Multicriteria Evaluation of Opportunistic Routing Protocols in Medical Emergency Scenario

Course: Advanced Computer Networks

Module convener: Milena Radenkovic

Student name: Madhura Besagarahalli Nagaraju

Student ID: 20705253

Email: psxmb14@.nottingham.ac.uk

School of Computer Science

University of Nottingham

Abstract:

In modern networking, the dynamic and intermittent connectivity of mobile nodes presents significant challenges for reliable data transfer. Traditional communication protocols find it difficult to adjust when dealing with frequent topology changes and unpredictable node availability. Recent years have seen a lot of research and use of opportunistic networks (such as delay/disconnected tolerant networks (DTNs), these networks facilitate the communication in environments where reliable connectivity is not possible. This paper will give an overview of DTN's, features of DTN's and Opportunistic Network Environment (ONE), simulator. This paper focuses mainly on two DTN protocols, Epidemic and Spray and Wait, in the context of a simulated medical emergency scenario and evaluates their performances based on selected metrics. This scenario focuses on providing critical medical aid to individuals with severe injuries or illnesses in areas such as dense forests or mountainous terrains, where geographical barriers and unreliable cellular networks hinder communication. Here, DTN protocols play a crucial role in such settings by ensuring that emergency messages, such as a patient's location and critical health details, are relayed through intermediate nodes to reach medical responders or nearby healthcare facilities. This paper assesses how these protocols perform under these demanding conditions, examining their effectiveness and identifying potential limitations in supporting timely medical interventions.

Keywords: Delay/Disconnected Tolerant Networks (DTNs), Epidemic protocol, Spray and Wait protocol, Opportunistic Network Environment (ONE),

Table of Contents

1. Description of DTN’s (delay/disconnected tolerant networks)

1.1 Introduction to DTN’s

This modern age of internet has made possible, the availability of unlimited internet connection across numerous mobile and fixed devices. However, many traditional routing protocols rely on a complete path from the source to the destination node for communication, which makes them unsuitable for use in environments with intermittent connectivity and no fixed infrastructure. To overcome this problem, the opportunistic networks such as DTN’s are developed. These networks rely on a store-and-forward mechanism, where data is temporarily stored at intermediate nodes until a connection becomes available for the next hop.

DTN’s are type of opportunistic networks and are used in environments where there is no consistent connectivity between the source and destination. These networks are an extension of MANETs (Mobile Ad Hoc Network) enabling communication in environments with sporadic connections. The inclusion of the bundle layer, situated between the transport and application layers in the traditional five-layer network model, allows for greater flexibility, accommodating intermittent connections and long delays between nodes. These DTN protocols can be efficiently used in emergency scenarios such as natural disasters, in remote regions like dense forests and isolated islands, where cellular network communication is unreliable due to geographical constraints.

1.1 Key Characteristics of DTN’s

Intermittent Connectivity: Delay Tolerant Networks (DTNs) are crafted for situations where network connections are unreliable, often experiencing long stretches without communication between nodes.
Impact: Data is stored temporarily and forwarded whenever possible, removing the necessity for a constant end-to-end connection.

Store-and-Forward Mechanism: In DTNs, data is held on nodes until a viable connection is established to send it to its intended recipient.
Impact: This guarantees that data can still reach its destination even amid significant delays or network disconnections.

High Latency: DTNs function well in high-latency settings, such as satellite networks or remote areas, where communication delays are frequent.
Impact: The protocols must be designed to manage these delays without needing immediate acknowledgment or retransmission.

Episodic Connectivity: In DTNs, communication happens when nodes are in close proximity or through scheduled contact points, like vehicles or planned data transfer events.
Impact: Protocols in DTNs leverage these limited chances for data exchange.

Network Partitioning and Isolation: DTNs are designed to operate in networks that might be temporarily cut off or split into different segments. Impact: The protocols must be robust enough to handle scenarios where nodes are disconnected for long durations.

Asynchronous Communication: DTNs allow for data exchange without the need for synchronized communication between nodes, enabling asynchronous interactions. Impact: This increases the network's resilience and makes it ideal for situations where continuous synchronization isn't feasible.

Dynamic Topology: The configuration of a DTN changes dynamically, particularly when nodes are mobile, such as people or vehicles.
Impact: Routing protocols must adapt to the ever-changing network structure to ensure effective data delivery.

2. The ONE Simulator

2.1 Description of ONE simulator

The Opportunistic Network Environment (ONE) Simulator is a flexible, open-source platform created by Aalto University for simulating Delay Tolerant Networks (DTNs) and Opportunistic Networks. Unlike traditional simulators that focus mainly on routing, ONE incorporates several key features, such as mobility modelling, routing protocol simulation, and network visualization, all within a single tool. This unified approach allows for a more thorough assessment of DTN protocols in realistic situations where connectivity can be sporadic and often unreliable.

