G54ACN: Evaluating Performance Characteristics of Opportunistic Routing Protocols in ONE

Name:	Adam Walker
Student ID:	4190322
Date of Submission:	12/12/16

1 Opportunistic Networks

Opportunistic networks occur extemporaneously when, often heterogeneous, autonomous and usually mobile nodes of limited communication range form networks as they come into contact with one another through their mobility and/or through fluctuating connections [8]. End-to-end paths between nodes may never exist and it is only by routing messages through a series of hops consisting of other nodes that messages can be successfully routed. As networks form in an ad hoc manner and remain fully decentralised, the topology of an opportunistic network may be constantly changing and it is the complex dynamic topology of opportunistic networks which presents a significant challenge as networks become partitioned and nodes isolated.

In an opportunistic network the nodes themselves represent a computer with wireless networking capability attached to anything, including people, satellites, robots, and animals. The nodes are equipped with computing devices and networking capability and seek to communicate using their available interfaces, typically Bluetooth and/or WiFi. These nodes may face a number of constraints, this could mean limited computational resources, small buffer sizes, and reliance on battery power, amongst others. Often, nodes are attached to sensors of some kind and need to disseminate data they have aggregated but this is not always the case and opportunistic networks are highly flexible. In fact, because of their versatility opportunistic networks are suitable to a vast range of use cases and often prevent the most feasible solution to complex requirements. This is further explored in section 6 which takes an in depth view into the reasoning why opportunistic networks are well suited to specific environments as well as looking at real world implementations.

Traditional networking protocols such as TCP [21] are unsuited to opportunistic networks because of the high latency and frequent lack of an end-to-end connection between nodes. Efforts being made to extend opportunistic networks with interactivity with traditional protocols largely revolve around using 'coverage layer adaptors' and the concept of an intermediary bundle layer sitting atop the transport layer to mediate between protocols [5].

Data transmitted in opportunistic networks is usually termed bundles, as it represents a larger chuck of data than individual packets would in a typical wired network. To disseminate data, nodes within opportunistic networks employ a strategy called 'store and carry forward', which fairly explicitly describes how routing occurs. Using fig. 1 as an example, at time t_0 node S wishes to send a message to node D but there is no direct connection between them. To achieve successful communication with D, S must make use of intermediary node(s) to route the message. At time t_0, S passes its message to node 1, which stores the message in its buffer and begins the carry-forward stage. At time t_1, node 1 has changed position and

forwards the message to node 4. At a later stage - t_2 - node 4 encounters the destination node D and completes the message delivery process.

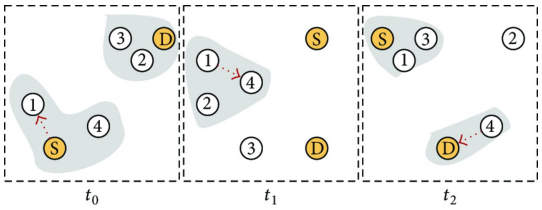


Figure 1: Store and carry forward routing paradigm [11]

In terms of classification there is no significant distinction made between opportunistic networks and Delay/Disruption Tolerant Networks (DTNs) but it is commonly seen that DTNs are a subset, and the most extreme case, of opportunistic networks [2]. It is important to note that Mobile Ad Hoc Networks (MANETs), and thus also Vehicular Ad Hoc Networks (VANETS), are not opportunistic networks despite sharing a number of features such as hop-by-hop connections. The distinction lies in the fact that MANETs often seek ephemeral routes between nodes for end-to-end communication and the routing protocols for MANETs are designed to support this, these protocols cannot handle the extreme cases that DTNs represent.

Opportunistic networks can be augmented via the inclusion of static nodes, e.g. road side units (RSUs) or access points (APs), which are usually connected to infrastructure and more performant. Static nodes can take on a number of roles so as to increase the performance of the network; in DTNs a common use of static nodes is to alleviate congestion when large data bundles are being transmitted. Nodes can recover buffer space for future use by offloading messages to the static points which can then route the messages to passing nodes.

2 ONE

The Opportunistic Network Environment simulator (ONE) [10] is a free software application and package collection for modelling DTNs built using the Java 1.6 programming language and made available under GPLv3. ONE facilitates advanced modelling for layers over, and including, the network layer (i.e. excluding the physical layer [16]).

ONE is suitable for the simulation of opportunistic networks and DTNs as well as less restricted networks such as MANETs [13].

In ONE, nodes, also termed hosts, are autonomous units with computing power, energy and network interfaces; groups of nodes communicate when in range using a designated routing protocol. ONE permits the modification of any of these aspects so that it is possible to model any network

No.	Layer	
7	Application	Layers accessible in ONE
6	Presentation	
5	Session	
4	Transport	
3	Network	
2	Data Link	
1	Physical	

Figure 2: Open Systems Interconnect Model

interface and thus through configuration devise scenarios consisting of varying types of host with different interfaces, energy capacity, buffer sizes, and transmission ranges. In order to make use of the interfaces ONE allows the specification of network events, which dictate to the simulator which nodes should create network traffic as well as how and when the messages should be disseminated. It is also possible to use real traffic to accurately design representative scenarios. Even by fine tuning the artificial network traffic, realistic traffic of any desired type can be modelled for later evaluation.

In terms of the network topology, ONE makes available an extensive framework for node mobility, including support for real world traces [17], some of which are available to researchers through the CRAWDAD collection of data-sets [6]. In ONE, movement and communication rules can be imported from other applications which generate mobility models from map data. In this way, it is possible to develop complex hybrid mobility models using a combination of real world traces and map rules to define groups of nodes which move highly realistically for any desired type of node (e.g. trams, busses, pedestrians etc.). Furthermore, the versatility of ONE's mobility support means that nodes can transition between types to represent pedestrians using public transport, for instance.

To make use of ONE it is required to devise config files in the requisite format. ONE's config files allow the user to create completely custom scenarios including setting applications, mobility, routing protocols, interfaces, and network events.

To go beyond what is already provided within ONE's packages, such as to add an unimplemented routing protocol, it is necessary to modify the source code itself and provide any implementations required by the desired scenario. As ONE conforms to the standards expected of a typical Java project, an integrated development environment can be employed to assist with programming and debugging. Extending the software itself is largely expected and the software architecture strongly supports this; many convenience classes are provided to assist in easing development of extensions.

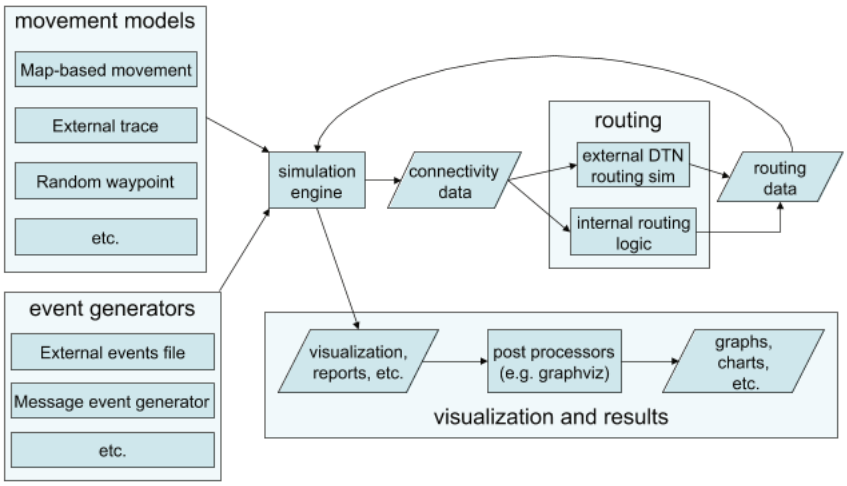


Figure 3: ONE Architecture [10]

```
# GUI underlay image settings
GUI.UnderlayImage.fileName = data/helsinki_underlay.png
# Image offset in pixels (x, y)
GUI.UnderlayImage.offset = 64, 20
# Scaling factor for the image
GUI.UnderlayImage.scale = 4.75
```

Figure 4: Snippet of a one configuration file.

Gathering data from scenarios is the most crucial part of working with a network simulator and ONE, by default, supports the gathering of comprehensive reports. There is a dedicated package containing classes for data aggregation and third-party report classes have also been made available. Any number of reports can be gathered and are used by naming them in the aforementioned config file.

ONE provides a graphical user interface for live visualisation of events, this includes nodes' movement on the chosen map as well as interaction between nodes. The GUI makes initial debugging of modified source code easier but slows down the run-time of simulations and provides no additional data than that which can be gleaned from the reports. In fact, despite support for graphical monitoring of simulations, ONE is most efficiently run in batch mode from the command line where there is additional support for running scenarios sequentially and collating reports for each run of the scenario.

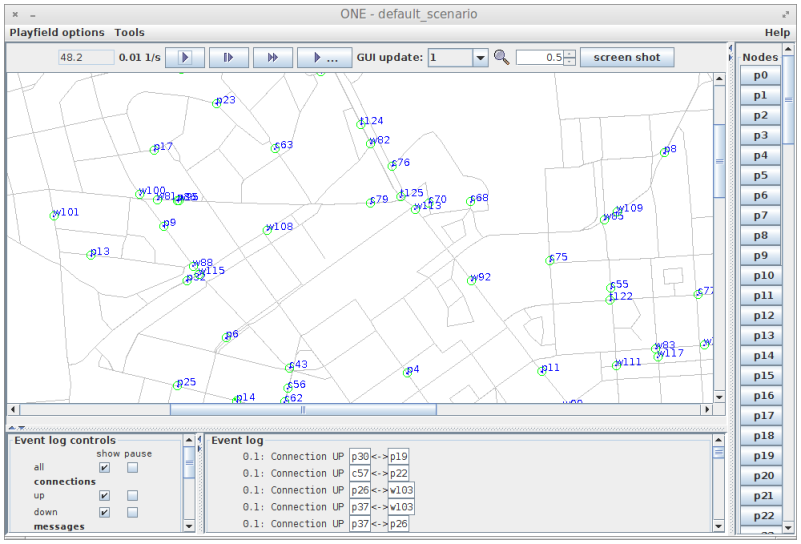


Figure 5: ONE GUI

3 Opportunistic DTN Protocols

3.1 Epidemic

Epidemic [25] is a replication based DTN routing protocol and is the amongst the least intelligent of all DTN routing schemes. Nodes using the Epidemic routing protocol replicate messages they receive and pass these replications to any nodes they happen to come into contact with who do not already have it. This means that nodes encountering one another in a DTN using Epidemic exchange with one another the contents of their buffer which the other node does not have. Epidemic places no upper limit on the number of copies of a message that can exist in the network at any time and thus effectively uses flooding to disseminate data.

Because it indiscriminately replicates packets, messages distributed using Epidemic have a high chance of reaching the destination node and should reach the destination with minimal possible delay. In resource unconstrained environments Epidemic is seen as the best routing protocol but deployed DTNs have practical constraints on resources [28]. Epidemic can consume a a large amount of resources as nodes are not informed if the message has been delivered meaning messages can will be needlessly flooded into partitions until the message expires, nodes in these partitions may have no chance of encountering the destination at all. Efforts to mitigate this have included sending 'vaccine' messages (EVS) to notify that a message can be discarded [29].

3.2 Spray & Wait

Spray & Wait [24] is a two-phase replication based routing protocol that works by restricting the number of times a message can be replicated during the Spray phase. In this way, Spray & Wait seeks to maintain a high probability of delivery whilst being considerate of network resources and has been shown to improve upon these when compared with other protocols [24]. There are two key variants of Spray & Wait, Binary and Vanilla; both of these are looked at in this investigation.

During the Spray phase nodes replicate and disseminate messages based on a system that aims to restrict the maximum number of copies of a message that can exist in the network, L. In Binary Spray & Wait, the source node disseminates $L - 1$ copies of its message to neighbouring nodes on a first-come first-serve basis, as and when it encounters them; the $L - 1$ nodes and the source node then each have a single copy of the message and they all enter the Wait phase. Conversely, in the Spray phase for the Binary version, the source node, and each subsequent node, passes $\frac{N}{2}$ messages to nodes they encounter - leaving the node with $\frac{N}{2}$ messages remaining - where N is the number of messages a given node is carrying. The source node initially has $N = L$ messages. As with the Vanilla version, nodes enter the Wait phase when they have 1 message remaining. As can be seen in fig. 6 the Binary version of the protocol has an advantage over the Vanilla version in that messages can be propagated further from the source node at a faster rate.

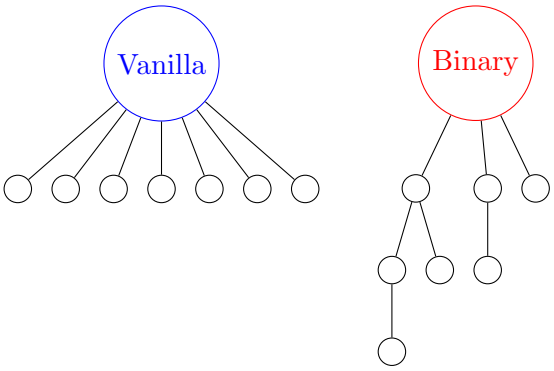


Figure 6: Propagation of messages in Spray & Wait when the upper limit is set to 8. Each node is now in the Wait phase and is therefore carrying 1 message.

The Wait phase of the protocol is identical for both Binary and Vanilla versions of the protocol and begins once a node has only one copy of the message remaining. It keeps this message until it either encounters the destination node or the (time-to-live) TTL of the message expires.

4 Experiment

4.1 Problem Description

4.1.1 Problem Background

Blackhole attacks are a network layer attack in which malicious nodes seek to destabilise the network and prevent dissemination of data. Blackhole attacks can be divided into three categories, those being common black holes, passive black holes and active black holes [27]; the types are distinguished by how the malicious nodes interact with the routing protocol.

Common blackhole nodes are the simplest of the types and indiscriminately discard all data they receive; they may not represent an attack per se but could be indicative of nodes malfunctioning. Active and passive blackholes interact with the routing protocol to either disguise their behaviour or increase the damage they cause. For example, in a MANET routing protocol, such as Ad Hoc On Demand Distance Vector Routing (AODV) [20], a malicious active blackhole node can exploit the routing protocol itself in order to maximise it's impact intercepting and dropping packets. As such routing protocols are unsuited to DTNs because of the expected increase in disconnections and delays, DTN protocols choose not to focus on establishing paths and instead seek to make each node an intelligent distributor of bundles. DTNs are therefore largely invulnerable to active and passive blackhole attacks. Of the categories of blackhole outlined, this leaves DTNs only vulnerable to common blackholes which are the focus of this investigation.

Emergency messages are small bundles of data which contain only crucial information related to an event and are disseminated to both alert and inform of a critical situation. The dissemination of emergency messages is a key use case for terrestrial DTNs and so we choose to focus on their use in Vehicular DTNs (VDTNs) because of recent research and published standards exploring their uses [9]. VDTNs are largely unconstrained by computation and energy and as vehicles are constrained to roads and speed restrictions they have predictable mobility.

4.1.2 Problem

We seek to explore the impact of a blackhole attack on temporal emergency messages being sent between nodes in an urban VDTN during normal traffic, thus envisioning a situation where bad actors propagate malicious nodes so as to prevent the dissemination of emergency messages.

4.2 Scenarios

Vehicles in all scenarios conform to the 802.11p [9] standard which has a transmission range of 300 - 1000 m. To reflect the usage of 802.11p, in all scenarios the vehicles are configured to conservative levels, with a 250 m transmission range and a consistent 10 Mbit/s data rate. The limited transmission range is to account for the higher levels of signal attenuation expected in urban areas, such as those presented. Given that the computational limitations of computers mounted in vehicles are less strict than with typical mobile nodes, we allocate a buffer size of 25 MB. Emergency messages such as those used by WAVE are of small size, to best reflect that data transmitted in the network consists of bundles between 500 B and 250 kB. The TTL of the messages is set to 45 minutes; given that emergency messages are time sensitive we impose a strict TTL to ensure that they do not congest the network any longer than they are needed.

The scenarios are run with the Vanilla Spray & Wait, Binary Spray & Wait and the Epidemic routing schemes. For Spray & Wait, we choose a message replication limit of 10 as it has been shown that only minor reductions in delay can be expected from reducing this much further but at the cost of increased overhead [24].

Blackhole nodes are implemented through the addition of a new `core.DTNHostBlackhole` class and by extending several of the routing classes so that they indiscriminately discard all packets they receive. To assess their impact, blackhole nodes are increased in increments of 10 % and each run of the experiment is repeated 10 times for a total of 100 runs per scenario.

4.2.1 Scenario 1

The first scenario devised makes use of the Helsinki model available in ONE. Sixty vehicles are constricted to navigable roads in the city and move according to the `CarMovement` model, a ONE provided mobility model whereby each node preselects a valid destination and moves towards it using Dijkstra's algorithm. At each turn according to the map, the vehicle changes speed until it ultimately reaches the destination, at which point it remains stationary for a period of time (between 0 and 5 min) before the cycle repeats.

