attributed to its holistic method of assessing and retaining high-value messages by evaluating multiple criteria, which ensures the buffer contains messages that can be delivered successfully and swiftly.

Although this approach leads to a moderate increase in overhead, the significant improvements in reliability and timeliness make it a rational and highly effective trade-off, demonstrating that a sophisticated buffer management approach is essential for building dependable and efficient DTN systems. Future work should focus on enhancing the system's intelligence, potentially by incorporating machine learning to develop adaptive drop policies, and bridging the simulation-to-reality gap by creating a Digital Twin of Jakarta’s integrated rail network for more robust evaluation. Advancing these research paths will be critical in transforming this theoretical framework into a life-saving tool for urban communication.

CONFLICT OF INTEREST

The authors declares that there is no conflict of interest between the authors or with research object in this paper.

ACKNOWLEDGEMENT

Thanks to the Master's Program in Information Technology, UPN "Veteran" Jawa Timur, which has become a platform for researchers to develop this journal's research. Hopefully, this research can make a significant contribution to the advancement of technology in Indonesia.

REFERENCES

[1] R. Martinez and I. N. Masron, "Jakarta: A city of cities," *Cities*, vol. 106, no. 102868, p. 102868, Nov. 2020, doi: https://doi.org/10.1016/j.cities.2020.102868.

[2] R. Oktorini and L. Sari Barus, "Integration of Public Transportation in Smart Transportation System (Smart Transportation System) in Jakarta," *Konfrontasi: Jurnal Kultural, Ekonomi dan Perubahan Sosial*, vol. 9, no. 2, pp. 341–347, Jun. 2022, doi: https://doi.org/10.33258/konfrontasi2.v9i2.223.

[3] National Academies of Sciences, Engineering, and Medicine, *Emergency Alert and Warning Systems*. Washington, D.C.: National Academies Press, 2018. doi: https://doi.org/10.17226/24935.

[4] B. Ravaei, K. Rahimizadeh, and A. Dehghani, "Intelligent and hierarchical message delivery mechanism in vehicular delay tolerant networks," *Telecommunication Systems*, vol. 78, May 2021, doi: https://doi.org/10.1007/s11235-021-00801-1.

[5] C. Chakrabarti and Samir Pramanick, "Implementing Data Security in Delay Tolerant Network in Post-disaster Management," *Lecture notes in electrical engineering*, pp. 77–92, Oct. 2021, doi: https://doi.org/10.1007/978-981-16-4035-3_8.

[6] W. Khalid, N. Ahmed, S. Khan, Z. Ullah, and Y. Javed, "Simulative Survey of Flooding Attacks in Intermittently Connected Vehicular Delay Tolerant Networks," *IEEE Access*, vol. 11, pp. 75628–75656, Jan. 2023, doi: https://doi.org/10.1109/access.2023.3297439.

[7] M. Matracia, N. Saeed, M. A. Kishk, and M.-S. Alouini, "Post-Disaster Communications: Enabling Technologies, Architectures, and Open Challenges," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1177–1205, 2022, doi: https://doi.org/10.1109/ojcoms.2022.3192040.

[8] A. Agussalim, I. Gede, E. Adi Saputro, M. Idhom, A. N. Sihananto, and S. Mukarromah, "Exploring Store-Carry-forward Networking on Emergency Alert Dissemination of Jakarta's Integrated Rail System," pp. 363–368, Nov. 2024, doi: https://doi.org/10.1109/itis64716.2024.10845704.

[9] S. Singh and A. Verma, "Buffer Management Techniques in Delay Tolerant Networks: A Comprehensive Survey," *Recent Advances in Computer Science and Communications*, vol. 16, no. 8, Oct. 2023, doi: https://doi.org/10.2174/2666255816666230817101831.

[10] O. Ur Rehman, I. Ahmed Abbasi, H. Hashem, K. Saeed, M. Faran 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, pp. 1–12, Jun. 2021, doi: https://doi.org/10.1155/2021/9773402.

[11] S. Khan, K. Saeed, M. Faran Majeed, S. A. AlQahtani, K. Aurangzeb, and M. Shahid Anwar, "NPBMT: A Novel and Proficient Buffer Management Technique for Internet of Vehicle-Based DTNs," *Computers, materials & continua/Computers, materials & continua (Print)*, vol. 77, no. 1, pp. 1303–1323, Jan. 2023, doi: https://doi.org/10.32604/cmc.2023.039697.