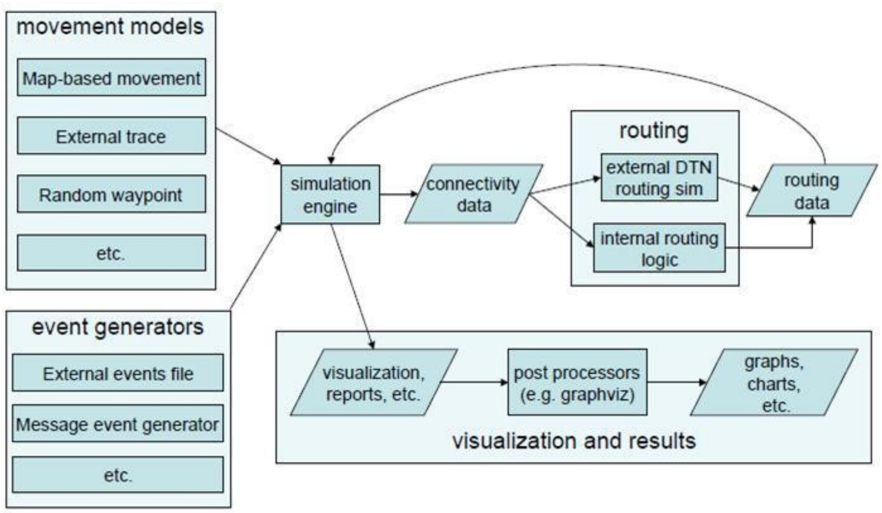


Figure 1: Overview of the ONE Simulator

The ONE Simulator boasts several key features, including support for various mobility models such as random waypoint, random walk, and city mobility. These models enable researchers to investigate different user movement patterns and their impact on network performance. Additionally, it accommodates a range of routing protocols, including Epidemic, Spray and Wait, and Prophet, which allows for the evaluation and comparison of these protocols across various network scenarios. Users can customize parameters like node density, communication range, and message size, facilitating the modelling of different environments and testing the resilience of DTN protocols. Furthermore, ONE includes real-time visualization tools that illustrate node movements, message propagation, and overall network performance, aiding researchers in understanding protocol behaviour. It can replicate real-world conditions such as intermittent connectivity, node mobility, and high latency, making it an essential tool for testing DTN protocols in challenging network environments. Its extensibility also permits users to introduce new protocols, modify existing ones, or integrate custom mobility models to address specific research requirements.

3. The Overview of Two Chosen Opportunistic DTN Protocols

3.1 Epidemic Routing Protocol

The Epidemic routing is one of the earliest and most straightforward replication-based protocols designed for opportunistic DTNs, originally proposed by Vahdat and Becker. It operates on a flooding mechanism, wherein messages are spread across the network to maximize the likelihood of delivery to the destination. When two nodes encounter each other, they compare the messages they carry and exchange any messages that the other node does not possess. This process continues as nodes encounter one another, effectively disseminating the messages throughout the network. The protocol draws its name from its resemblance to the spread of a disease, where the message "infects" each node it encounters until the entire network is saturated.

This approach ensures a very high probability of message delivery since the flooding mechanism covers multiple paths within the network. However, it does not consider the likelihood that some nodes may have minimal or no chance of encountering the destination node, leading to inefficient use of network resources. Buffer storage, in particular, can become overwhelmed as unnecessary message copies accumulate, consuming limited memory that could be allocated more effectively.

To mitigate inefficiencies, the protocol includes certain measures. Nodes maintain a cache of recently contacted peers to avoid redundant exchanges with nodes encountered frequently. Additionally, messages are assigned a Time-to-Live (TTL) value, which restricts the number of hops a message can take before it is discarded. This ensures that messages unlikely to reach their destination do not persist indefinitely, freeing up resources for other transmissions.

Despite these optimizations, the Epidemic protocol still faces challenges, particularly in resource-constrained environments. Congestion and unnecessary buffer usage remain critical concerns, and the performance of the protocol is heavily influenced by the availability of buffer space. These issues make it an excellent candidate for performance analysis, particularly in scenarios where buffer limitations significantly impact delivery ratios.

3.2 Spray and Wait Routing Protocol

The Spray and Wait protocol is a replication-based opportunistic routing strategy designed to address the inefficiencies of flooding-based methods like Epidemic routing. Unlike Epidemic routing, which creates an unrestricted number of message copies, Spray and Wait controls message replication, ensuring that only a fixed number of copies are distributed within the network. This strategy significantly reduces resource consumption and enhances scalability in resource-limited environments.