To further the realism of the scenario ONE was extended to support two additional cases. Firstly, nodes entering and exiting the scenario. Throughout the simulation new nodes and existing nodes will enter and depart respectively from the map with probability 0.75 upon each node entering and exiting its stationary period. Secondly, the nodes within the scenario that become stationary for a period of over 2 min are disabled and only reactivated when the node resumes movement. These extensions seek to enhance the scenario devised by more realistically emulating city traversing vehicles in general rather than focusing unintentionally on vehicles which follow a

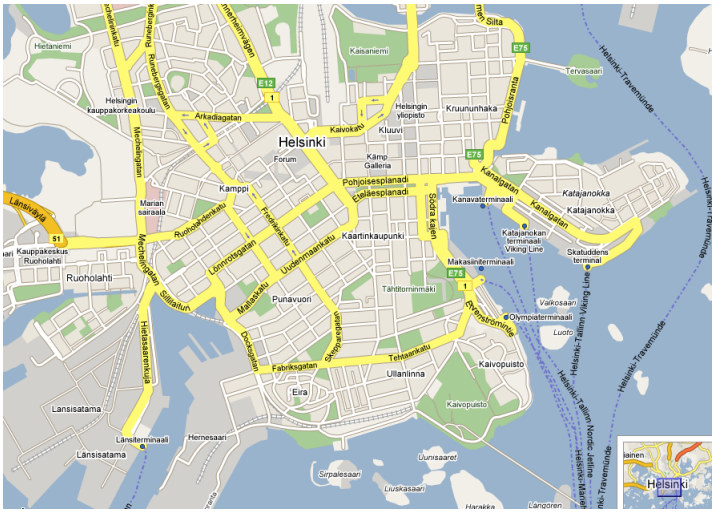


Figure 7: Helsinki map included in ONE

niche and predictable mobility pattern, notably taxis. Whilst some vehicles may persistently move around the area making periodic stops, taxis for example, we would expect that the majority would pass through the area only briefly or make single journeys through the city. Furthermore, vehicles making extended stops would not realistically keep their engine idling for a long time and so we assume that this also disables the networking capability of the node for the duration of the stationary period.

```
if (this.nextTimeToMove > SimClock.getTime() +
    DEFAULT_ENGINE_OFF_TIME) {
    netDisabled = true;
    for (NetworkInterface ni : net) {
        // store old values to be restored later
        netRanges.add(ni.transmitRange);
        ni.transmitRange = 0;
        ni.update();
    }
}
```

Figure 8: Snippet of new `core.CarDTNHost` class responsible for disabling networking for extended periods of stationary time.

The total duration of the simulation is set to 4 h, representing the time between - but discounting - 'rush hour' periods. These are omitted from the simulation because of the starkly different topology we would expect from stagnated traffic moving en masse into and out of the city.

4.2.2 Scenario 2

For the second scenario, a real trace of taxicabs in Rome [4] was used to assess the impact of a blackhole attack. So as to provide useful comparison with the previous scenario and better match the desired mobility of daytime traffic outside of 'rush hour', ten weekdays (excluding weekends) of the overall data-set were selected at random and all movements for active vehicles between 09:00 and 13:00 on the selected days were used to form the traces. As with Scenario 1, the results given in section 5 are based on the mean of these days. A median of 160 vehicles are active at some point during these periods.

Figure 9: Map showing the area covered by the Rome taxi trace.

Making use of the data was an involved process. Once the data had been obtained from CRAWDAD, the CSV file was reformatted with standard shell utilities and then relevant time periods extracted into GPX[1] format using a devised Python script. The BonnMotion [3] tool was then used to parse the GPX data into the custom format readable by BonnMotion and then exported to ONE compatible trace files.

```
5089.0 0 18659.91509363705 15885.76010912304
5090.0 0 18639.080776698465 15878.151538152177
5091.0 0 18618.246459759877 15870.542967181313
```

Figure 10: Segment of a converted `.one` file containing the Rome trace.

5 Performance Evaluation

5.1 Without Blackhole Nodes

At 0 % blackhole nodes Spray & Wait achieves lower delivery ratio than Epidemic in both scenarios but does so with several orders of magnitude less packet drops figs. 19 and 20. The delivery ratio in the denser Helsinki scenario is surprisingly low which could indicate that the impact of the type of mobility modelled in Scenario 1 is underestimated in existing literature. The delivery ratio in Scenario 2 is lower still however this is largely attributable to the shorter TTL of the emergency messages and the sparseness of the trace, which covers a far larger geographical area. Greater analysis should be conducted on new and existing traces to evaluate the effect of the time of day on traffic conditions, notably during daytime, morning 'rush hour', evening 'rush hour' and nighttime. We focused only on a very specific period of time on weekdays and would not expect these results to apply to other periods. Highly mobile nodes with high up-time are important to VDTNs and a key consideration to future deployment of a VDTN in an urban environment would be to make use of taxis and public transport to extend connectivity. A hybrid trace using the Rome data-set would help in exploring this further. An alternative to mobile, high up-time nodes could be to deploy RSUs but reliance on infrastructure presents its own issues. In the case of Spray & Wait, further analysis needs to be conducted on the limit of message replication under the constraints of Scenario 1 to provide better protocol configuration recommendation, the message limit of 10 was perhaps too low.

In figs. 13 and 14 we observe that the median number of hops required for a message to reach its destination are markedly different between the two protocols but largely the same between scenarios, reinforcing the idea that in Epidemic the 20 % boost in delivery ratio arises from routing messages through a large number of hosts, almost 3 times the number of hops used by Spray & Wait.

[1]The GPS Exchange Format is a widely used XML data format for semantically collating GPS data. `http://www.topografix.com/gpx.asp`

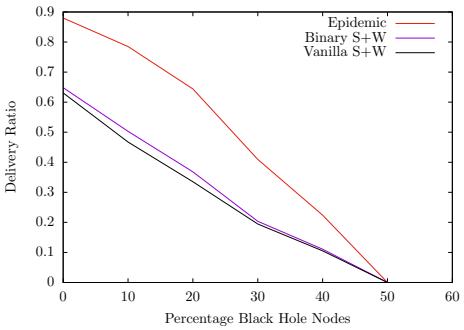


Figure 11: Mean Delivery Ratio at Increasing Percentage Blackhole Nodes for Secenario 1

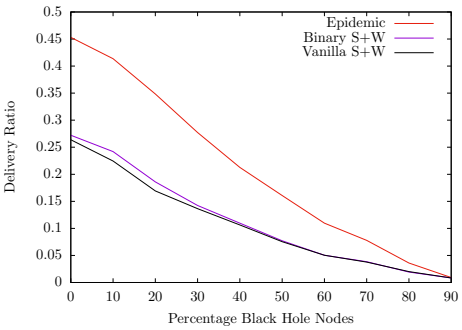


Figure 12: Mean Delivery Ratio at Increasing Percentage Blackhole Nodes for Scenario 2

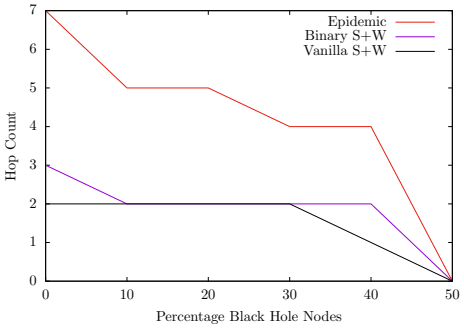


Figure 13: Median Hop Count at Increasing Percentage Blackhole Nodes for Secenario 1

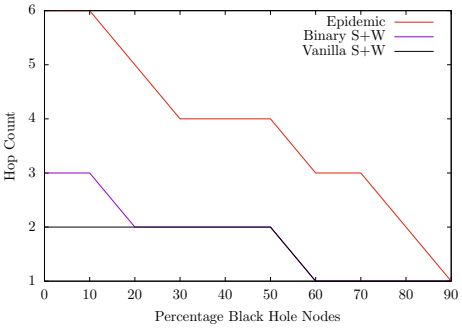


Figure 14: Median Hop Count at Increasing Percentage Blackhole Nodes for Scenario 2

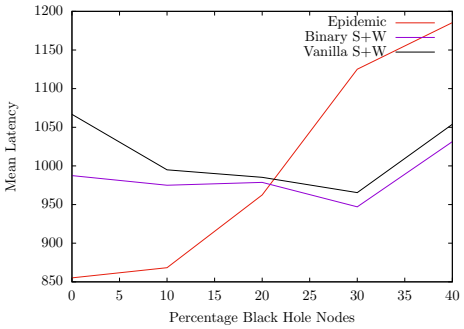


Figure 15: Mean Latency at Increasing Percentage Blackhole Nodes for Secenario 1

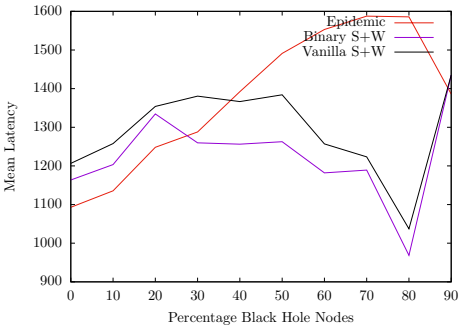


Figure 16: Mean Latency at Increasing Percentage Blackhole Nodes for Scenario 2

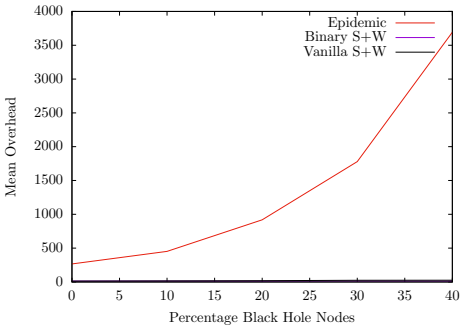


Figure 17: Mean Overhead Ratio at Increasing Percentage Blackhole Nodes for Secenario 1

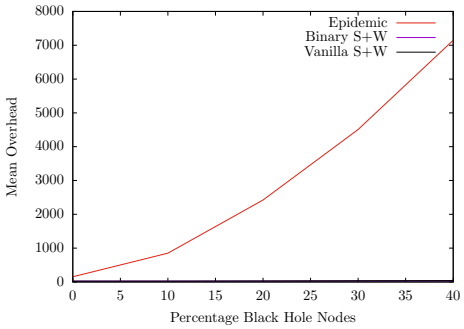


Figure 18: Mean Overhead Ratio at Increasing Percentage Blackhole Nodes for Scenario 2

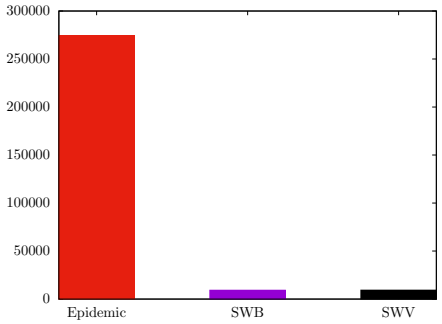


Figure 19: Mean Packets Dropped in Secenario 1 with 0 Blackhole Nodes

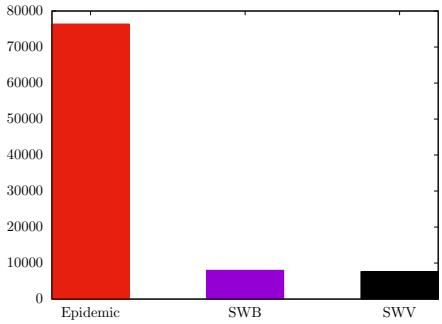


Figure 20: Mean Packets Dropped in Secenario 2 with 0 Blackhole Nodes

$$\frac{total\ messages\ relayed - total\ messages\ delivered}{total\ messages\ delivered}$$

Figure 21: Formula used in the calculation of the overhead ratio in figs. 17 and 18.

5.2 With Blackhole Nodes

In both scenarios the increase in blackhole nodes corresponds to a decrease in the delivery ratio of the messages. This is to be expected of both protocols as the blackhole attack restricts the dissemination of messages however the effect is exacerbated in Spray & Wait in both scenarios. As shown in figs. 11 and 12, the impact of blackhole on delivery ratio nodes varies between the scenarios. The denser Helsinki model performs substantially worse under blackhole attack than the Rome data-set. In Scenario 2 packets are still successfully delivered until over 90 % of nodes are participating in the blackhole attack whereas in Scenario 1 the network cannot handle 50 % of its nodes being blackholes.

It could be thought that under Binary Spray & Wait blackhole nodes have the potential to cause more severe network disruption than in the Vanilla scheme because blackhole nodes that receive messages could receive and discard a large portion of the replicated messages. Despite this, in the scenarios assessed we see little difference between the two versions in terms of message delivery except that Vanilla Spray & Wait under a blackhole attack performs marginally worse at lower percentages of blackhole nodes, indicating that propagation at distance is a better tactic than disseminating to neighbours.

As expected, in both scenarios the number of hops attainable between source and destination decreases as nodes are unable to communicate at distance due to the blackhole nodes and the number of possible hops reduces (figs. 13 and 14). Corresponding with this, the time required for delivery of data extends as fewer useful hops are made (figs. 13 and 14). This is again because mobility is the main reason for successful message at high percentages of blackhole nodes rather than message passing.

As evidenced in figs. 11 and 12, the Epidemic protocol achieves a notably better delivery ratio in both scenarios tested, suggesting that a naive approach to counteracting blackhole nodes in a VDTN would be to increase the message replication limit. It is important to consider however that in section 4.2.1 the overhead of Epidemic increases from 10 to 140 times that of Spray & Wait when the percentage of blackhole nodes increases to 40 % (fig. 17), a figure that is worse in section 4.2.2 when the overhead ratio of Epidemic is almost 300 times higher than Spray & Wait (fig. 18). This is partially due to the increase in the number of nodes; as Epidemic places no limits on the number of times a message can be replicated, with more nodes in the Rome data-set (over double) than in the Helsinki model there are more messages to be sent. This could indicate that Spray & Wait scales better but consideration should be given to the replication limit and investigation conducted to determine whether it would need to be raised. As observed in fig. 11, if commuters are considered in addition to taxis the replication limit could well be too low.

Returning to the idea that increasing replication could help alleviate a black hole attack it is important to note that the evidence strongly suggests that the performance gain in terms of delivery success is not worth the substantial rise in overhead Epidemic requires unless it can be guaranteed at the time of network deployment that the large overheads can be handled. In the scenarios presented, for small emergency messages over a high capacity network, the trade off may be worth it in order to better disseminate data but even in the VDTN scenario selected, if the messages were larger and more frequent then the overhead would increase substantially, an issue that would only be exacerbated by a longer TTL. All of these conditions could be expected of a non-emergency application and would therefore lead to a congested and unusable, effectively collapsed network. Though it has been suggested that improved buffer management can result in the need for a smaller buffer [29], we would still expect epidemic to demand more buffer space than schemes with message replication limits. In the scenario tested vehicles have enough buffer space to handle the overheads of the emergency messages but in a general purpose DTN the message sizes are often large and hosts are more resource constrained than vehicles. In these situations the overhead of using Epidemic is likely prohibitive.

6 Wider Discussion

Opportunistic networks have been deployed for use in the military, emergency services, space, underwater, in natural disaster and war zone situations, for environmental monitoring, and improving internet coverage amongst countless other scenarios. In this section we select and describe two real-world cases benefiting from opportunistic networks followed by a look at some current issues.

Underwater networks (UWNs) seek to facilitate communication between heterogeneous nodes, including purely autonomous nodes which may need to disseminate sensor data and remotely controlled robots which permit human operators to interact in the environment. Communicating in an UWN, particularly at depth, is extremely difficult due to the steep increase in propagation delay and restricted bandwidth [22]; it is also highly energy expensive, requiring either high powered, low frequency acoustic/electromagnetic waves or focused optical beams [12]. Thus, close range communication is essential and DTN protocols for UWNs focus on utilising the most mobile nodes to ferry data between network partitions [22]. Delay-tolerant Data Dolphin (DDD) [14] for example uses more mobile 'dolphin' nodes to collect and transport data throughout the network. DTNs therefore make feasible communication underwater, where the steep cost of deploying nodes and equipment makes building network infrastructure cost prohibitive. DTNs are essential in UWNs, notably for science and industry applications such

as oil exploration [23]. Although deployed on the surface SeNDT (Sensor Network with Delay Tolerance) presents an example of the usefulness of DTNs for environmental monitoring; SeNDT consists of mobile sensors in a lake in Ireland for monitoring increasing pollution [15]. The Littoral Ocean Observatory Network (LOON) is an example of an UWN with completely submerged heterogeneous nodes that was deployed off the coast of the Palmaria Island, Italy [1].

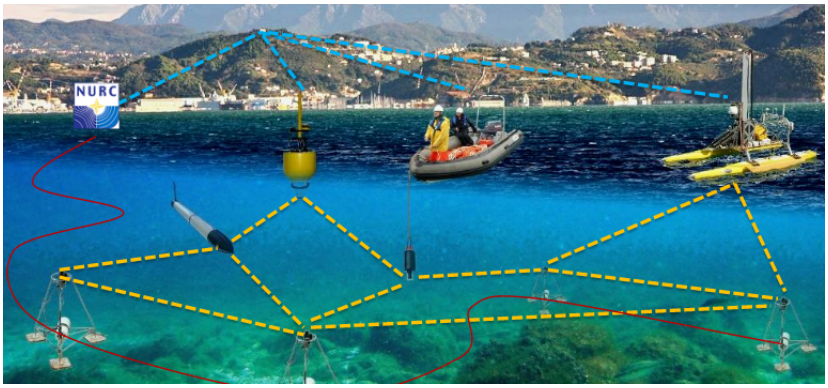


Figure 22: LOON [1]

DTNs are becoming indispensable for providing connectivity in rural and developing regions. DakNet [19] is a DTN deployed in Cambodia and India which was deployed at a substantially lower cost than traditional infrastructure and serves to connect people who could not ordinarily make use of internet based services due to cost and accessibility. Vehicles which routinely travel between villages - particularly busses but also motorcycles and bicycles - are fitted with mobile access points which automatically collect and deposit data from fixed kiosks, when in range. This data is then ferried between access points and ultimately synchronised with the wider internet when in range of a connected access point. Other such implementations of DTNs to facilitate internet access include Bytewalla [18], a secure Android smartphone tool supporting email and health-care utilities. The ad hoc and 'store carry forward' paradigm of opportunistic networks makes such deployments possible and scalable at low cost to the benefit of many.