[12] A. Prasad, S. Gurung, and K. Sharma, "Challenges and opportunities to Enhance Buffer Management for Disaster Area in Delay Tolerant Network: A Review with Performance Analysis," *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)*, pp. 1–9, Aug. 2024, doi: https://doi.org/10.1109/iceect61758.2024.10739286.

[13] S. Rashid and Q. Ayub, "Social Community Buffer Management Policy for Delay Tolerant Network," *Wireless Personal Communications*, vol. 130, no. 3, pp. 2099–2120, Apr. 2023, doi: https://doi.org/10.1007/s11277-023-10373-9.

[14] A. Balasubramanian, B. Neil Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," Aug. 2007, doi: https://doi.org/10.1145/1282380.1282422.

Vol. 6, No. 4, August 2025, Page. 2239-2252
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.5241

[15] Agussalim, A. Pratama, E. M. Safitri, N. C. Wibowo, and A. B. Putra, "Message Time-to-live based Drop Policy Under Spray and Hop Distance Routing," *IEEE Xplore*, Oct. 01, 2020. https://ieeexplore.ieee.org/abstract/document/9321110 (accessed Sep. 13, 2023).

[16] 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, Jan. 2023, doi: https://doi.org/10.3390/s23020922.

[17] S. Iqbal, I. A. Choudhry, and A. M. Lodhi, "Intelligent Buffer Management Policy in Post Disaster Network Using DTN," *2023 25th International Multitopic Conference (INMIC)*, pp. 1–8, Nov. 2023, doi: https://doi.org/10.1109/inmic60434.2023.10465769.

[18] E. Rosas, O. Andrade, and N. Hidalgo, "Effective communication for message prioritization in DTN for disaster scenarios," *Peer-to-Peer Networking and Applications*, Dec. 2022, doi: https://doi.org/10.1007/s12083-022-01416-2.

[19] M. Gurmessa, Y. Negesa, and S. Nagarajan, "Comparative Analysis of Simbet with Ttl and Simbet with Social Based Message Dropping Policy Routing Protocol," *SSRN Electronic Journal*, Jan. 2023, doi: https://doi.org/10.2139/ssrn.4385650.

[20] Y. Harrati and A. Abdali, "MaxHopCount: A New Drop Policy to Optimize Messages Delivery Rate in Delay Tolerant Networks," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 1, pp. 37–37, Jan. 2016, doi: https://doi.org/10.9781/ijimai.2016.417.

[21] Agussalim and M. Tsuru, "Spray Router with Node Location Dependent Remaining-TTL Message Scheduling in DTNs," *Journal of information processing*, Jan. 2016, doi: https://doi.org/10.2197/ipsjjip.24.647.

[22] F. Rashidjafari, N. Derakhshanfard, B. Shahrokhzadeh, and A. Ghaffari, "Selfish Node Detection in Delay Tolerant Networks Using Fuzzy Logic," *Fuzzy Information and Engineering*, vol. 16, no. 4, pp. 285–299, Dec. 2024, doi: https://doi.org/10.26599/fie.2024.9270046.

[23] N. Sonkar, S. Darshan SL, and S. Acharekar, "Enhancing Multihop Routing in DTNs via Fuzzy Logic-Driven Relay Selection Mechanisms," pp. 1–6, Apr. 2025, doi: https://doi.org/10.1109/otcon65728.2025.11070422.

[24] J. Wu, F. Yuan, Y. Guo, H. Zhou, and L. Liu, "A Fuzzy-Logic-Based Double *Q*-Learning Routing in Delay-Tolerant Networks," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, Jan. 2021, doi: https://doi.org/10.1155/2021/8890772.

[25] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," *Proceedings of the Second International ICST Conference on Simulation Tools and Techniques*, 2009, doi: https://doi.org/10.4108/icst.simutools2009.5674.

[26] P. Garg, H. Kumar, R. Johari, P. Gupta, and R. Bhatia, "Enhanced Epidemic Routing Protocol in Delay Tolerant Networks," pp. 396–401, Feb. 2018, doi: https://doi.org/10.1109/spin.2018.8474132.