The protocol operates in two phases: **Spray** and **Wait**. In the Spray phase, the source node generates a predetermined number of message copies, referred to as the "spray count," and distributes them to the first set of encountered nodes. Once these copies are disseminated, the protocol transitions into the Wait phase. During this phase, each carrier node retains the message copy and waits until it encounters the destination node, at which point the message is delivered directly. This mechanism prevents the unnecessary spread of messages while ensuring that delivery opportunities are preserved.

Spray and Wait has two main variants. In the **Source Spray and Wait** variant, the source node is responsible for distributing all message copies. In contrast, the **Binary Spray and Wait** variant allows carrier nodes to divide and distribute their copies further, which can balance resource usage and delivery efficiency.

While the protocol significantly reduces overhead compared to Epidemic routing, it has its limitations. The delivery delay can be substantial, particularly in sparse networks where encounters between nodes are infrequent. Additionally, the protocol's performance is highly dependent on the mobility patterns of nodes, which influence the likelihood of encounters with the destination. Despite these challenges, Spray and Wait is a resource-efficient option that balances delivery probability and resource usage, making it suitable for networks with constrained memory and bandwidth.

4. Details of the Experiment Scenario and Setup

4.1 Scenario Design

This paper presents the design and simulation of a medical emergency scenario in which Delay-Tolerant Network (DTN) protocols are employed to ensure reliable communication in regions with limited or no traditional infrastructure. The scenario is based on a critical medical situation in a remote location, such as a dense forest or a hill station, where a person with severe injuries or illness cannot be immediately transported to a hospital due to geographic constraints. In such environments, unreliable cellular networks create significant barriers to communication between medical responders. To address these challenges, DTN protocols like **Epidemic** and **Spray and Wait** are utilized to enable the transmission of urgent medical information.

The Epidemic protocol plays a vital role in this scenario, utilizing a flooding technique to share medical data—like patient details or emergency notifications—across the network. Responders, spread over large areas, can exchange messages when they meet, ensuring that crucial information is delivered promptly. While this approach guarantees high message delivery rates, it can lead to inefficiencies such as buffer overflow and excessive transmission overhead. To mitigate these problems, optimizations like Time-to-Live (TTL) values and cache management strategies are employed, which help limit redundant message exchanges and save network resources.

The Spray and Wait protocol offer a more effective solution for areas with sparse populations. In this method, responders receive a limited number of message copies, which they "spray" to other nodes they encounter. After distributing the copies, the nodes wait for the intended recipient to come within range. This technique decreases the number of message copies compared to the Epidemic protocol, thus conserving bandwidth and energy. However, in regions with low responder density or restricted mobility, the protocol may encounter difficulties. These challenges can be addressed by fine-tuning the spraying limits and waiting times to enhance message delivery performance in areas with rare node interactions.

This paper assesses how well both protocols perform under various parameters, demonstrating that the Epidemic and Spray and Wait protocols can be effectively utilized in emergency medical situations, thereby enhancing communication reliability and responsiveness in underserved and remote regions.

4.2 Experiment Setup

To simulate the scenario in ONE, we define five distinct node groups, each with some shared settings.