Opportunistic networks are vulnerable to a range of attacks including packet drops, traffic analysis, data spoofing, denial of service, and resource abuse [7]. Resource abuse attacks are a high threat to DTNs because of their simplicity to execute and impact on the network as a whole. As noted, nodes in a DTN can struggle with limited resources, any of these resources can be abused forcing nodes to ultimately be at best useless to the network, at worst participate in further network disruption. Flooding, blackholes, greyholes, wormholes, bundle manipulation, and node impersonation are

challenging vulnerabilities facing future deployment of DTNs, particularly in situations where there is a gain to be had from exploiting these attacks.

DTNs present a unique privacy risk, with concerns largely focusing on data explicitly provided by the routing protocol, such as geolocation data but also on the information that can be gleaned from metadata. Although the privacy risks are relevant for expensive autonomous nodes which could be a target of theft or vandalism, privacy concerns associated with DTNs are particularly important to scenarios where nodes represent people such as pedestrians or cars. Vanet/DTN based on Trend of Delivery (VDTN-TOD) [26] uses vehicle trajectory to better route packets and demonstrates improved packet delays compared to less intelligent schemes such as Spray & Wait; to gain this improvement it uses the movement of nodes to predict future locations and encounters. It is clear that such schemes can improve on the quality of service in DTNs but at the clear cost of privacy to nodes. Tracking users movements and being able to predict with high accuracy where they will be at any given time presents a significant privacy risk. Balancing privacy and network quality at the routing protocol is an open problem as even without being given explicit access to GPS coordinates it is possible to build up detailed profiles of node' routines.

The ethics of DTNs present another concern although one that largely is shared by networks at large. An application gathering sensor data on individuals for instance would need to obey strict security protocols to avoid leaking personal data. Legislating and providing standards for companies to follow in development could certainly be accomplished but whether they would be adhered to is another issue entirely, as evidenced by recent attacks on various 'internet of things' devices.

As DTNs have evolved over recent years they have become more stable and reliable. LOON [1] achieved 98 % up-time with its underwater nodes; an impressive figure given the constraints. Such up-time may be suitable for sensor data which requires further aggregation and processing before semantic meaning can be gathered from it but for many uses particularly, service oriented and emergency applications, a loss in connectivity could be unacceptable. An unavailable network in an emergency scenario could well lead to loss of life and so reliability and availability are important considerations to the deployment of DTNs. The ongoing increase in number and sophistication of equipped sensors means that more frequent, larger quantities of data needs to be transmitted between nodes; opportunistic networks need to be scalable, not just at current rates of data.

Ultimately given their numerous advantages, we can expect DTNs to become even more commonplace in the future however further research is needed to mitigate the concerns outlined and explore the issues raised in section 5.

References

[1] J. Alves, J. Potter, P. Guerrini, G. Zappa, and K. LePage. The loon in 2014: Test bed description. In *2014 Underwater Communications and Networking (UComms)*, pages 1–4, Sept 2014.

[2] S. Benedetto, M. Luise, and L.M. Correia. *The Newcom++ Vision Book: Perspectives of Research on Wireless Communications in Europe*. SpringerLink : Bücher. Springer Milan, 2012.

[3] BonnMotion. *BonnMotion: A Mobility Scenario Generation and Analysis Tool*, accessed 30th November, 2016. http://sys.cs.uos.de/bonnmotion/.

[4] Lorenzo Bracciale, Marco Bonola, Pierpaolo Loreti, Giuseppe Bianchi, Raul Amici, and Antonello Rabuffi. CRAWDAD dataset roma/taxi (v. 2014-07-17). Downloaded from http://crawdad.org/roma/taxi/20140717, July 2014.

[5] C. Caini. 2 - delay-tolerant networks (dtns) for satellite communications. In J.J.P.C. Rodrigues, editor, *Advances in Delay-Tolerant Networks (DTNs)*, pages 25 – 47. Woodhead Publishing, Oxford, 2015.

[6] Dartmouth College. *CRAWDAD A Community Resource for Archiving Wireless Data at Dartmouth*, accessed 12th October, 2016. http://crawdad.org/about.html.

[7] S. Domancich. *Security in Delay Tolerant Networks (DTN) for the Android Platform*. Trita-ICT-EX. Skolan för informations- och kommunikationsteknik, Kungliga Tekniska högskolan, 2010.

[8] C. M. Huang, K. c. Lan, and C. Z. Tsai. A survey of opportunistic networks. In *22nd International Conference on Advanced Information Networking and Applications - Workshops (aina workshops 2008)*, pages 1672–1677, March 2008.

[9] D. Jiang and L. Delgrossi. Ieee 802.11p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040, May 2008.

[10] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.

[11] Jiradett Kerdsri and Komwut Wipusitwarakun. Dynamic rendezvous based routing algorithm on sparse opportunistic network environment. *International Journal of Distributed Sensor Networks*, 11(2), 2015.

[12] Liu Lanbo, Zhou Shengli, and Cui Jun-Hong. Prospects and problems of wireless communication for underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8):977–994, 2008.

[13] Mengjuan Liu, Yan Yang, and Zhiguang Qin. *A Survey of Routing Protocols and Simulations in Delay-Tolerant Networks*, pages 243–253. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[14] E. Magistretti, J. Kong, U. Lee, M. Gerla, P. Bellavista, and A. Corradi. A mobile delay-tolerant approach to long-term energy-efficient underwater sensor networking. In *2007 IEEE Wireless Communications and Networking Conference*, pages 2866–2871, March 2007.

[15] P. McDonald, D. Geraghty, I. Humphreys, S. Farrell, and V. Cahill. Sensor network with delay tolerance (sendt). In *2007 16th International Conference on Computer Communications and Networks*, pages 1333–1338, Aug 2007.

[16] S. Misra, B.K. Saha, and S. Pal. *Opportunistic Mobile Networks: Advances and Applications*. Computer Communications and Networks. Springer International Publishing, 2016.

[17] J. Nin and D. Villatoro. *Citizen in Sensor Networks: Second International Workshop, CitiSens 2013, Barcelona, Spain, September 19, 2013, Revised Selected Papers*. Lecture Notes in Computer Science. Springer International Publishing, 2013.

[18] H. Ntareme and S. Domancich. Security and performance aspects of bytewalla: A delay tolerant network on smartphones. In *2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 449–454, Oct 2011.

[19] A. Pentland, R. Fletcher, and A. Hasson. Daknet: rethinking connectivity in developing nations. *Computer*, 37(1):78–83, Jan 2004.

[20] C. Perkins, E. Royer, and S. Das. Rfc 3561 ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.

[21] J. Postel. Transmission Control Protocol. RFC 793 (INTERNET STANDARD), September 1981. Updated by RFCs 1122, 3168, 6093, 6528.

[22] R.H. Rahman and M.R. Frater. 5 - delay-tolerant networks (dtns) for underwater communications. In J.J.P.C. Rodrigues, editor, *Advances*

in Delay-Tolerant Networks (DTNs), pages 81 – 103. Woodhead Publishing, Oxford, 2015.

[23] Fabrício Jorge Lopes Ribeiro, Aloysio de CastroPintoPedroza, and Luís Henrique Maciel Kosmalski Costa. Underwater monitoring system for oil exploration using acoustic sensor networks. *Telecommunication Systems*, 58(1):91–106, 2015.

[24] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, WDTN '05, pages 252–259, New York, NY, USA, 2005. ACM.

[25] Amin Vahdat, David Becker, et al. Epidemic routing for partially connected ad hoc networks, 2000.

[26] Antonio S. S. Vieira, Joao Goncalves Filho, Joaquim Celestino, and Ahmed Patel. Vdtn-tod: Routing protocol vanet/dtn based on trend of delivery. In *in AICT 2013, The Ninth Advanced International Conference on Telecommunications, 2013*, pages 135–141.

[27] Hao Qu Yuanming Ding and Guang Li. Black hole attack model and simulation for mobile ad hoc network. *International Journal of Innovative Computing, Infomation and Control*, 11(1):203–211, 2 2015.

[28] X. Zhang and D. Qiao. *Quality, Reliability, Security and Robustness in Heterogeneous Networks: 7th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, QShine 2010, and Dedicated Short Range Communications Workshop, DSRC 2010, Huston, TX, USA, November 17-19, 2010, Revised Selected Papers*. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg, 2012.

[29] Xiaolan Zhang, Giovanni Neglia, Jim Kurose, and Don Towsley. Performance modeling of epidemic routing. *Computer Networks*, 51(10):2867 – 2891, 2007.

EVALUATING PERFORMANCE CHARACTERISTIC OF OPPORTUNISTIC ROUTING PROTOCOLS IN ONE

CONTENTS

1.0 WHAT IS AN OPPORTUNISTIC NETWORK?

With the cost of computing continuing to decrease yearly (Greg O'Connor, 2014) consequently there is an equal increase in the amount of data that these new devices need to transfer over a network (BIG DATA UNIVERSE BEGINNING TO EXPLODE, 2012). Due to this networks are becoming more and more imbedding into our everyday life. However not all networks are born equal, and with the huge range of different appliances with communication abilities being made available to a consumer, some networks are just not suited to deal with challenges that are being put before them. Network topology is the main

reason for this, and stems from the huge difference in the nature of different devices. For instance, a common desktop computer will require a steady, reliable stream of information and so the protocols designed for this kind of interaction are not able to deal well in scenarios where transmission of data is frequently disrupted and the data's destination is not always in the current reach. To deal with these issues a plethora of new protocols and systems have been devised to tackle some common issues in today's society.

An opportunistic network is common name given to networks that attempt to handle such issues. These types of networks allow for the intercommunication of devices even if a static route does not exist. Typically, these systems are wireless that require no infrastructure to operate, this is because each device within the network acts as a node for data to pass to and from. A node can be made from any device that has the ability to communicate to its neighbouring devices so that messages can be sent through a network from one node to another, however how a node chooses to send this data is dependent on the protocol chosen. These nodes should also be self-assembling and self-organising so not to require human interventions to function. (http://shair.media.mit.edu/publications/saso13/infra_p2p.png)

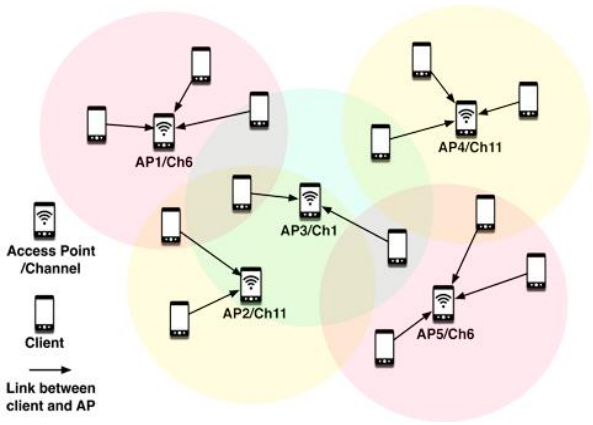


Figure 1: Mobile Ad Hock Network Example

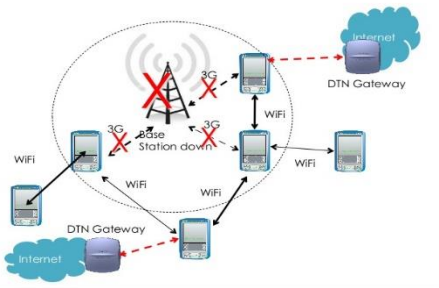
The most common use is when a static communication is not available and so nodes within a network are mostly mobile giving this system the name MANET(**Mobile Ad-hoc NETwork**). Despite not requiring infrastructure to operate if required a network can exploit static nodes to improve the reliability of communication throughout a network. The manner in how these nodes are used are vastly different to a traditional static communication relay and shall be explored further later in this report. (Corson, 1999)

As mentioned prior due to the nature of the internet in the 21st century, most systems commonly required an end to end connection, and protocols like the TCP/IP were developed with only static connection in mind. This protocol will then only attempt to find the destination of a message a few times before giving up and dropping the data. Whilst this is appropriate for networks where a single message in a flow of thousands is not important and large scale wiring can be built for these static machines, in areas like a battle field or a disaster zone this is far from optimal. In such cases this style of network and MANET's would struggle to deal with the frequent delays and large network partitions and so the likelihood that an important message potentially involving a life or death scenario not meeting its destination is very high. (LAFRANCE, 2014)

2.0 DISCONNECTED TOLERANT NETWORKS DTNS

In order to deal with these challenges, disconnection tolerant networks (DTN's) where designed and made to implement a store-carry-forward paradigm. This tactic centres around holding on to a message becoming its custodian until either the destination node becomes visible or another node who can increase the probability for the destination node to be found can reached. In either case the message is passed on and the new node is now the new custodian. (Disruption Tolerant Networking, 2014)

Figure 2: DTN Example



This method however comes at an expected cost; each node now requires additional space used as a buffer to hold on to each message before it is sent on. The danger with this is that a node's buffer can become full preventing further messages being held and leading to packets being dropped. These issues then lead to a range of other challenges that DTN's must deal with in order to have a strong message transfer rate. (http://image.slidesharecdn.com/trendsandchallengesindelaytolerantnetworkver1-140403092514-phpapp01/95/trends-and-challenges-in-delay-tolerant-network-dtn-or-mobile-opportunistic-network-oppnet-22-638.jpg?cb=1405987306)

For instance, in order for a DTN to deal with node receiving a large amount of traffic, various congestion protocols must be implemented to alter the path a packet of data passes through a network. In order to for a DTN to understand the state of a network, nodes must frequently communicate and provide up to date information about the reliability of each node in the network, so a sudden surge can be spotted and data re-routed to relief stress in a particular area.

With nodes being required to provide data about other neighbouring nodes security can become a huge issue. A malicious/faulty node can be used to manipulate traffic or prevent the transfer of data all together. This could be done be adding a node to the network that purposely provides incorrect information or made to simply drop data without passing it on. In the case this is a safety critical message with only one copy on the network the loss of this message is a disaster. Nodes that cause this kind of disruption are commonly referred to as a black hole's due to it likening of the galactic structure that devours light. The drawback of having this style of Ad hoc network is that there are no trusted nodes or infrastructure, making the identification of these hazards even the more challenging.

3.0 VEHICULAR AD HOC NETWORKS VANETS

One of the core concept behind opportunistic networks is its ability to exploit the nature of human social networks and predict how data can be passed to a destination. Building on top of previous work VANETS are a type of MANET that are used to provide communication between vehicles but also infrastructure. Like their counterparts they are also self-organizing but offer a different perspective on solving the issue of communication within vehicles. This type of network can be categorised by the very high mobility of the nodes it contains however as typically cars tend to drive in on roads and in a linear direction they also have a limited degree of freedom in mobility patterns. (Wenshuang Liang, 2015)

These types of systems could provide a huge range of potential to drivers in a similar way to how a phone can aid pedestrian. These types of message can come in three forms.

- Safety

- Convenience
- Pleasure

Crucially a vehicular node would also be able to send data that a human driver would not always be able to do. A common example of this would be in an accident situation where the driver has become incapacitated and safety signal would be sent. This signal would be given priority over the other types of messages and pass through the network to the appropriate authorities.

The key drawbacks of using this method is that communication can only occur for a limited time if cars are passing past each other, and as traffic can flow in the opposite direction a message could be passed in a way that leads it further away from the destination. A wireless communication standard was also in the writing (WAVE) this stated any wireless communication beyond 300m would be outside of coverage. This means that vehicles beyond this range would not know what is happening in the environment, and so in an accident only vehicles within range would be alerted, potentially too small in some cases.

4.0 VEHICULAR DISCONNECTED TOLERANT NETWORKS VDTNS

Naturally a combination of the efforts led to the expansion of DTN's for vehicular nodes (VDTN) to deal with long variable delays, high error rates and spares intermittent connectivity. Like a DTN the nodes within the system can become custodians of the messages only passing on the message when the opportunity arrives. Data is stored in bundles allowing for it to be forwarded to another vehicle when in range, this procedure allows to vehicles to effectively communicate.

However, in rural areas the number of nodes could become too sparse for nodes to send bundles between each other. In order to improve bundle, transfer reliability fixed DTN nodes can be placed, allowing nodes to deposit and collect data as they pass.

One main difference between VDTN and a DTN however is within the systems architecture. Within the VDTN the bundle protocols are placed below the network layer in contrast to being above the network layer. This allows for large bundles to be routed instead of small IP packets, allowing for fewer packet processing and less routing decisions.

One of the core concept behind opportunistic networks is its ability to exploit the nature of human social networks to predict how data can be passed to a destination. Following this VDTNS are a further expansion of a DTN explicitly designed for an environment where nodes are represented by a vehicular modes of transport. The challenges behind this lies behind huge increase in mobility with nodes frequently coming into and out of range of other nodes. (Syed Hassan Ahmed, n.d.)

5.0 ONE SIMULATOR OVERVIEW

With the great variation of networking protocols available it can be difficult to test the effectiveness on a level playing field in a real environment. Luckily a team of computer scientist (SINDTN and CATDTN), supported by Nokia developed a piece of software that can simulate the effects of networking protocol on the city of Helsinki. Named the Opportunistic Network Environment, the software is very complex

offering a range of tools that allows the user to examine a specific detail about the protocol's performance so an accurate evaluation can be made. (The ONE, n.d.) (http://www.nitdgp.ac.in/MCN-RG/images/result/onesim.png)

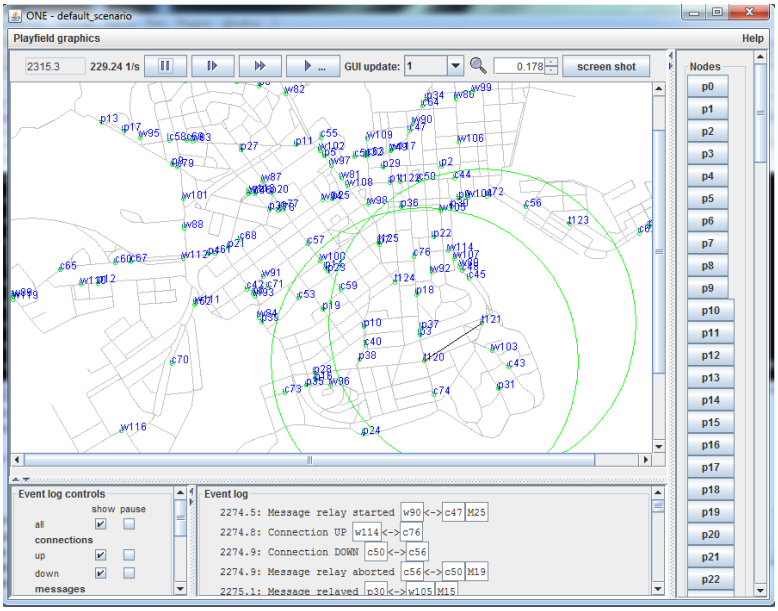


Figure 3: One simulator interface

The ONE simulator is an open source java based project that allows for the simulation, visualisation and reporting of a routing algorithm. The simulator can use provided data or generate its own in order to map out the movement and messaging of nodes for a simulation. One's aim is to act as a discrete event simulator, with events being broken down into stepped intervals. After each step the engine completes a number of updates modules that deal with each of the primary simulation functions.