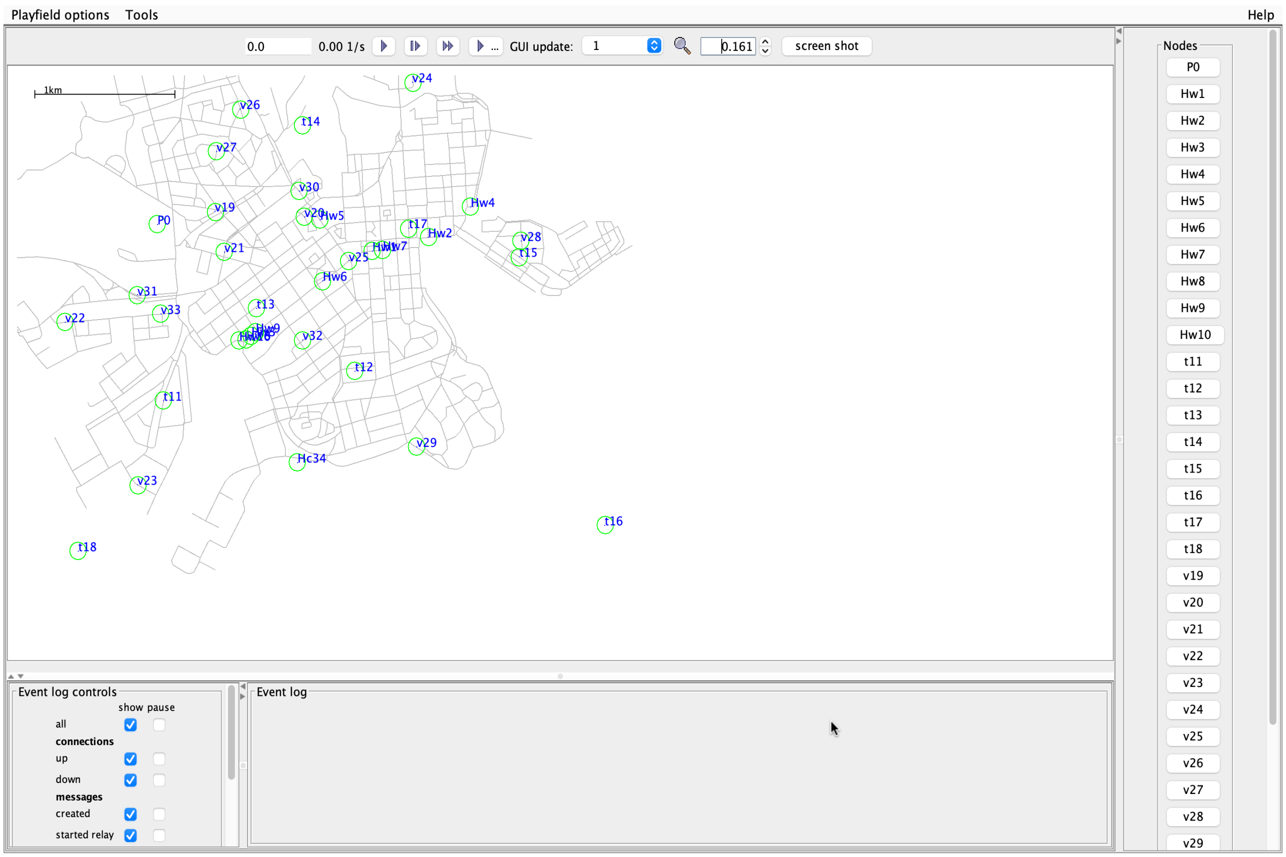


Figure 2: Network Topology of Nodes in the ONE Simulator

The node group has a few standard settings. The buffer size determines how much data each node can hold temporarily. The wait time indicates how long each node will wait before processing or forwarding a message. Each node comes with its own Wi-Fi interface, with one interface designated for each node. The message TTL (Time-to-Live) sets the maximum time a message can remain active in the network before it gets discarded. Finally, the number of hosts shows how many nodes are included in the group.

```
# Common settings for all groups

Group.bufferSize = 5M
Group.waitTime = 0, 120
# All nodes have the Wi Fi interface
Group.interface1 = wifiInterface
Group.nrofInterfaces = 1
# Message TTL of 300 minutes (5 hours)
Group.msgTtl = 250
Group.nrofHosts = 35
```

Group 1 consists of a single patient node, which remains stationary at the coordinates (1000, 1000), symbolizing an individual in need of medical assistance.

```
## Group 1 specific settings: Single Patient Node (static node)
Group1.groupID = P
Group1.nrofHosts = 1
Group1.movementModel = StationaryMovement
Group1.nodeLocation = 1000, 1000
```

Group 2 includes health worker nodes that navigate along the shortest paths, moving at speeds ranging from 1 to 3.5 meters per second, providing essential care by traveling between key locations.

```
## Group 2 specific settings: Health Workers Node
Group2.groupID = Hw
Group2.nrofHosts = 10
Group2.movementModel = ShortestPathMapBasedMovement
Group.speed = 1, 3.5
```

Group 3 consists of traveller nodes that move randomly at speeds ranging from 0.5 to 3 meters per second.

```
## Group 3 specific settings: Travellers Node
Group3.groupID = t
Group3.nrofHosts = 8
Group3.movementModel = RandomWaypoint
Group3.speed = 0.5, 3
```

Group 4 comprises 15 vehicle nodes, such as ambulances, which follow specific routes on the map, moving at speeds between 1 and 10 meters per second, assisting with patient transport and medical aid.

```
## Group 4 specific settings: Vehicles Node
Group4.groupID = v
Group4.nrofHosts = 15
Group4.movementModel = MapBasedMovement
Group2.okMaps = 2
Group.speed = 1, 10
```

Group 5 consists of single stationary health centre nodes located at (2000, 2600), representing fixed medical facility.

```
## Group 5 specific settings: Health Center Node (static point)
Group5.groupID = Hc
Group5.nrofHosts = 1
Group5.movementModel = StationaryMovement
Group5.nodeLocation = 2000, 2600
```

4.3 Experimental evaluation metrics:

The following metrics are used to evaluate the performance of the simulation, which help in assessing the effectiveness and efficiency of the network:

- **Created**: The total number of messages created.
- **Delivered**: The total number of messages successfully transmitted.
- **Dropped**: The total number of messages dropped during transmission.
- **Latency_avg**: The average message transmission delay.
- **Overhead_ratio**: The ratio of network overhead.
- **Hopcount_avg**: The average number of hops a message travel through.