In order to make full use of the software, the libraries can be imported into an eclipse IDE so to change any aspect of the simulators functionality. This allows for full control over every aspect of the program, however requires allot of attention to detail as there is an extensive code base to get to grips with. Non the less for those with less experience with java, the default version can be built, with the settings files still offering great customisation off the program.

The setting files enable the user to alter many factors about each node ability to transmit data, as well as it behaviour and number within the environment. These variables allow a user to compare the effects of various protocols on a range of conditions. For instance, we can reduce the number of nodes within a system to experience how a protocol result would change with scarcity or increase the number of nodes to see the effects on congestion.

One uses a visual representation of the real world map of Helsinki. This gives the software the ability to model protocol on a real world environment, built with roads, pedestrian pathways and buildings allowing for reliable results. However, control over this environment can be greatly altered through the GUI and the settings files that are provided.

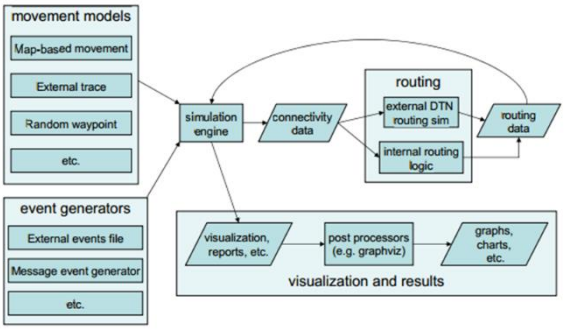


Figure 4: One simulator process diagram

The software also has a very robust reporting system that can be extended with additional reporting options in order to allow for the examination of new protocols not provided within the initial software. This would allow the user to potentially alter or develop new protocols, and compare them to the existing standard on the

same environment for a fair assessment. The report of the protocols performance itself offers a range of details about the outcome of a scenario with the additional feature of separating priority and non-priority messages to allow for focused aims. Accident awareness.

6.0 AIMS

In order to demonstrate and explore the functionality that the One simulator has to offer, two different protocols must be selected. Each protocol will be discussed so a full understanding can be achieved, then using this knowledge each of the selected protocols will be simulated multiple times for a test scenario. Using the results provided by the One simulator an accurate description of the protocols performance can be examined and then compared.

6.2 PROTOCOL A: EPIDEMIC

Epidemic is a DTN based protocol, meaning that it exploits use the store, carry and forward approach. This protocol aims to focus on message delivery rate and completes this through replication. (Becker, n.d.)

One of the main challenges with message delivery is that, a bundle of data send by a node could become lost quite easily. Within this type of network, the destination node will not know if its receiving message has become lost unlike in an end to end system. The nature of this network means that there is no easy method of making use of acknowledgment messages like the TC/IP protocol. The Epidemic protocol works on the basis that the more copies there are of a message within a system the higher the probability that the message is able to get to the destination node.

This replication event is triggered when one node encounters another, as this is a mobile network, nodes will be constantly moving in and out of range each other due to the limited 300m radius. In this event nodes will compare the messages they both have and exchange copies of all messages they do not have a copy of. Unlike some messaging systems now both nodes will be “custodians” of the messages and will continue this process with every node encountered. This type of method gets its name from the similarity with viruses, and their spread through a human population. Whilst there is a greater advantage of success there is also a huge amount of wasted resources when using this approach. Nodes require space in their buffer in order to hold bundles of data, by making a node use its buffer space to hold data for a destination node that is likely not going to be able to pass the data on to we cause congestion on the system. This congestion will reduce the effectiveness of other messages being sent on the same network, and could even prevent node refusing to take further data bundles, in this case this make this protocol very selfish.

The second major flaw with this protocol is also related to the buffer usage of nodes. As this system propagates the message data throughout the network and the message's destination is effectually has been reached, at this point the objective has been achieved but nearly all nodes who have received a copy of the message do not know this. The effect of this is that the message will continue to be passed on new nodes, whilst an unneeded copy of the message will sit in the buffers until the node decides to remove it further wasting potential resources. Other replication methods do attempt to limit the number of copies within the system so to greatly reduce the inefficiency.

One major advantage of this method comes from its implementation. This protocol is quite simple to program and quite easy to follow, and lacking the complexity that many of its rivals have, it is easier to understand and examine where an area of failure has occurred. If in an emergency where a highly

specified solution has not been developed and tested then deploying a protocol like this could be strong candidate, until a better option is made available.

6.3 PROTOCOL B: SPRAY AND WAIT

Spray and wait is a good example of a protocol aiming to refine the issues that come with many replications based approaches. As this protocol is also a DTN it too makes use of the store, carry and forward mechanism, because of this, it too will be working with buffers and limited node range. (Thrasyvoulos, n.d.)

The core concept of Spray and wait is to limit the amount of replication message within a network, by placing an upper limit. The result of this is to reduce the amounted resources while also having a high probability of message delivery.

Currently there two main versions of this protocol but both can be broken down into two phases, the "Spray" and then the "Wait". The only key difference in the two methods is how message is disseminated during the spray phase.

6.3.1 VANILLA

The simpler of the two versions, the process will start when a message is created. In combination to this alongside each message the node will have a limit value, representing the number of times a node can replicate the message. Once two nodes meet, like in epidemic a comparison is made concerning the messages each node has and node will swap the required data. The limit value is then decreased by one after the transmission is completed. Once a node's limit value reaches 1 then it is unable to create any more replicates of that message and so enters the wait phase.

6.3.2 BINARY

The second method is very much similar to the first, and the message is created in the same format however the number of messages sent changes. In this version a node passes half of its available copies to the receiving node. This is repeated until a node has only one copy remaining and like in the vanilla method then starts the wait phase. The benefit of sending messages in the way is that data is disseminated from the source node at a much quicker pace allowing for a greater spread.

These methods' do offer some improvements over the epidemic approach however the resource effectiveness is purely dependent on the size of the limit parameter. If too large of a value is selected when only a small cluster of node is present, then the less effect this method is over its counterpart. This is because with a large limit value the number of replications would be more than enough to reach the destination, causing a large amount of wasted resources. On the other hand, if too small of a value is selected then we greatly reduce the probability that a message is delivered to the destination. So for this protocol to be truly effective a limited parameter needs to be intelligently selected based on the data that the node knows about its neighbouring environment.

6.3.3 WAIT PHASE

When a node's limiting replication value reaches one, then no more copies can be produced by that node, from this point only direct transmission will be used to pass the message to its destination.

Direct transmission is the simplest mode of forwarding. Once a node has a message, the node will only transfer the message to the destination node. This vastly reduces the amount of resources used, but makes the latency unbounded as it based on the chance that the carrier node randomly encounters the destination. For the same reason the delivery probability is also based on the chance the two nodes meet, in this case if the carrying node is traveling in an opposite direction to the destination for a long period of time there is a strong chance this message will never get delivered.

7.0 SCENARIO EXPERIMENT – A COMPARISON OF POWER USAGE AND EFFECTIVNESS OF THE EPICIDEMIC AND SPRAY AND WAIT PROTOCALS

Transport is one of the key areas that is becoming difficult to deal with as human population continues to rapidly increase. With huge amounts of pressure of urban areas to handle increased traffic flow whilst businesses are being pressured to cut costs and cut emissions. For these reasons communication between vehicles is looking like one of the factors that could aid in these areas. To test the effectiveness of existing protocols for vehicular transport within the One simulator would provide a realistic outlook on the some of the challenges that implementing these types of systems could have.

In order to complete this task from a government and business perspective I will be focusing on the power consumption and effectiveness of each protocol within a variety of real life scenarios. Power usage is critical for any business, as alongside fuel it would add considerable costs to operations undertaken. Power usage is also important in optimising the amount of resources used within a network, there is no point having a very high message rate if the cost is having the hardware use considerable amounts of power.

```
56 Group.router = SprayAndWaitRouter
57 #Group.router = EpidemicRouter
```

Code altering routing Type

However even through the One emulator has a huge amount of potential, monitoring power usage is just not possible as this would require a large base of realistic data based on the power usage of cars processing instructions. To deal with this I am proposing to extrapolate the power usage to the amount of space within a buffers system and messaging overheads. A large and full buffer would require more processing when handling the engagements of node's in any protocol. This action can be compared to how a traditional desktop system uses more power to deal with main memory if it is full as it is required to search the drive. From this I can monitor the power usage in comparison to the message delivery success and number of nodes in the system.

```
133 Events1.size = 500k,1M
134 # range of message source/destination addresses
135 Events1.hosts = 0,3
136 #Events1.hosts = 0,10
137 #Events1.hosts = 0,20
138 #Events1.hosts = 0,30
139 #Events1.hosts = 0,40
140 #Events1.hosts = 0,50
141 #Events1.hosts = 0,60
142 #Events1.hosts = 0,70
143 #Events1.hosts = 0,80
144 #Events1.hosts = 0,90
145 #Events1.hosts = 0,100
```

Code altering the number of nodes in the network

In order to get a good variation of data, I will be altering various conditions that affect the nodes. First varying the number of nodes within a system, since we are working with a replication methods altering this aspect will allow both protocols best attributes to shine. Secondly from this data about the radio overhead that is created by each protocol can be extracted, giving an insight to the amount of power used changes as the number of nodes increases, in order to reflect on how a protocol

would behave in a real world environment where the number of nodes are never static.

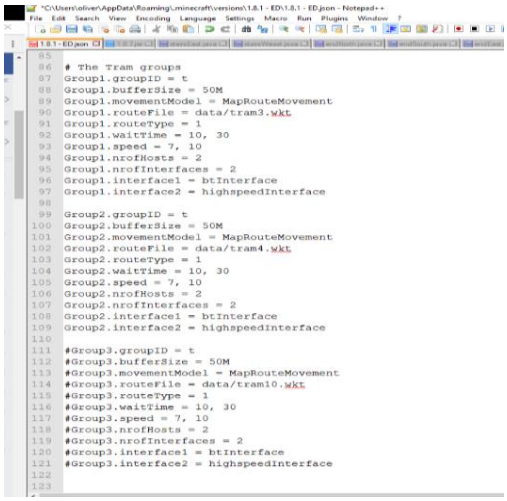
```
## Default settings for some routers set
ProphetRouter.secondsInTimeUnit = 30
SprayAndWaitRouter.nrofCopies = 2
#SprayAndWaitRouter.nrofCopies = 2
#SprayAndWaitRouter.nrofCopies = 3
#SprayAndWaitRouter.nrofCopies = 4
#SprayAndWaitRouter.nrofCopies = 5
#SprayAndWaitRouter.nrofCopies = 6
#SprayAndWaitRouter.nrofCopies = 7
#SprayAndWaitRouter.nrofCopies = 8
#SprayAndWaitRouter.nrofCopies = 9
#SprayAndWaitRouter.nrofCopies = 10
SprayAndWaitRouter.binaryMode = true

```

Code altering the replication limit in the Spray and Wait protocol

So to ensure a fair examination we will be also testing the Spray wait protocol by manipulating its replication limiting value. By doing this, a comparison can be drawn from the amount of replication a message effect its power consumption and effectiveness.

Code adding trams into the network

Lastly the method/type of transport would also provide key information based on the speed and mobility of nodes. Testing protocols on cars who are more numerous, less predictable and move faster in some cases could have a massive effect of the success of a protocol, whereas testing on slower transport such as busses or trams may allow for more predictable, slow movement allowing for longer and easier communication between nodes.



```
# The Tram groups
Group1.groupID = t
Group1.bufferSize = 50M
Group1.movementModel = MapRouteMovement
Group1.routeFile = data/tram3.wkt
Group1.routeType = 1
Group1.waitTime = 10, 30
Group1.speed = 7, 10
Group1.nrofHosts = 2
Group1.nrofInterfaces = 2
Group1.interface1 = btInterface
Group1.interface2 = highspeedInterface

Group2.groupID = t
Group2.bufferSize = 50M
Group2.movementModel = MapRouteMovement
Group2.routeFile = data/tram4.wkt
Group2.routeType = 1
Group2.waitTime = 10, 30
Group2.speed = 7, 10
Group2.nrofHosts = 2
Group2.nrofInterfaces = 2
Group2.interface1 = btInterface
Group2.interface2 = highspeedInterface

#Group3.groupID = t
#Group3.bufferSize = 50M
#Group3.movementModel = MapRouteMovement
#Group3.routeFile = data/tram10.wkt
#Group3.routeType = 1
#Group3.waitTime = 10, 30
#Group3.speed = 7, 10
#Group3.nrofHosts = 2
#Group3.nrofInterfaces = 2
#Group3.interface1 = btInterface
#Group3.interface2 = highspeedInterface
```

8.0 RESULTS OVERVIEW

8.1 DIRECT COMPARISON

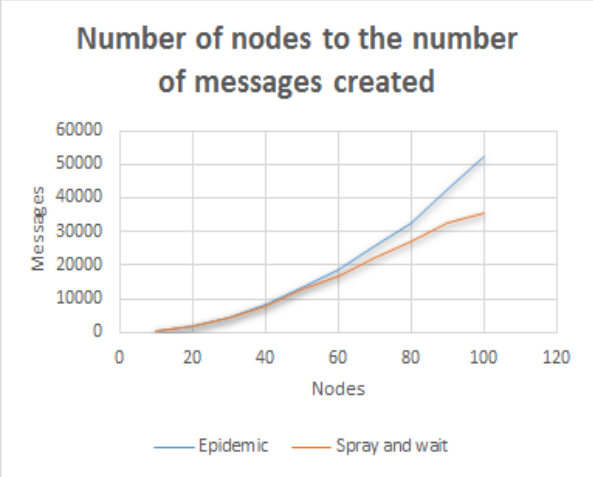
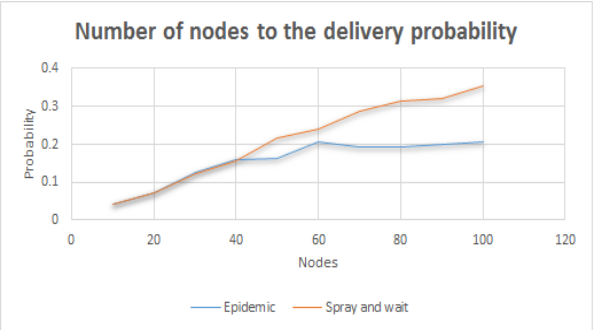
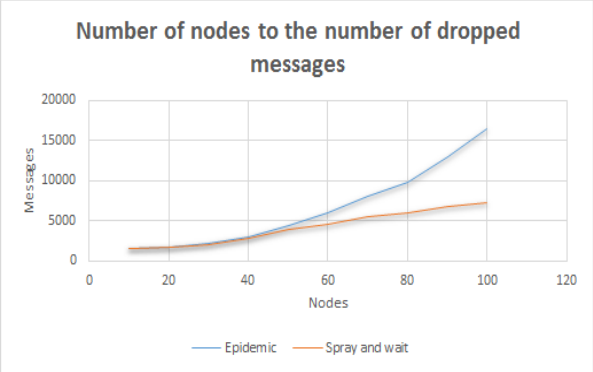


Figure 5: One simulator results comparing Spray and Wait and Epidemic protocols directly

The first stage of the comparing the power usage and effectiveness between the Epidemic and Spray and Wait protocols created two sets of key data. This data was formed by increasing the number of nodes in the system by ten, for each iteration until the number of total nodes reached one hundred. For these scenarios to be accurate removal of any other node that did not represent a car was needed, so that a true simulation of a VANET could take place.





Applying the Epidemic and binary version of the Spray and wait protocol, allowed me to understand how the differences in replication methods would have the network, and how the network would react as the number of nodes grew.

Initially the two protocols did not differ greatly in the number of messages created, this however is mainly because of the limit value selected for this stage in the scenario. In this stage of the experiment a limit value of six was used as this was the default provided by the one simulator.

This results in the early stages of the graph looking similar as the proportion of the nodes holding a replication of the message is large, the wait phase of the Spray and wait protocol has been made redundant in these cases as most nodes in the network are still receiving duplicates of the message.

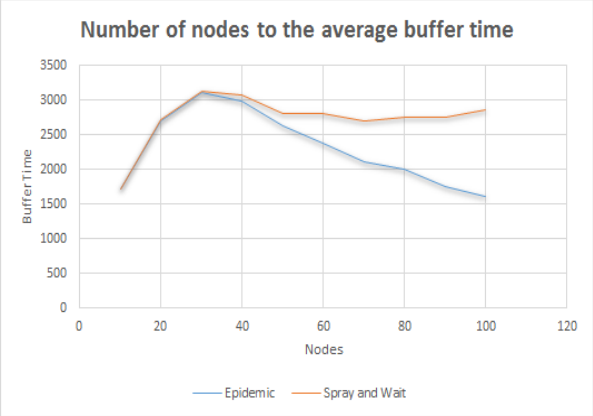
This is the leading factor to what causes the similar behaviour up to the point where the number of nodes increases to a point where the wait phase is having more effect on the network. As less of the total proportion of nodes in the network have a copy of the message.