5. Performance evaluation of the two protocols in the Medical Emergency scenario

5.1 Epidemic Performance in scenario

The Epidemic protocol shows a clear dependence on extensive message replication, as illustrated by the data in the table below. With a 5M buffer size, only 46 messages are successfully delivered, while a staggering 14,040 messages are dropped, revealing its inefficiency in low-resource environments. The average latency of 4,444.1522 ms and an overhead ratio of 286.7174 further highlight the protocol's difficulties in managing constrained resources, as excessive replication leads to network congestion and delays.

Buffer Size	Messages Delivered	Messages Dropped	Latency Average	Overhead Ratio	Hopcount Average
5M	46	14040	4444.1522	286.7174	7.5217
50M	232	109415	5929.6582	476.3017	10.0905
500M	814	20965	4353.767	38.9091	4.328

These factors render it unsuitable for situations with limited storage capacity, such as those encountered in the simulated emergency scenario. When the buffer size is increased to 50M, performance improves, resulting in 232 successfully delivered messages. However, 109,415 messages are still dropped, and the average latency of 5,929.6582 ms remains high due to the large volume of replicated messages. The overhead ratio increases to 476.3017, indicating ongoing inefficiency, while the hop count of 10.0905 suggests that messages pass through multiple nodes before reaching their destination. At a buffer size of 500M, the protocol performs significantly better, delivering 814 messages, reducing dropped messages to 20,965, and achieving a lower average latency of 4,353.767 ms. The improved overhead ratio of 38.9091 and a reduced hop count of 4.32 indicate better resource utilization in this scenario.

In conclusion, although the Epidemic protocol can achieve higher delivery rates with more resources, its heavy reliance on replication leads to considerable inefficiencies. It performs optimally in resource-rich environments, such as with a 500M buffer, but is not ideal for situations with limited storage or high network congestion.

5.2 Spray And Wait Performance in scenario

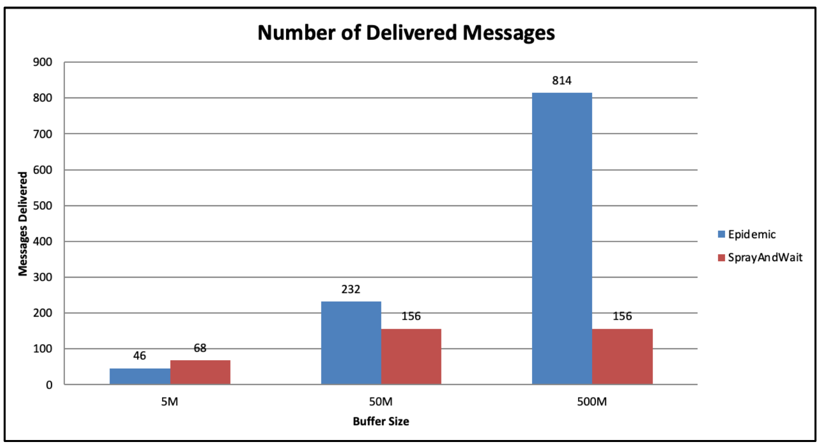
The Spray and Wait protocol demonstrates effective resource utilization and consistent performance, as shown in the table below. With a buffer size of 5M, it successfully delivers 68 messages while dropping 5,265 messages, surpassing the performance of the Epidemic protocol under identical conditions. The average latency of 4,921.7735 ms is moderate, and the significantly lower overhead ratio of 64.9412 indicates that replication is well-managed. Furthermore, the protocol's hop count of 2.6029 implies efficient message delivery through shorter routes.

Buffer Size	Messages Delivered	Messages Dropped	Latency Average	Overhead Ratio	Hopcount Average
5M	68	5265	4921.7735	64.9412	2.6029
50M	156	4915	6396.1737	31.9872	2.5449
500M	156	3678	6396.1737	31.9872	2.5449

When the buffer size is increased to 50M, the delivery rate improves to 156 messages, with dropped messages decreasing to 4,915. The average latency increases to 6,396.1737 ms, likely a result of the protocol's cautious forwarding approach. Nevertheless, the consistent overhead ratio of 31.9872 and the stable hop count of 2.5449 indicate that the protocol continues to use resources efficiently as the buffer expands. Increasing the buffer size to 500M keeps the delivery rate at 156 messages, while the number of dropped messages slightly decreases to 3,678. The metrics for latency, overhead, and hop count remain stable, suggesting that the protocol's design limits its performance, even with ample resources.