Going on from this point the number of messages starts to greatly differ and from the data provided it looks as if it would continue to for some while. These changes in the number of messages created is greatly important for measuring the effectiveness and power consumption. As a message is created, this uses CPU processing and in effect using more power, secondly the more messages in a network the more a network is congested, and can cause knock on effects on the reliability of a protocol, so to further increase power usage.

Continuing the comparisons between the two protocols, it is evident that the knock on effects of the number of messages on the network is detrimental to the performance. In both the number of dropped message's and in the delivery probability we see the Spray and wait protocol outperforming the other, with a greater success rate with a lower number of dropped packets. However similar to the effects in the number of message's created, same pattern occurs in the number of dropped message's and delivery probability, where both protocols start equal at early stages where a smaller number of nodes are present, then gradually the Spray and wait gains the advantage as the wait phase has more influence on the behaviour.

Congestion of the network is likely to be the main reason for this; as epidemic message exponentially increases based on the number of nodes in a system that causes a huge flood of redundant data in the system. The dropped packets occur when a node's buffer is full and so rejects the incoming messages, therefor contributing to the reduction of the message delivery probability, as seen in the graphs. Alongside this a filled buffer is also a huge cause of power consumption and further reduces the long term viability of the epidemic protocol. This effect is amplified by the fact out scenario plays out for forty-two scenario hours, as a filled buffer will continue to drop packets unless the data it is holding is dropped.

On the other hand, the Spray and wait protocol, whilst still doing better, still causes a large number of dropped messages and in the case of an emergency, where the number of potential nodes are low neither protocol would perform well. In a node scarce environment, a less than 10 percent delivery probability would not be satisfactory could cause potentially important message to be lost.



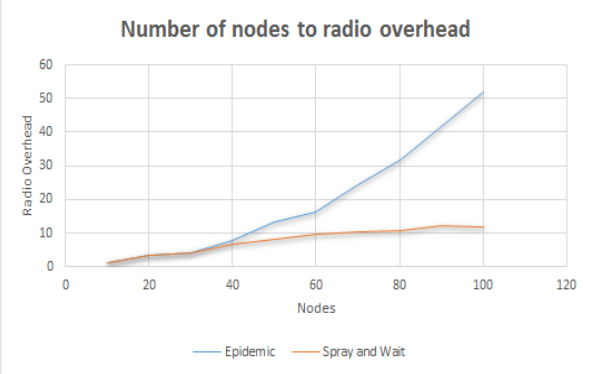


Figure 6: Comparing power overheads between Spray and Wait and Epidemic protocols

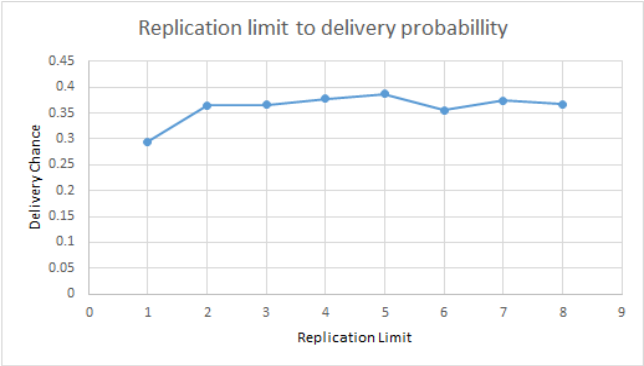
Lastly continuing the focus on power consumption. Looking at the comparison of the average time a message is kept in the buffer allows an understanding of how frequent messages are being sent. With both protocols the buffer time starts of low, mainly due to the lack of messages actually being created as seen in previous graphs. This rises massively as an exponential number of messages is created, but then drops off very quickly.

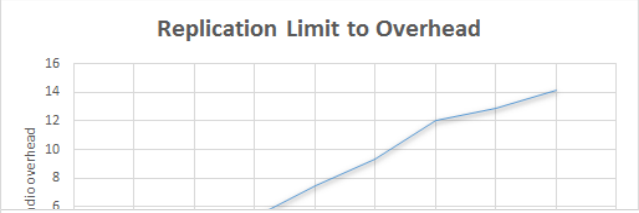
One explanation for this is that because the number of nodes in the system is high enough to generate a large amount of messages but too small enough so that nodes are frequently finding themselves out of range of other nodes, preventing message transfer. To improve this increasing the power of the radio signal would allow for further communication however but would also increase power consumption significantly.

The fall in the average buffer time occurs when the number of nodes are frequent enough so that not only a large amount of messages being generated but the number of nodes are so high that nodes are frequently coming into range with each other, causing more messages to be sent, reducing the average. The difference in the decline amount between the two protocols can be explained by the wait phase of Spray and wait protocol. Previously with a limit number of six it is unlikely the wait phase was having much effect explaining the similarity in the beginning. But as the number of nodes increases this limit value is being reached the wait phase is occurring quicker, from this point message replication stops vastly reducing the increase in the number of message sent. At this point where direct transmission occurs messages are being sent less frequently, thus keeping the average buffer time high.

This information is backed up by the levelling out that occurs in the radio overhead when using Spray and wait protocol as the frequency of messages being sent declined massively in comparison. Overall from this section is it apparent that not only is the buffers becoming full when using Epidemic, but the messaging rate is so high the network has a huge radio overhead. This overhead is directly proportional to the power usage the protocol uses. In this cases it is apparent the radio transmitter is being used so frequently it looks to be exponential as the number of nodes rise. Continual use of this protocol in this situation would cause a huge unneeded power consumption and would be equivalent to having GPS enabled on a mobile device when no application requires it.

8.2 COMPARISON OF REPLICATION LIMIT VALUE IN SPRAY AND WAIT





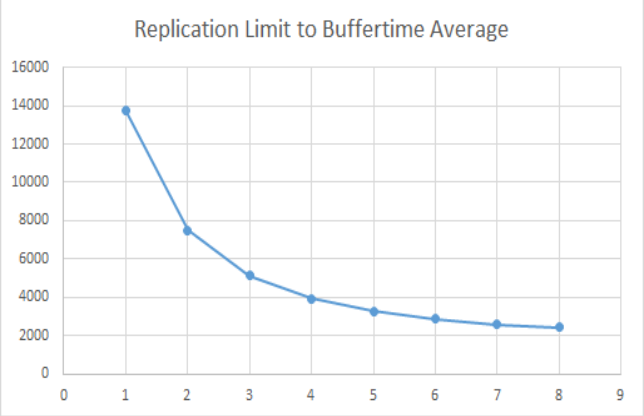


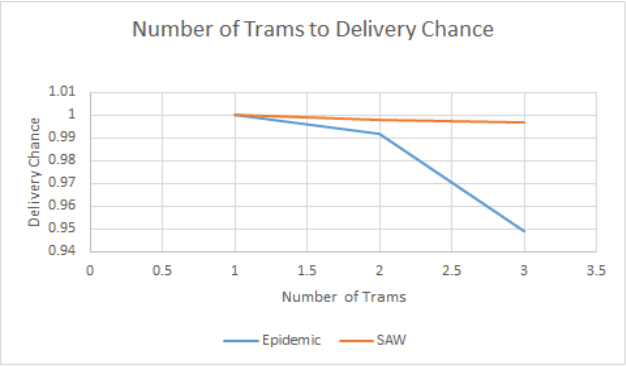
Figure 7: Comparing One simulator results of an increasing replication limiting value of the Spray and Wait protocol

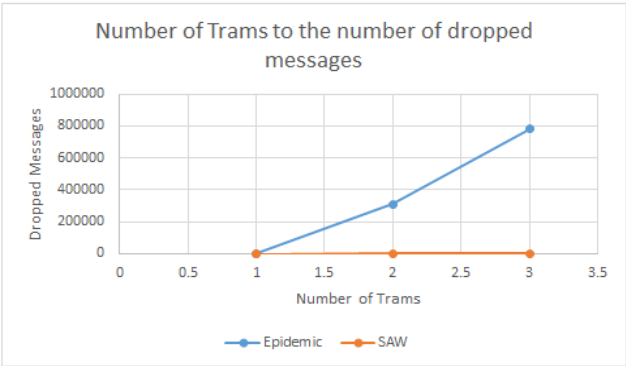
From the previous efforts I have been comparing the Epidemic protocol to the binary version of Spray and wait with a replication limit of six. Through examination it is clear the protocols ability to limit the amount of replication has allowed it to be vastly superior in effectiveness, however as noted this ability is dependent on the value set for replication. In order to fully compare the two in this scenario a full examination of effects of changing this limit value would be required.

Using 100 nodes and increasing the replication limit value it is evident that the replication has some effect on delivery probability, with a limit value of two showing how using some replication increases the probability somewhat. Increasing the limit value further improves the delivery probability slightly, however after using a value of five this probability decrease as the network suffers the signs of having too much replication and so congestion in the system. The consequence is that as seen in the Epidemic protocol, increasing the replication vastly increases radio overhead. It is then very important that context behind the network is made a crucial factor when choosing a limiting value to see whether cost is as important as delivery successes. Secondly to become truly optimal a possible extension to this protocol could be explore and allow the protocol to dynamically alter the replication limit based on the number of nodes in a network. This however is outside the scope of this experiment and would take considerable time to undertake develop a system to do so.

Noted however that increasing replication reduces the average time a message remains in the buffer, and so promotes more messages being sent, further backing the radio overhead increase. But also uses CPU power in order to send and create messages promoting congestion of a network.

8.3 TRAM LINE TESTING





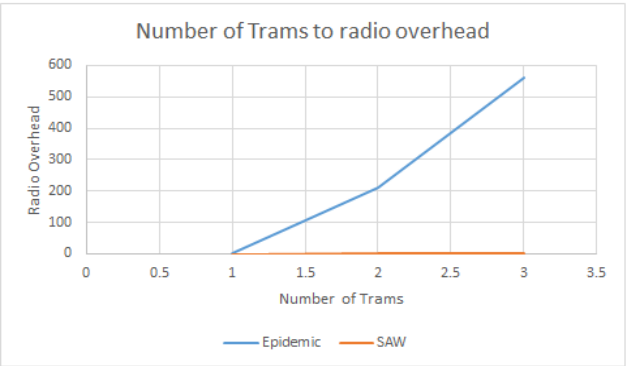


Figure 8: Comparing the results of the One simulator in an environment only containing an increasing number of trams

As a VANET network does encompass cars, it does also take into account many other forms of vehicle's whose behavior are very different to the ones currently explored. So that these two protocols can be better evaluated data must be collected from other sources.

To represent this the One simulator can have support grouped nodes allowing us to create a front and back position of a tram vehicle. Luckily data for a trams path line is also provided, creating fixed routes on the maps that the nodes will follow. This is much more realistic than previous car based experiments, as vehicle's driven by individuals are susceptible to the whim of their will, making the effects pseudorandom.

All three of these graphs show the effects of the increased number of trams running the protocols in a network. It must be noted though that applying the knowledge from previous experiments we alter the Spray and Waits limit value as the number of trams increases; this is done in L = 2N. This allows the wait phase of the protocol to have a sizable effect on the results.

The comparative results support the same conclusion of previous experiments however in this case the data is much more extreme, showing a huge difference in the performance. The source is the same, a huge difference in the number of messages due to Epidemics uncontrollable replication rate.

The key difference between the experiments however is the frequency that node are interacting, as the trams are traveling on fix liner paths communication is a near constant, leading to very high delivery success rate as nodes are forced into contact with one another. This creates an exponential amount of replication in the case of Epidemic where the repeated contact with nodes sparks another wave of replication resulting in ridiculous overheads and dropped packets when the buffers inevitably become full

8.4 CONCLUSION OF RESULTS

Concluding the combination of areas studied, using the Spray and wait protocol provides a vastly superior message success rate within the number of nodes test on. This stems from the protocols ability to control the amount of replication, in order to prevent network congestion and allows messages to be sent more reliably. Because of this the protocol causes a comparative reduction in the amount of wasted power consumption and the relative overheads that come with it.

One way to improve the effectiveness of both protocols would be to increase the amount of memory space in the buffer, this would increase the amount of message that can be stored in each individual node. The result of this would be a decline in the amount of messages dropped, thus increasing delivery chance. However, the main reason why this was not included with in the study was because both protocols are based on replication based ideologies. The exponential effect this on results as the number of nodes increases would result in an exponential amount of buffer memory required. To simulate such effects, a real number of nodes would be required. E.g 1000's and given the equipment available to me I was not able to perform such a simulation on the One program. Therefor for the benefit of this study increasing the amount of buffer memory was excluded, so that an accurate test could be produced within the bounds of my ability.

However as mentioned before strong control is still required over the limiting value, that maybe not always possible for a dynamic situation that occurs in a VANET environment where cars or other fast moving vehicles are at the forefront. Secondly one assumption within the One network is that it assumes the nodes will circulate this city consistently. Whilst one allows for roads, a VANET's true behaviour is allot harder to replicate and more commonly a vehicle will drive A to B then will be parked in an offline state for a large amount of time. In this case more nodes will leave the region causing more aborted nodes to occur. Vastly reducing message delivery chances and potentially giving more support for more replication within a network, making epidemic more of a viable candidate. To further expand on this work more accurate tests will be required in a real world environment so that a VANET's nature can be truly taken into account. However, given the resources supplied these result do provide a logical representation of the events that could occur.

Lastly it should also be noted my decisions are based on the few numbers of tests completed and to further validate my conclusion's a larger range of testing could be tested. As the behaviours of each protocol could alter beyond the range of my current tests.

9.0 FURTHER EFFECTS ON THE WORLD

Through this examination is it clear that opportunistic networks have the ability to implement a reliable mode of communication between vehicles. Offering clear cut advantages such as the ability to send out distress signals or provided additional comforts like connection to the internet media. These features will drastically alter an individual's relationship with their mode of transport as this field continues to grow. However crucially as humanity begins to take the back seat when it comes to driving and transport with the inevitable introduction of self-driving cars as seen in the DARPA challenge, this is where I feel opportunistic networks will be at their most useful in fighting some of today's biggest transport issues.

Figure 9: Darpa grand challenge logo.

Through a reliable network automated vehicles would allow a high amount of statistical data to be sent and received providing big data information about the vehicle and road conditions. This data could be processed either by the nodes or a stationary point, allowing decision making systems to make an analysis of a situation. In this instance self-driving cars would have the ability to dynamically deal with not only the information directly presented in front of them, but also keep up with the changing road conditions further in the distance. As these type of systems grow, so too would the commercial aspect expanding on the platform similar to how we have seen in mobile phone devices designs. (http://www.3rd-st.com/Darpa/GrandChalleng2005Logo.jpg)

Whilst it is likely these protocols will not ever be implemented for this scenario due to the sophistication of a real world problem, these protocols offer a strong foundation to what can be achieved already until the market fully develops. Safety messages already have the potential in be implanted and offer lifesaving capabilities, allowing for help to be request even if a driver was to be incapacitated. These protocols do offer a potential solution to these issues and as I mentioned earlier for a cheap and quick answer where time is not available for an optimal protocol to be fashioned both of these protocols can be deployed and tested relatively quickly allowing for rapid implementation. One good example for this would be within a disaster zone where no infrastructure is available, and time is of the essence.

However, for these developments to occur in a VANET environment, extensive research will be required to further test the effectiveness of protocols in the field as current simulations are not fully reliable and with possible life's and business on the line definite result rates are required.

Secondly a continued demand for technology within vehicles will be required so that funding for these types of networks can continue, whilst we have already seen growth in this sector this has only been with the small region of safety and traffic mechanism's. One potential issue is that demand for additional feature's plateau and other issue such as ethics and government prevent further development. This was seen with mobile devices and extending the reach and effectiveness has been a long impacting campaign, whilst these types of networks do not require infrastructure making such development's easier there are potential other infrastructures that could be required like data processing plants if this idea every took off to the same level as current phone networks.

An additional challenge opportunistic networks will face in the near future is ensuring that these life critical networks are secure. With an increasing number of cybercrimes being committed it is inevitable that some of these networks will become targeted for various reasons. As these networks commonly lack infrastructure is can be difficult to choose what nodes to truest, as a node may leave and join a network at any point, allowing it to release harmful message bundles. If an event of this kind is not planned for, the potential loss in life would very much dampen investors' enthusiasm for implementing these systems.

Lastly government's will have the last say on how these operations develop all over the world, with the potential for this to mature into something like phone networks, who would own an opportunistic network? Competition could promote further and improved research into the field, but could also slow implementation as seen already with most car manufacturer's already trying to do things "their own way" rather than fully cooperating.