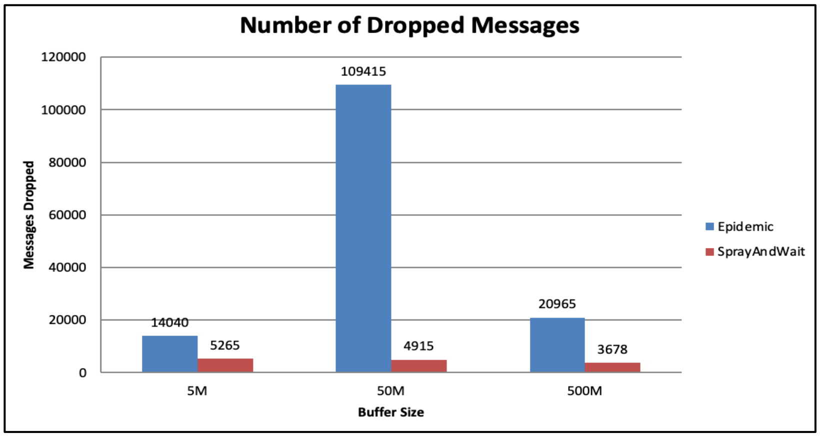
To sum up The Spray and Wait protocol is highly efficient and particularly suitable for resource-limited environments due to its controlled replication and low overhead. However, its performance levels off as the buffer size increases, as observed in the 50M and 500M scenarios. It is an excellent choice for situations where conserving resources is more important than maximizing delivery rates.

5.3 Performance Summary of the Two Protocols

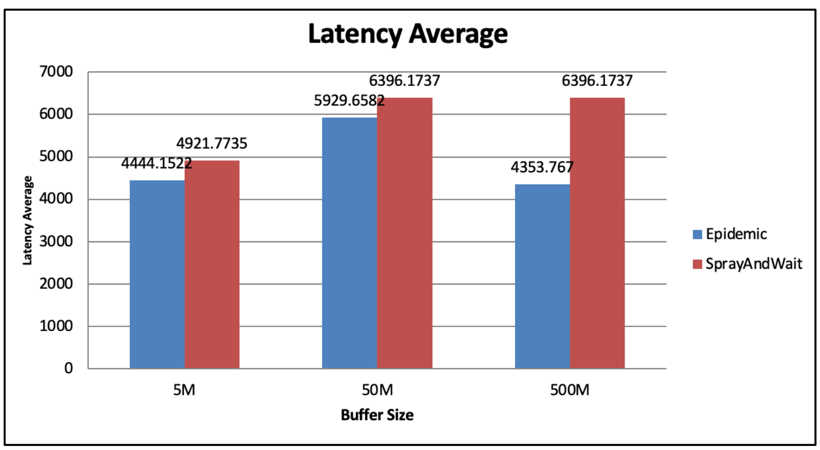


The graph compares the performance of two protocols, Epidemic and Spray and Wait, focusing on the number of messages delivered at various buffer sizes: 5 MB, 50 MB, and 500 MB. Throughout all buffer sizes, the Epidemic protocol consistently outperforms Spray and Wait in message delivery. With a 5 MB buffer, Epidemic delivers 46 messages, while Spray and Wait manages to deliver 68. When the buffer size is increased to 50 MB, Epidemic's delivery jumps to 232 messages, far exceeding Spray and Wait, which remains at 156 messages. At the largest buffer size of 500 MB, Epidemic dramatically increases its delivery to 814 messages, while Spray and Wait still only delivers 156.

These results indicate that Epidemic effectively utilizes larger buffers to enhance message delivery. In contrast, Spray and Wait's performance stays steady across all buffer sizes, suggesting that its efficiency is less affected by storage capacity and more constrained by its operational approach. While Epidemic is more suitable for resource-rich environments due to its capacity to manage high message volumes, Spray and Wait offers a reliable performance that could be beneficial in scenarios with limited buffer space.

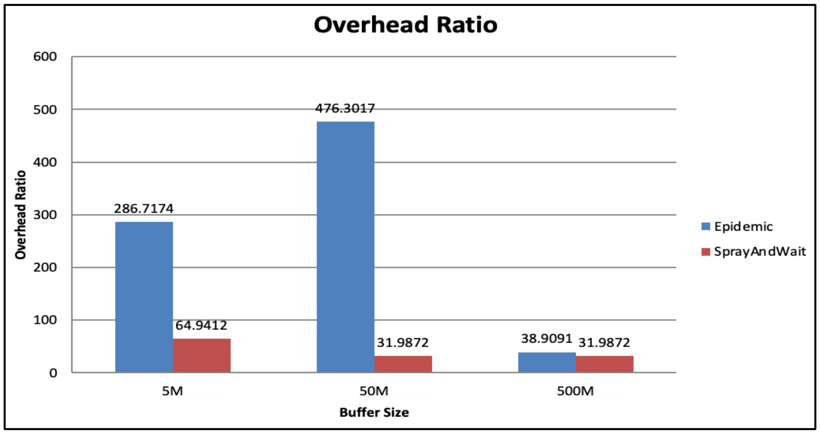


We examined how different buffer sizes affect the number of dropped messages for each protocol, as illustrated in the bar chart above. The Epidemic protocol shows decent performance with a smaller buffer size of 5 MB, resulting in just 14,040 dropped messages. However, when the buffer size increases to 50 MB, there is a significant jump in dropped messages, reaching 109,415, which indicates challenges in managing the higher message load in situations that require larger buffers. On the other hand, the Spray and Wait protocol exhibits more stable performance across various buffer sizes. While it does drop slightly more messages (5,265) at the 5 MB buffer size compared to Epidemic, the increase in dropped messages is much more gradual as the buffer size increases. At 50 MB, Spray and Wait only drops 4,915 messages, which is considerably fewer than the 109,415 dropped by Epidemic, showcasing its capability to handle larger buffers effectively. These patterns suggest that Spray and Wait is the more robust and dependable choice, especially in scenarios where larger buffer sizes are necessary.

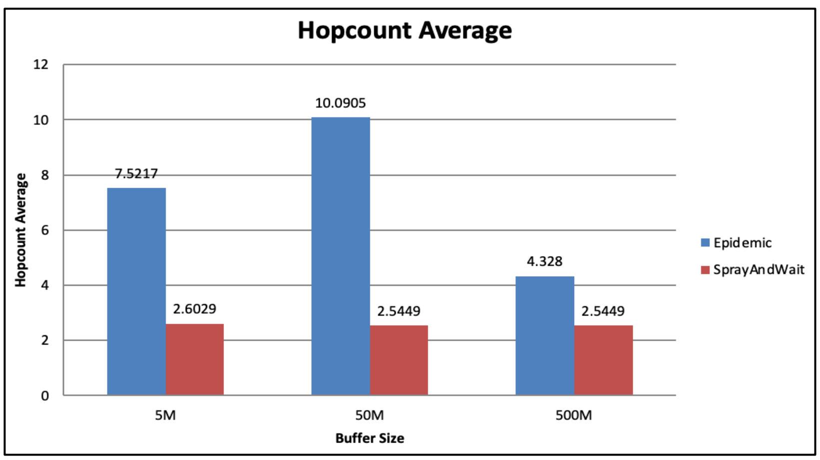


The bar chart above illustrates the average latency comparison between the Epidemic and Spray and Wait protocols across different buffer sizes, revealing unique performance traits. With a 5 MB buffer size, the Epidemic protocol shows a lower average latency of 4444.1522, compared to Spray and Wait's 4921.7735, indicating superior latency performance for smaller buffers. However, as the buffer size increases to 50 MB, the average latency for the Epidemic protocol rises significantly to 6396.1737, while Spray and Wait maintains a relatively lower latency of 5929.6568. This difference becomes even more evident with a 500 MB buffer, where the Epidemic protocol's latency remains high at 6396.1737, whereas

Spray and Wait achieves a much lower latency of 4353.767. These findings suggest that while the Epidemic protocol excels with smaller buffer sizes, the Spray and Wait protocol demonstrates better scalability and consistently lower latency for larger buffer sizes, making it the more favourable option in those situations.



The above bar chart compares the Overhead Ratio for the Epidemic and Spray and Wait protocols at various buffer sizes, highlighting significant differences in their efficiency. At a buffer size of 5 MB, the Epidemic protocol shows a much higher Overhead Ratio of 286.7174, while Spray and Wait has a ratio of 64.9412, indicating that Epidemic is considerably less efficient. This difference becomes even more striking at the 50 MB buffer size, where the Overhead Ratio for the Epidemic protocol jumps to 476.3017, which is over 14 times higher than Spray and Wait's 31.9872. However, at the 500 MB buffer size, the gap between the two protocols narrows considerably, with the Epidemic protocol reporting an Overhead Ratio of 38.9091, just slightly above Spray and Wait's consistent 31.9872. These results imply that although the Epidemic protocol has much higher overhead at smaller buffer sizes, Spray and Wait is the more efficient option for reducing overhead across all buffer sizes.