BIBLIOGRAPHY

BIG DATA UNIVERSE BEGINNING TO EXPLODE. (2012). Retrieved from CSC: http://www.csc.com/insights/flxwd/78931-big_data_universe_beginning_to_explode

Corson, S. (1999, January). *IETF*. Retrieved from Mobile Ad hoc Networking (MANET): https://www.ietf.org/rfc/rfc2501.txt

Disruption Tolerant Networking. (2014, March 14). Retrieved from NASA.gov: https://www.nasa.gov/directorates/heo/scan/engineering/technology/txt_dtn.html

Greg O'Connor, A. (2014, April 19). *Moore's law gives way to Bezos's law*. Retrieved from Gigaom: https://gigaom.com/2014/04/19/moores-law-gives-way-to-bezoss-law/

http://image.slidesharecdn.com/trendsandchallengesindelaytolerantnetworkver1-140403092514-phpapp01/95/trends-and-challenges-in-delay-tolerant-network-dtn-or-mobile-opportunistic-network-oppnet-22-638.jpg?cb=1405987306. (n.d.). *DTN Image.*

http://shair.media.mit.edu/publications/saso13/infra_p2p.png. (n.d.). *Mobile Ad hock Image.*

http://www.3rd-st.com/Darpa/GrandChalleng2005Logo.jpg. (n.d.). *Darpa Grand Challenge Logo.*

http://www.nitdgp.ac.in/MCN-RG/images/result/onesim.png. (n.d.). *One image.*

LAFRANCE, A. C. (2014, April 25). *This was not just a nice thing, it was the very nature of the Internet.* Retrieved from Theatlantic: http://www.theatlantic.com/technology/archive/2014/04/the-best-writing-on-net-neutrality/361237/

Syed Hassan Ahmed, H. K. (n.d.). *Vehicular Delay Tolerant Network (VDTN): Routing perspectives*. Retrieved from ieeexplore: http://ieeexplore.ieee.org/document/7158095/?reload=true&tp=&arnumber=7158095

The ONE. (n.d.). Retrieved from Github: https://akeranen.github.io/the-one/

Wenshuang Liang, Z. L. (2015, August 31). *Vehicular Ad Hoc Networks: Architectures, Research Issues, Methodologies, Challenges, and Trends*. Retrieved from sagepub: http://dsn.sagepub.com/content/11/8/745303.full

Becker, A. V. (n.d.). *Epidemic Routing for Partially-Connected Ad Hoc Networks*. Retrieved from http://issg.cs.duke.edu/epidemic/epidemic.pdf

Thrasyvoulos. (n.d.). *Spray and Wait: An Efficient Routing Scheme for*. Retrieved from http://chants.cs.ucsb.edu/2005/papers/paper-SpyPso.pdf

Evaluating Performance characteristics of opportunistic routing protocols in ONE

1.Introduction

In the last decade there has been a large increase in the production and usage of mobile technology. Human levels of communication has risen greatly and become more mobile as the introduction of smartphones has made most aspects of communication more convenient. The usage of mobile technology has expanded year on year and now worldwide the number of smartphones in use has risen to 2.16 billion.[1]

One of the key problems with smartphones and mobile technology in general is a lack of user connectivity. Designing an environment which supports such unique networks like mobile communication networks, as well as supporting fixed networks, is currently restricted by wireless communication constraints and the mobility of the user. However, with current networks it is assumed that there exists an end to end path where there is low message loss and low delay[2].

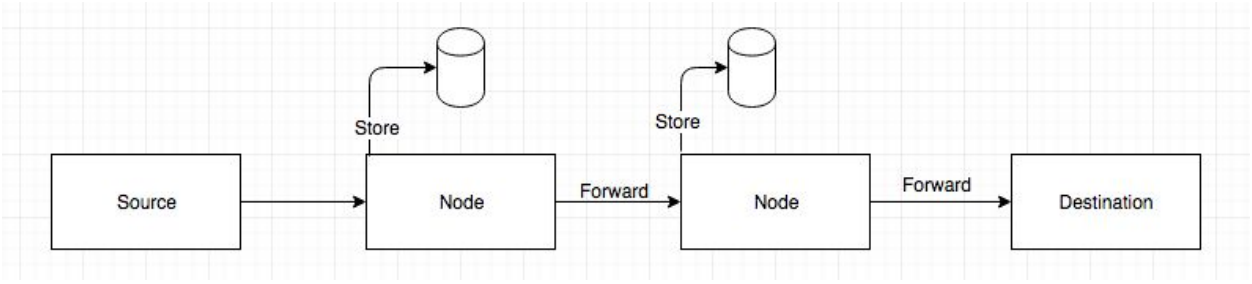
In emergency situations where someone requires an ambulance, a message should be sent to the emergency services with minimal delay and high reliability so that messages are not dropped, however with a network of nodes that are constantly moving, a fixed path may not be feasible or the most effective solution. This raises questions about a different kind of network called opportunistic networks that use DTN routing protocols which would be used if there was no end to end path. This project will be evaluating several routing protocols against an emergency scenario and trying to find ways of improving the protocols.

2.Opportunistic Networks

Opportunistic networks have complex topologies that can be modelled as complex temporal graphs with partitions due to potential lack of connectivity between sources and destinations, it does this without any interaction or utilisation of infrastructure and therefore reducing any overhead caused by infrastructure level contact.

Opportunistic networks are distinctly different to fixed networks such as wide area networks (WANs) which assume that there is an end to end route of communication from source to destination with minimal packet loss and low delay. However, in a mobile environment where human carried smartphones as well as vehicular mobile technology are communicating in different ways, these mobile technologies have several different variables to consider which can affect communication between nodes. These include if a node suffers a loss of power and disconnects, or if they move outside of the network, then an end to end route is not feasible. Opportunistic networks however have support for these unpredictable communication patterns with delay and disconnection tolerant network (DTN) routing protocols.[3]

Nodes within the opportunistic network have two functions, the first is node discovery; this is where nodes are able to search their vicinity or communication range for other nodes in which to communicate with. The second function is the message exchange where two nodes can forward messages to each other.
To support disconnection and delays, DTNs follow a paradigm of ‘Store-Carry-Forward’, this is where a node will receive a message from the source or a transmitting node, store the message within its buffer until a relay node is within the nodes communication range, in which case the node will then forward the message to the other node. This concept is suited for unpredictable communication patterns as the transmitting node does not know when or where the next node they will be in contact with. It also means the route is more dynamically formed and any node could potentially be the next hop if it provides a positive impact on the message reaching its destination.[4][5]



(Figure 1 : Diagram of store-carry-forward paradigm)[5]

The component that gives opportunistic networks its intelligence in terms of the handling of messages and efficient forwarding are the routing protocols. These routing protocols essentially give nodes a level of intelligence and organisation to know which node is the most beneficial node to forward the message to.

There are two kinds of routing protocol, the first is forwarding based routing, this is one of the simplest routing types as it only focuses on a single copy message and the custody of that message. The custody of the message is accountable to the node that is in possession of that message and that node alone, when that transmitting node forwards the message to another node then it relinquishes the custody of that message to the receiving node. The second kind of protocol is replication based protocols, these protocols focus on the spreading or dissemination of messages throughout the network by means of replicating messages. The difference between replication and forwarding based protocols is that instead of removing the message after the message has been forwarded to the relay node; which is how forwarding based protocols operate, replication based nodes retain the message as well as forwarding the message in case they come into contact with the destination. As there are multiple copies of the message on the network, this means there is an increased probability that the message will reach its destination. The advantages over forward based protocols are that the delay of delivery is significantly reduced, however the downside to this is that multiple messages can have an impact on the resources of the nodes as well as if that message has reached its

destination,there could potentially be redundant copies of that message in the network providing outdated information.

OppNets are related to three other types of network these are MANETs(Mobile Ad Hoc Network), VANETs (Vehicle Ad Hoc Network) and peer to peer networks. MANETs and VANETs are similar in their techniques and differ in their platforms as MANETs focus on mobile technology and VANETs focus on vehicle technology. These networks are both similar to OppNets as they are both Ad Hoc meaning the nodes within the network will directly communicate with each other and not use a centralised point like a router or a server to do their communication for them. They both support the element of node mobility and node availability as nodes can move in and out of a communication range as well as a node can connect and disconnect to the network unpredictably. The one key difference between them are their intended uses and how that has impacted on their architecture. MANETs occupy the networking layer which makes them responsible for routing potentially giving the network the ability to create an end to end route via an intermediate, MANETs where primarily used for emergency scenarios where each node trusted each other. However opportunistic networks are formed with unrelated nodes that are unknown to each other, there is no control over the routing as opportunistic networks occupy the application layer, so instead of using the MANETs multi-hop approach to exchanging messages, opportunistic networks use a single hop exchange where a message is directly exchanged to a neighbouring node.[3]

The other network which opportunistic networks relate to are Peer to peer networks (P2P), which rely on the nodes in the network to share their hardware resources like processing power and storage to help the network function with tasks like file sharing. Each node in the network can be seen as a resource provider and requestor depending on the role the node is requesting information or exchanging a message. The similarities between opportunistic networks and P2P are that they both have a client-server architecture where the network can be seen as a server consuming node information and a client requesting information. They both also have a similar purpose of initiating collaboration between nodes with no centralised component.[3]

3.One Simulator

The simulator which is being used for this experimental scenario is Opportunistic Network (ONE) simulator. The purpose of ONE is to provide an environment to evaluate a routing protocol given a pre-defined scenario. The simulator supports a range of customisation; as users can define their own scenarios in which to test the protocols, the movement patterns of nodes within the simulation can be defined as

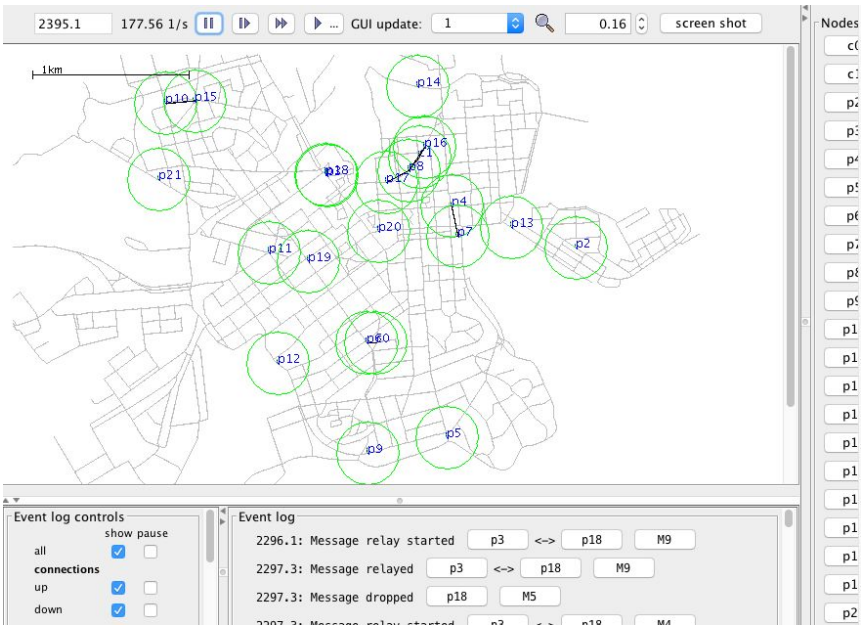
synthetic random movement patterns or real world movement. There are some of the most well known routing protocols pre-built in ONE and this can be added to if the user implements their own protocol. Finally post-simulation, reports are generated on the performance of the protocol, such as its latency and message delivery to message drop ratio etc. [6]

ONE simulator is based on the map of the city of Helsinki, with roads, paths and tram lines defined. There are several kinds of node that provides a more realistic interaction and feel that the city is being simulated. There are pedestrian, cars, buses and tram nodes which have their own movement patterns and behaviours which they follow during simulation. Static nodes can also be implemented if the user wants to specify a place of interest like a hospital or office.

These groups of nodes have their own sets of characteristics and attribute which can be altered by the user. Some of the characteristics that will come into in the scenario are the communication interfaces which is how the node will be communicating (bluetooth or wifi), these interfaces have their own attributes such as their speed and communication range for exchanging messages which can also be altered. The nodes buffer which would be used for storing the messages can have its size changed to affect the quantity of messages that can be stored. Finally more specific movement patterns can be defined such as evening or morning style movement.[6]

Messages in ONE are generated within the event generation module, this contains all the necessary attributes that make up the routing of the message, such as the size of the message and the interval of message creation. One of the attributes that will be an important part of setting up the scenario is the host and destination nodes as the source nodes will be set a number of pedestrian nodes within the group and the destination nodes will be set as the vehicle nodes in the network.

Finally ONE has a very simplistic graphical interface in which to interact and inspect the network scenarios activities and if there are any problems with the nodes when communicating. There are four sections of importance in the interface; the first and most central component is the map, this shows all the nodes in the network, there communication ranges (represented as a green ring around the node), the number of messages in the buffer(shown as blue and red blocks varying in size) and finally the transactions between the nodes shown as a black line. The top of the interface



controls the time frame of the simulation which can be paused, increased and reduces speed. The right side of the interface has a list of all the nodes in the network which can be selected and highlights the node of interest. Finally the bottom of the interface shows a log of all the messages and there encounters between nodes any action given.[7]

4.DTN Routing Protocols

4.1Spray and wait

Spray and wait [14] is a replication based protocol which has two phases of message forwarding, the initial stage is the ‘spraying’ stage where a set amount of copies of the message are sent from the source node to the neighbouring nodes which is within the nodes communication range. The second stage is the ‘wait’ stage, if the spraying phase is not successful in reaching the destination then the relay nodes which received a copy of the message perform a direct transmission, meaning that node will keep the message and the responsibility to get the message to its destination, this stage will end when the message has reached its destination or the message TTL(Time To Live) has expired. This two phase process provides a ‘jump-start’ by spreading the message in a similar way to the flooding based protocol epidemic but with the wait process so that the number of redundant messages are kept to a minimum.

There is another kind of Spray and wait know as ‘Binary Spray and wait’ which differs from the original, as with the spraying phase if the transmitting node has more than one copy of the message, then the node can forward half the number of copies to a neighboring relay node and the transmitting node keeps half the copies for themselves. [8][11]

4.2.PRoPHET

PRoPHET (Probability Routing Protocol Using History of Encounters and Transitivity) protocol is quite different from other routing protocols as the decision to forward a message depends on a calculation on the history of encounters that the two nodes have (the transmitting and receiving node).The greater the number of encounters the one node has, makes the probability of future contact more possible between nodes. If the probability of contact is higher than the transmitting node then a copy of the message will be forwarded to the relay node. [8][9][13]

There are however different types of PRoPHET that take into account other constraints not just the probability of encounters. ‘PRoPHET +’ takes into consideration the size of a node’s buffer size, battery power, location as well as the encounter probability. This solves an issue with the original PRoPHET protocol which was delay this is solved in scenarios where multiple nodes contact the transmitting node simultaneously as if some of contacting nodes had the same encounter value then these other parameters would be included and the highest value of that would receive the message. The other PRoPHET type is distance based PRoPHET which simply put checks the neighbouring nodes with regards to the distance; the closest neighbouring

node would receive the message as this would also reduce the delay time and the message has less to travel.[13]

5.Pedestrian to Ambulance emergency scenario

Simulation parameters	Simulation Values
Number of nodes	12-60(Increments of 12 - 5:1 pedestrians-vehicles)
Message size	300 Kb - 3 Mb
Buffer size	2 Mb - 25 Mb
Message TTL	300(5 hours)
Simulation time	10,000 (24 hours)
Communication range	250 meters
Movement model	Shortest Path Map Based Movement

The scenario that will be used to evaluate these two routing protocols is an emergency style scenario where set numbers of pedestrian nodes are sending a messages to the ambulance vehicles on the map. The constraints that are going to be changed to evaluate the routing protocols performance are, the number of nodes on the map (an increase in pedestrian and vehicle nodes), the size of the messages being sent(varying between 300Kb to 3 Mb) and finally the nodes buffer size.

With an emergency scenario like this, the statistics that would be relied on are the delivery probability, the speed of the latency as this would need to be kept to a minimum and the number of messages being delivered as opposed to dropped messages, these statistics will be monitored and measured for each routing protocol in the same environment with the same constraints to make the experiment accurate and fair.

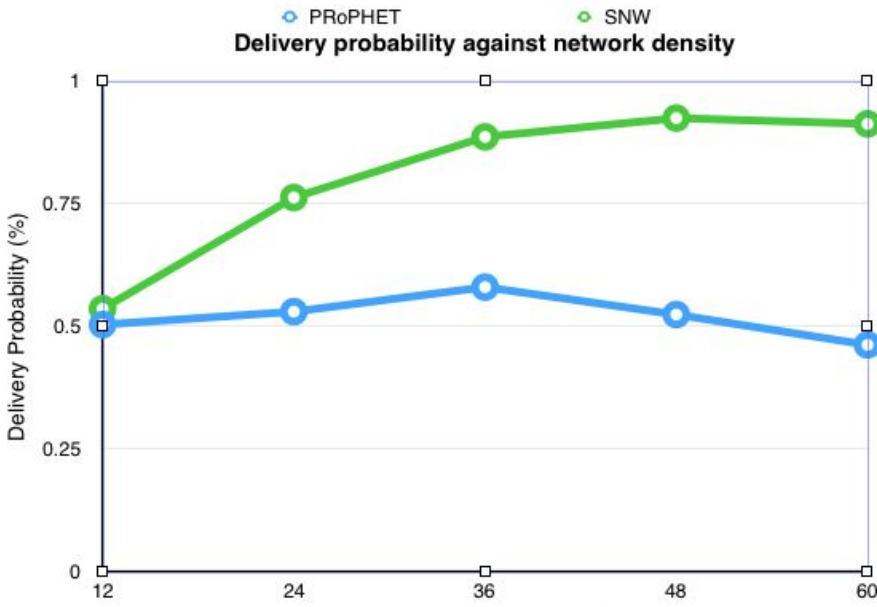
The environment setup with ONE simulator will be different from the defaults settings provided, instead of having the world with all types of nodes (Trams, Buses, Pedestrian and Cars). It will only have pedestrians and cars represented as ambulances, this is so the number of nodes can be changed with ratio for the number of ambulances to the number of pedestrians and that no other constraints are used to affect the results of the evaluation.