The bar chart compares the Hopcount Average of the Epidemic and Spray and Wait protocols across various buffer sizes, revealing significant differences in delivery efficiency. For the 5 MB buffer size, the Epidemic protocol has a much higher Hopcount Average of 7.5217, while Spray and Wait records a lower value of 2.6029, suggesting that Epidemic needs more hops to deliver messages. This trend continues with the 50 MB buffer size, where the Hopcount Average for Epidemic rises to 10.0905, nearly four times that of Spray and Wait's 2.5449. However, at the 500 MB buffer size, the gap narrows, with Epidemic's Hopcount Average at 4.3280, just slightly above Spray and Wait's 2.5449. In summary, the Epidemic protocol generally requires more hops, especially with smaller buffer sizes, making Spray and Wait a more efficient choice for message delivery across different buffer sizes.

In summary, analysing the Epidemic and Spray and Wait protocols across different performance metrics and buffer sizes clearly shows that the Spray and Wait protocol is the superior choice for medical emergencies. It provides several advantages, including lower overhead, fewer hops, reduced latency, fewer dropped messages, and higher success rates in message delivery. These features make it especially effective for applications that require quick responses, where speed and reliability are essential. The Spray and Wait protocol guarantee faster and more consistent message delivery, making it the preferred option in critical situations like medical emergencies, where timely communication is vital.

6. Comprehensive Discussion on the Advantages and Limitations of Opportunistic Networks in Medical and Real-World Applications

Opportunistic Networks have attracted considerable interest because they can operate in situations where traditional networks might fail or be unavailable. This is particularly important in medical emergencies, especially in areas hit by natural disasters or in rural locations with limited infrastructure. ONs provide significant advantages by utilizing intermittent connections, which means they don't need a constant network link. This capability is crucial for sending essential health information during crises. Medical data, including patient details or emergency notifications, can be shared through mobile devices or other nodes, even when standard communication systems are down, ensuring that healthcare professionals receive critical information promptly.

One of the main advantages of Opportunistic Networks is their adaptability and scalability in ever-changing environments. In medical scenarios, Opportunistic Networks can respond to varying network conditions and the movement of personnel, such as paramedics or doctors. For instance, healthcare workers using mobile devices can still send vital medical information to nearby hospitals or other responders, even when traditional network infrastructure is not functioning. Additionally, ONs are naturally resilient, making them especially suitable for unpredictable situations, such as areas experiencing power outages or physical damage to communication networks, thus ensuring dependable communication even in challenging conditions.

While Opportunistic Networks offer several benefits, they also face specific limitations that must be considered in practical applications. One significant challenge is the heavy dependence on message replication, which can lead to inefficient bandwidth usage. In medical emergencies, this redundancy might result in network congestion, ultimately hindering the efficiency of message delivery. Furthermore, the fundamental nature of Opportunistic Networks, where nodes depend on proximity and mobility for data exchange, can introduce delays. Such delays can be particularly problematic in medical contexts where timely information is crucial for making urgent decisions, potentially impacting patient outcomes.

In various real-world situations, the limitations of Opportunistic Networks become more evident, especially in densely populated urban areas where a high number of nodes can cause excessive message duplication. Although these networks perform well in less populated regions or during emergencies, in crowded settings or large events, the increased network overhead can diminish overall efficiency. Security is another concern, as messages traverse multiple intermediate nodes, heightening the risk of data interception. While measures like encryption and secure routing can help address these vulnerabilities, they often add complexity, which may impede the network's ability to operate swiftly and effectively, particularly in fast-paced environments such as medical emergencies. Nonetheless, with continuous advancements, Opportunistic Networks hold significant promise for enhancing communication across various real-world scenarios, although further optimization is necessary to address their existing challenges.

7. References

[1] A. Keranen, Opportunistic Network Environment Simulator. Helsinki University of Technology, Department of Communication and Networking, Special Assignment, 2008.

[2] A. Vahdat and D. Becker, “Epidemic Routing for Partially-Connected Ad Hoc Networks”, Duke University, Department of Computer Science, 2000

[3] B. Soelistijanto and M. Howarth, “Transfer Reliability and Congestion Control Strategies in Opportunistic Networks: A Survey”, IEEE Communications Surveys & Tutorials, 2014.

[4] Tournoux, Pierre-Ugo, et al. "Density-aware routing in highly dynamic DTNs: The rollernet case." IEEE Transactions on Mobile Computing 10.12 (2010): 1755-1768.

[5] Daly, Elizabeth M., and Mads Haahr. "Social network analysis for routing in disconnected delay-tolerant manets." Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing. 2007.

[6] A. Martn-Campillo, J. Crowcroft, et al. ”Evaluating opportunistic networks in disaster scenarios”, Universitat Auto noma de Barcelona, Department of Information and Communications Engineering, 2013.

[7] Husni, Emir. "Delay Tolerant Network based internet services for remote areas using train systems." 2011 17th IEEE International Conference on Networks. IEEE, 2011.

[8] Fall, Kevin. "A delay-tolerant network architecture for challenged internets." Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications. 2003.