The decision to evaluate the routing protocols against these three constraints as these will have the most impact on the protocols performances but also these can happen in scenarios similar to this one, the number of nodes and the size of the messages sent will testing the protocols ability to handle congestion as well as memory management with their buffers. The number of

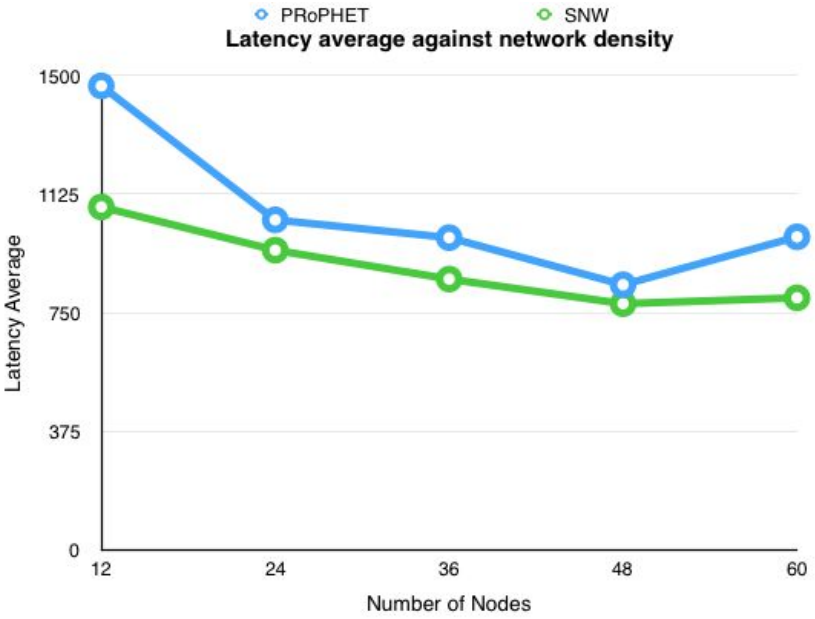
nodes in the scenario is related to real world scenarios as the number of devices entering an environment can increase and decrease exponentially so the protocols need to handle the traffic or even lack of traffic to make sure the message reaches its destination. Sizes of messages can also vary depending on the kind of message that is being sent, a handshake or a ping is a very small message(64 bytes), however streaming or image messages can be huge into the megabytes so the protocols need to handle any size message.

With the buffer size and range of the node, the protocols need to be evaluate on the worst case performance under these constraints as well as what their optimum performances are with the constraints more suited for them.

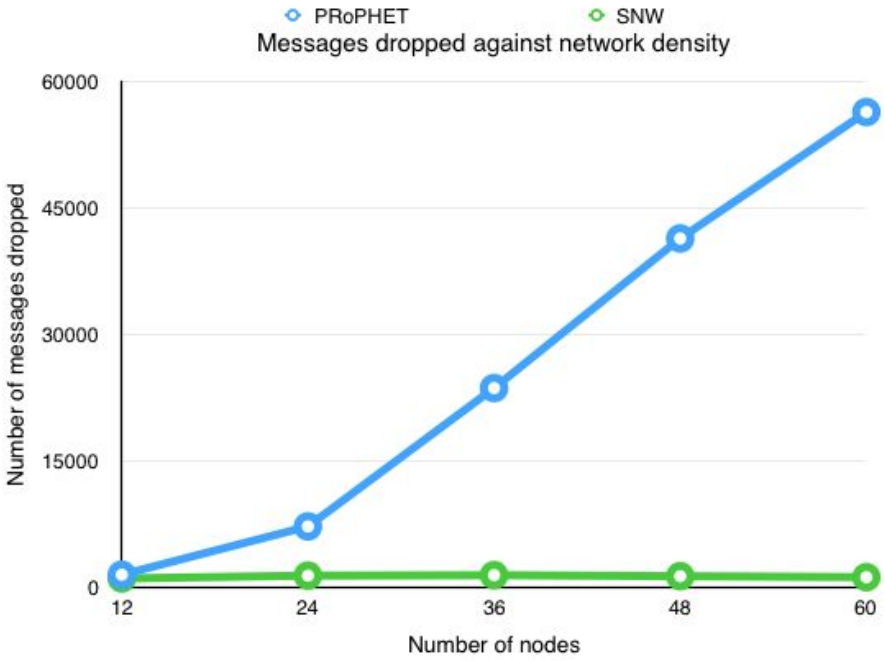
6.Data Analysis



The delivery probability that both PRoPHET and Spray and wait has with relation to the network density follow a similar pattern, where there is an increase in the probability for the messages reaching their destination, this is due to the number of nodes making the network more spread out and with the range that each was allocated it would be quite easy for various nodes to enter their communication range to exchange messages. Both protocols started at approximately the same probabilities however the spray and wait progression increases at a faster rate due to the spraying phase giving the probability an increase as more nodes within the transmitting nodes communication range are receiving a copy of the message, which makes the number of copies in the network larger than at the start. PRoPHET on the other side has quite a low rate of progression only going from 50% to 58% delivery probability this is because being based on the history of node encounters, at the start of the simulation, there are no nodes with a history of encounters so forwarding at the start is slow until nodes start making contact. Finally with both nodes, when the number of nodes in the network goes over 36 then the growth of probability starts to slow in growth in spray and wait, or in PRoPHETs case the probability starts to decrease rapidly. This could be caused for two reasons, the first is that the buffers on the nodes are starting to fill and therefore the number of available nodes is starting to decrease making the route less clear or the density of the network is getting to the point that transmitting nodes are finding it difficult to make contact with the destination node whilst getting contact from other nodes.



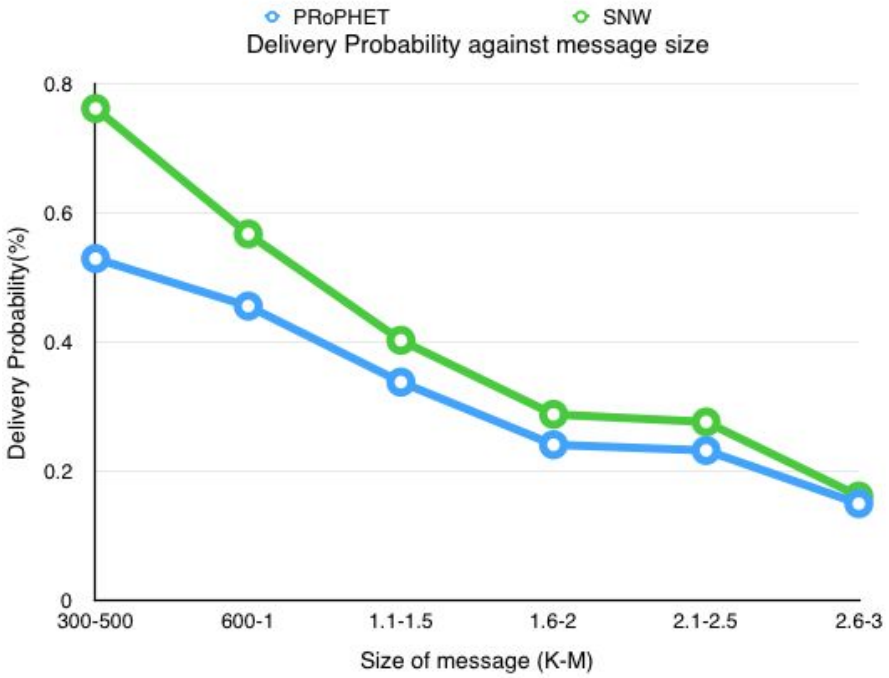
With the latency rate in the network, both protocols start with a high level of latency and then slowly reduce as the network density increases. The reason both protocols have a level of latency with such a small of network of only 12 nodes is it is more difficult for the transmitting nodes and other nodes to discover each other within a set communication range if the number of nodes are so few and spread out over the map, which means there is a low probability of contact with other nodes and therefore it would take longer to forward the message. This is also why PRoPHET and spray and wait have different latency rates as PRoPHET focuses on the encounters of other nodes as a metric to decide whether or not to forward a message, if it has been difficult to make contact with other nodes then the potential relay nodes will be having the same issue where as spray and wait does not use this metric and will just forward the message to any neighbouring node. The latency rate starts to reduce as the number of nodes in the network grows, this is because the probability of contact with potential relay nodes increases and therefore adds to the delivery probability. The reduction rate with spray and wait is quite slow and consistent but this is because the number of messages being sent only increases a small amount, which differs to the PRoPHET reduction rate because the number of messages increases a large amount to accommodate the number of nodes.



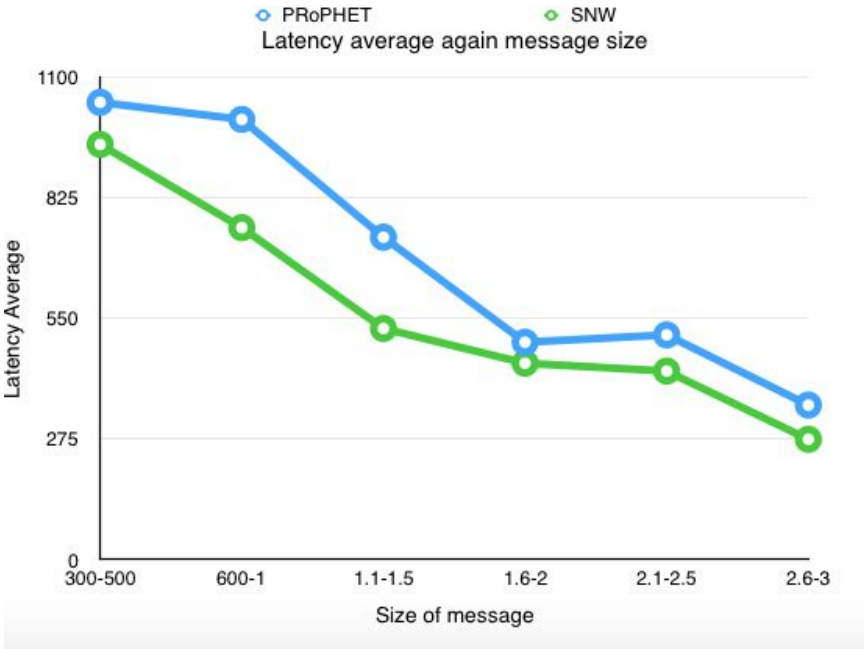
There is a similar reason the number of messages dropped from PRoPHET and spray and wait which is that the number of messages that were sent from PRoPHET were a greater proportion than the number of messages sent from spray and wait, which is why PRoPHET has an exponential increase in the number of messages

dropped. However, spray and wait is very consistent and has a low drop rate. This shows that spray and wait stays conservative with the number of messages and therefore has a low drop rate which is scalable to any network density.

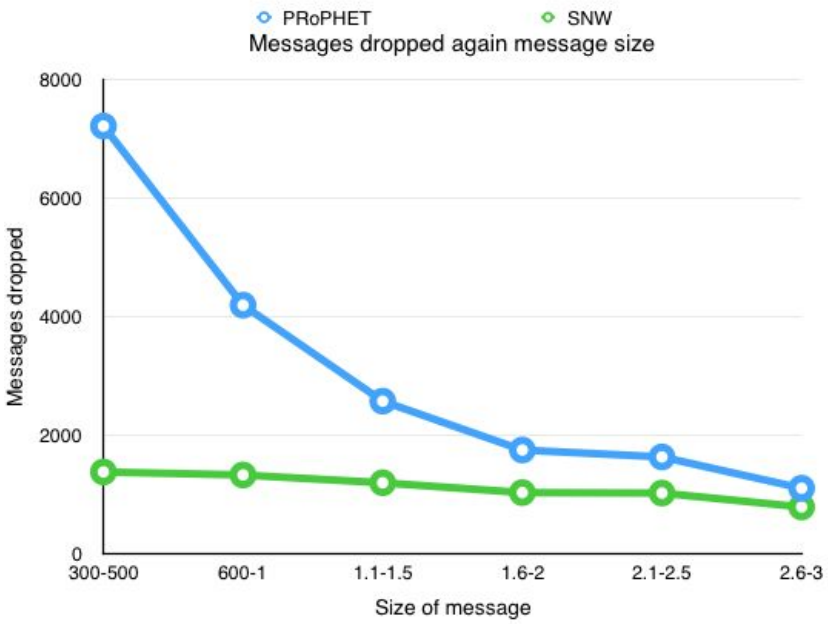
With the delivery probability, the probability seems to decrease as the message size gets larger. With spray and wait; although it starts at a high probability, its seems to decrease at a steeper rate than PRoPHET which has a very consistent reduction rate throughout the scenario. This could be the destination nodes buffer being filled with larger messages and dropping any other message. Both protocols follow a similar pattern and at the largest message size to be tested, both protocols are at approximately the same probability which could be because the buffer size for all nodes is set to 5Mb and with the message size of upto 3Mb, each node can only store 1 message at a time.



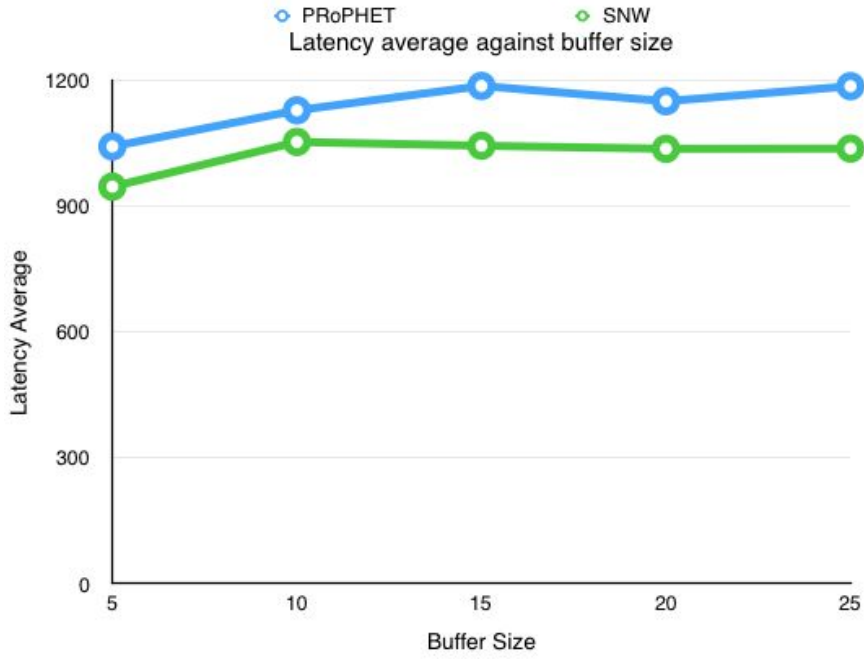
The latency rate also has a correlation similar to that of the delivery probability. The reason for latency decreasing even though the message size would affect it in a negative way is that less messages were sent from each node because of the size of the messages, which meant the network was a lot more conservative with the spread of messages, for example compared to evaluation on the density of the network was significantly less and that means there is less of a delay when transmitting or receiving the message.



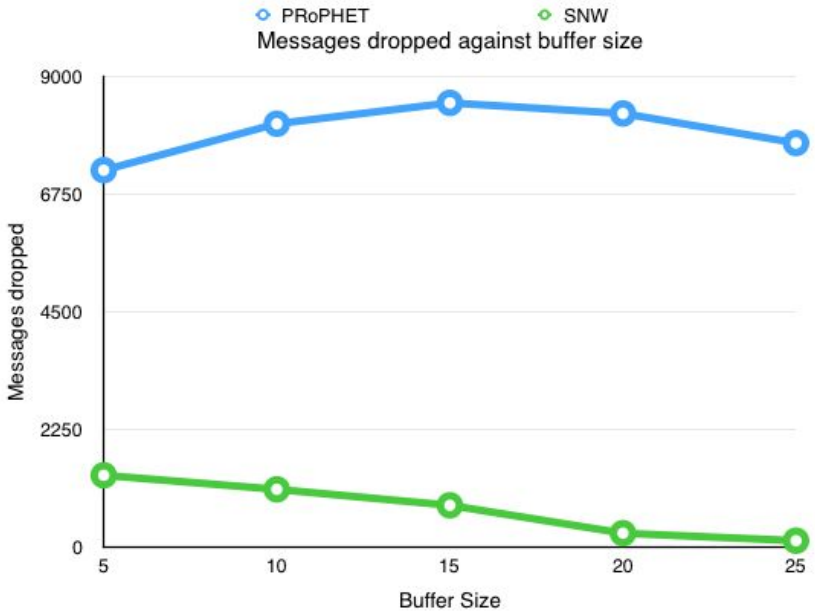
The number of messages dropped concludes two things with the message size metric and that is, firstly spray and wait was not affected in a significant way by the message size variations as the number of messages sent stayed the same and the losses stayed approximately the same, only towards the 3Mb messages did the number of sent messages start to decline. Secondly the message size had a significant impact on PRoPHET as the number of dropped messages declined rapidly however this is also in correlation with the number of messages PRoPHET sent as this started to decrease too.



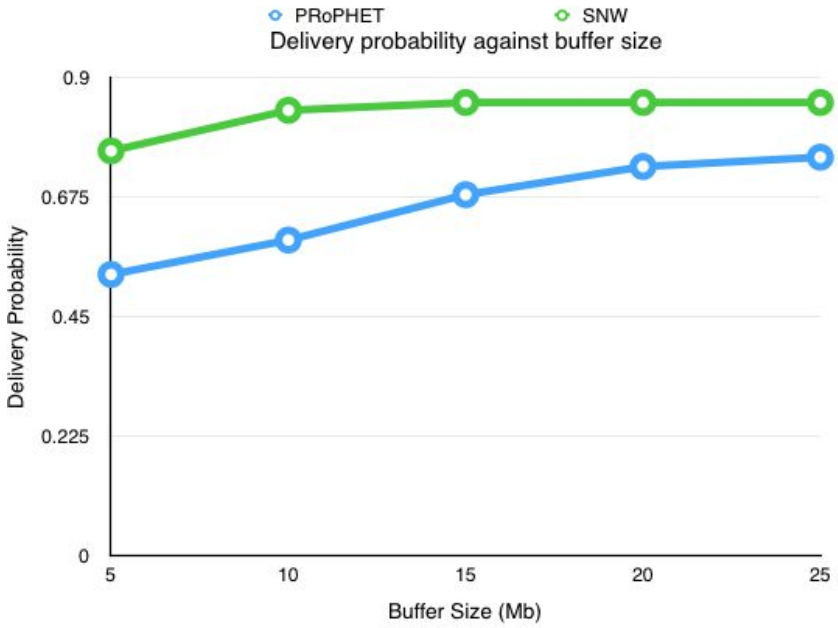
With latency, the buffer size would make a positive impact if the buffer utilisation, which is the buffer size and the amount of the buffer which is occupied. If the size or number of messages were high then the latency would decrease, however as this is an emergency scenario, the message size needs to be kept to a minimum, the cause of this is that both protocols performed poorly. With spray and wait the latency increases a small amount and then keeps to a consistent metric which suggests that spray and wait can handle a variation of buffer size as the latency does not increase any further. PRoPHET on the other hand has a slow increase as the buffer size progresses which implies that more messages are being stored and filling up the buffer causing the delivery to be slow.



The number of messages dropped with the buffer size has very different outcomes from each protocol. For spray and wait the number of dropped messages has decreased which means that the majority of the copies of messages can be stored within the nodes buffer mainly because there is not are large number of copies being sent. PRoPHET on the other hand has an increase in dropped messages which peaks at the 15 Mb buffer, this is because the number of copies grows and fills the buffer. It is not until the buffer size has reached 20mb when the drop rate starts to decrease and the number of messages has less of an impact on the buffer.



Finally the delivery probability of both protocols seems to be positively impacted by the buffer size as the larger the buffer size the high the probability that the message will be delivered. With spray and wait the probability already quite high and the buffer size has less of an effect because the number of copies does not change with the same growth that PRoPHET does. After 10 Mb the grow stops and stays at 85%. With PRoPHET however, there is a large increase mainly because more copies can be generated and stored therefore the probability is higher.



7.Conclusion based on results

Firstly, although the scenario resembles an emergency scenario, the way that a pedestrian is attempting to send a message to a vehicle, there are a few constraints that affect the accuracy of the experiment. These are that with the vehicle movement model in the experiment the movement is random as long as the node follows the road, however emergency vehicles like an ambulance have a set movement model, which is to go to the location of the person that requires assistance and then to drive them to the hospital. Also with the pedestrian node movement model, it is set to short path movement, however this is not a realistic movement compared to real world traces where people has different movement models for various times of the day and whether they are commuting, working or inside a building.

The conclusion that this experiment can gather when identifying the difference with both spray and wait and PRoPHET compared to how they work and process messages, is that PRoPHETs ability to create an unlimited number of message copies means that its message losses will be the highest in most cases compared to spray and wait. This is because the majority of the message copies, once one message has been delivered to its destination the rest of its copies will become redundant and most likely dropped.

Spray and waits message spreading process is quite conservative in the number of copies which were measured. The effect of the messages sizes from protocols were primarily because the buffer size of each node was kept at a constant 5Mb which meant that the buffer was get fuller at a faster rate as the message size increases. However even with this spray and wait handling the size variations with a consistently low message loss.

The constraint that suited PRoPHET the most in the three which were tested was the network density as the delivery probability had grown at a steady rate and the latency rate also dropped which shows that PRoPHET can handle larger sizes of networks even though the message loss grew at an exponential rate.

With spray and wait although it performed well to minimise message loss and latency the strongest constraint it was tested against was the buffer size as the copies which were sent was low, nodes could store the majority of messages within its buffer and therefore drop rate decreases and delivery probability increase however the latency made little if not any movement were PRoPHET has increased each time.

Given this scenario the most suited routing protocol was spray and wait, the delivery probability was high and message loss was low in all cases, with network density and message size these metrics were the most important as they affect the real world with connectivity as they can change at unpredictable times. The consistent performance means that is future scenarios where messages sizes increase or the network contains more nodes, the protocol would be able to deal with problem with little issue.

8.Further Work and Improvements of routing protocols

One aspect of the protocols which was unexpected in that the node buffers did not manage themselves when it came to removing messages or clearing occupied space, in the buffer to add more messages instead filling up the buffer and dropping all other messages that they receive. This often meant that there would be the same message in one node's buffer which clogs up the buffer with unnecessary redundant messages. When looking at the metrics during the experiment; one of the largest statistics was the number of messages dropped however in both protocols in all cases there was not a message which was removed from the buffer.

Some future implementation could have a buffer removal mechanism that would remove duplicate messages this could potential reduce the latency and the amount of message loss. Even in an emergency scenario like this one, messages could be queued on a priority basis and the ones seen as either low priority or not emergency messages could be sent to a different route which may not be as efficient as a route that the high priority takes.

With this there have been two proposed implementations of routing and buffer management which could potentially solve this problem. The first is an implementation of spray and focus however the style of spraying could be implemented with spray and wait which is an ‘binary spraying’ where the message can only be sent to new nodes or nodes that contact the transmitting node for the first time, this implies that no other node that has already made contact can receive the message as they already have a copy.[11]

Secondly in a paper called ‘Buffer Management for Preferential delivery in opportunistic delay tolerant networks’, the paper suggests that most DTN routing protocols assume that the buffer size and bandwidth are infinite which is not the case in the real world and that because nodes are resource limited in their buffer size, bandwidth and movement; a buffer management policy needs to be implemented that prioritises more significant messages like emergency alerts over regular messages such as a chat message. The proposed policy was implemented on the epidemic routing protocol however could be used in a variety of protocols like PRoPHET and spray and wait. The policy splits all messages into three levels of priority (High, Medium and Low priority), these are also make clear within the bundle protocol which is used to send the messages between nodes, the buffer of the node is then split is three queues representing the three priorities, the size of the queue is dynamic to the requirements of the application.[12]

Although this experiment was used as an emergency scenario from the pedestrian needing help to the ambulance. It would also suit a number of other real world situations where messages need to be seen as a priority such as if a vehicle broke down on the motorway that vehicles message would be prioritised more than other vehicles and even in hospital wards where status messages about patients with life threatening illnesses should be prioritised.

9.References

[1] : P. Kissonergis, "Smartphone ownership, usage and penetration by country," 2015. [Online]. Available: http://thehub.smsglobal.com/smartphone-ownership-usage-and-penetration. Accessed: Nov. 27, 2016.

[2] : L. Song and D. F. Kotz, "Evaluating opportunistic routing protocols with large realistic contact traces," *Proceedings of the second workshop on Challenged networks CHANTS - CHANTS '07*, 2007. [Online]. Available: http://www.cs.dartmouth.edu/~dfk/papers/song-dtn.pdf. Accessed: Nov. 27, 2016.

[3] : M. K. Yogi, V. Chinthala, and S. Assistant, "A study of opportunistic networks for efficient ubiquitous computing," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 3, no. 1, 2014. [Online]. Available: http://www.ijarcce.com/upload/2014/january/IJARCCE7B__a_manas_a_study.pdf. Accessed: Nov. 27, 2016.

[4] : M. Faloutsos, S. Krishnamurthy, and J. Eriksson, "Routing Scalability in MANETs," in *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks*. Informa UK, 2005. [Online]. Available: http://www.cs.ucr.edu/~michalis/PAPERS/RoutingScalability.pdf. Accessed: Nov. 27, 2016.

[5] : A. Haris, "A DTN study: Analysis of Implementations and tools," 2010. [Online]. Available: http://etd.dtu.dk/thesis/266537/ep10_65_net.pdf. Accessed: Nov. 27, 2016.

[6] : N. Krystyna, K. Supervisor, E. Lupu, and A. Russo, "User behaviour and security in opportunistic networks," 2009. [Online]. Available: http://www.doc.ic.ac.uk/teaching/distinguished-projects/2009/n.kulesza.pdf. Accessed: Nov. 27, 2016.

[7] : "The ONE,". [Online]. Available: https://www.netlab.tkk.fi/tutkimus/dtn/theone/. Accessed: Nov. 27, 2016.

[8] : A. Martín-Campillo, J. Crowcroft, E. Yoneki, and R. Martí, "Evaluating opportunistic networks in disaster scenarios," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 870–880, 2013. [Online]. Available: http://www.cl.cam.ac.uk/~ey204/pubs/2012_JNCA.pdf. Accessed: Nov. 27, 2016.

[9] : A. Haris, "A DTN study: Analysis of Implementations and tools," 2010. [Online]. Available: http://etd.dtu.dk/thesis/266537/ep10_65_net.pdf. Accessed: Nov. 27, 2016.

[10] : . [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.695.9277&rep=rep1&type=pdf. Accessed: Nov. 27, 2016.
[11] : T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility," *Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerComW'07)*, 2007. [Online]. Available: http://www-bcf.usc.edu/~kpsounis/Papers/sprayfocus_icman.pdf. Accessed: Nov. 27, 2016.
[12] : Fathima, "Buffer management for preferential delivery in opportunistic delay tolerant networks," *International Journal of Wireless & Mobile Networks*, vol. 3, no. 5, pp. 15–28, 2011. [Online]. Available: http://airccse.org/journal/jwmn/1011wmn02.pdf. Accessed: Nov. 27, 2016.
[13] : S. Deok, Chung, Y. Won, and H. P. Corporation, "An improved pRoPHET routing protocol in delay tolerant network," *The Scientific World Journal*, vol. 2015, Jan. 2015. [Online]. Available: https://www.hindawi.com/journals/tswj/2015/623090/. Accessed: Nov. 27, 2016.
[14]:T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait," *Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking - WDTN '05*, 2005. [Online]. Available: http://chants.cs.ucsb.edu/2005/papers/paper-SpyPso.pdf. Accessed: Dec. 12, 2016.

The Effect of Proximity from Source to Destination on the Performance of Epidemic and Spray and Focus

Rory Croucher
psyrjcr

Milena Radenkovic
supervisor

December 12, 2016

Contents

1 Introduction

The aim of this project is to evaluate the characteristics of two different opportunistic routing protocols within a real world scenario. The scenario will be modelled within the simulation software ONE. This document will first give background information on what an opportunistic network and the ONE simulation software is. The two protocols chosen will then be described in detail including how they aim to propagate messages towards the destination. The scenario to be considered will then be explained including a plan of how it will be modelled. The actual implementation of the scenario will be explored and an evaluation of the protocols will be drawn from the results of the simulations. The final step will be to draw conclusions from the evaluation and discuss real world scenarios in which opportunistic networks could be used.

2 Literature Review

2.1 Mobile Ad Hoc Networks (MANETs)

Mobile Ad Hoc Networks [1, 4] are networks made up of mobile nodes with no real infrastructure. To combat the lack of infrastructure, messages are transferred through a series of nodes to reach their destination. Routing in MANETs usually requires the discovery of an end-to-end path before actual data is sent. However, in sparse networks where connections are intermittent, complete paths are unlikely to remain viable for an appropriate amount of time therefore displaying the need for Delay Tolerant Networks (Section 2.3).

2.2 Vehicular Ad Hoc Networks (VANETs)

Vehicular Ad Hoc Networks [9] are a specialised type of MANET in which the nodes are made up of vehicles and fixed infrastructures. These networks need to be able to deal with the constant change in topology due to the fact that vehicles are often in motion. However, unlike the nodes within MANETs, vehicles are much more restricted in the way in which they move as they have to stick to roads and cannot change direction quickly. Lots of types of information can be sent to drivers, for example alerts about accidents which have occurred or updates on road conditions. As with MANETs, VANETs need end-to-end paths to be established before transmission of information can be initiated.

2.3 Delay Tolerant Networks (DTNs)

Delay Tolerant Networks have become necessary due to the mobility or low transmission range of nodes within a network which cause unreliable and unstable connections. DTNs use routing protocols (Section 2.6) to decide how a message should be propagated through the network, without the need of a pre-established path. The protocols use the store and forward paradigm in which messages are stored by a node and moved to certain other nodes depending on how the protocol makes routing decisions. In this way, messages get from source to destination by hopping from one node to another therefore creating an on-the-fly path. The routing protocols can be split into two groups depending on how they make decisions: flooding-based and forwarding-based protocols. Flooding-based protocols replicate a message and spread it across the network to increase probability of delivery. Whereas, Forwarding-based protocols gather heuristics about nodes within a network to figure out which path will most likely lead to the destination the quickest, therefore reducing the amount of resources needed.

2.4 Opportunistic Networks

Opportunistic Networks [5] make use of the different technologies discussed above to allow dynamic transmission over networks with changing topologies. When an opportunistic network is sparse it behaves like a DTN, using routing protocols and multiple hops to deliver messages. However, when an end-to-end path can be established it behaves like a MANET or VANET depending on the type of nodes available. Overall, the behaviour of an opportunistic network is fully based around what is currently possible within the scenario.

2.5 ONE Simulator

ONE [2] is a piece of simulation software written in Java which allows the modelling of opportunistic networks. Different protocols can therefore be compared, contrasted and evaluated within specific scenarios. Models are setup by creating a settings file which allows the manipulation of the simulator. Network nodes are created in groups and have default properties which can be overwritten on an individual basis to alter the behaviour of a particular group. The group router can be set to tell the nodes of the group which routing protocol to use. The movement speed and models of each group can also be altered to differentiate between actors within the scenario. For example, vehicles should be able to go faster than pedestrians and should be limited to moving on roads (which are specified within map files).

ONE provides a Graphical User Interface (GUI) to visualise the simuation as it is in progress. The user can interact with the simulation, such as slowing it down or viewing information about specific nodes within the scenario. ONE also has reporting modules which allows the data collected during the the simulations to be output in a variety of ways. For example, to record the packet transmission statistics including the delivery rate of packets from source to destination. The information obtained in this way is very useful when comparing Delay Tolerant Network protocols within the same scenarios.

2.6 Delay Tolerant Network Protocols

2.6.1 Epidemic

Epidemic [8] is a flooding-based DTN protocol which spreads copies of messages across the whole network to increase the probability that they will reach their destination. Each node has a summary vector which keeps track of the messages that the node is currently storing and a cache to know which hosts it has connected to recently. A connection is made to another node if they have not connected recently. The summary vectors are compared and messages are only transferred between the nodes if the other node does not contain a copy of that message. This method maximises the number of nodes which have a copy of each message whilst also preventing the nodes having multiple copies which would increase strain on the resources available. Many methods can be used to manage the buffers, however the most common is to use first-in-first-out (FIFO), meaning when a buffer becomes full new messages will overwrite the oldest message. Overall, this method uses a brute force approach which relies on the fact that if messages are spread to every node in the network then it will eventually reach its target.

2.6.2 Spray and Focus

Spray and Focus [7] is a two-phased DTN protocol. During the spray phase the source node has a specified number of “forwarding tokens”. Nodes that have forwarding tokens can create and forward

copies of the message being sent. Much like epidemic, each node within the network has a summary vector which keeps track of which messages the node is storing. When a node carrying a message connects with a node which doesn't have a copy, the carrying node creates and transfers a copy as well as half of its forwarding tokens. This allows the message to spread to nodes across the network therefore increasing the chance of it getting closer to its destination. The focus stage of the protocol occurs when a carrying node has only one forwarding token. The focus phase uses single-copy forwarding based on heuristics. The most common of which is using timers to record the time since two nodes have last had contact with one another. The heuristics are used to pass messages to nodes which are most likely to connect with the destination first. Spray and Focus is therefore a much more directed protocol with an intelligent approach to routing.

3 Design

The experiment to be conducted will focus around how protocol performance differs with proximity of source and destination nodes. The scenario to be considered is that of an injured person trying to contact a hospital after having an accident. Epidemic and Spray and Focus will be tested and evaluated to find out which performs better over the different distances by taking into consideration packet losses, delays and delivery ratios. Two scenarios will be set up to determine how the protocols are affected by distance. The first scenario will have a static node on the outskirts of the map and another in the centre, representing the person in need and hospital respectively (Fig. 1). The second scenario will have a similar setup, although the node simulating the injured person will be moved closer to the centre of the city while the hospital node will remain where it is (Fig. 2). Overall, four simulations will be conducted so that each scenario can be run for each protocol.

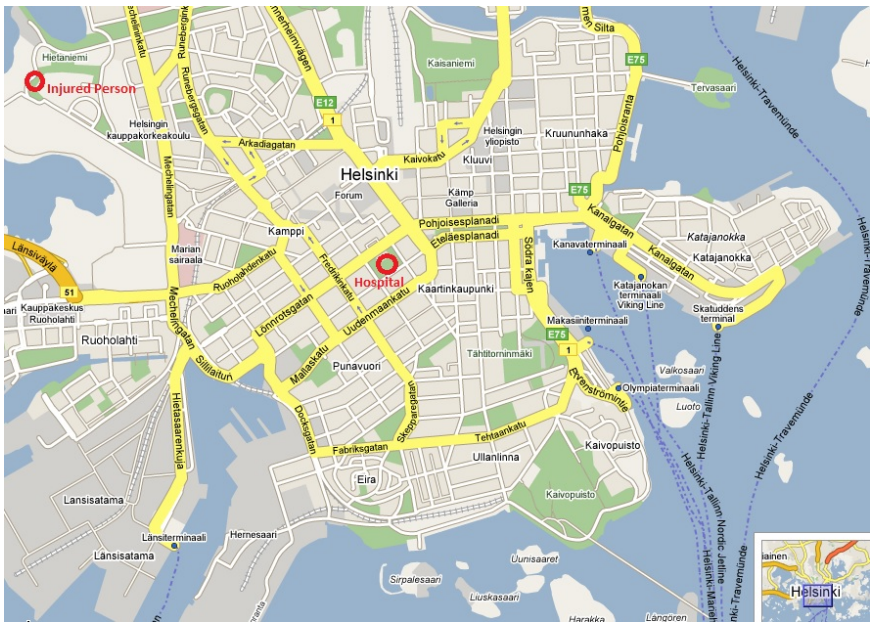


Figure 1: First Scenario with injured person on the outskirts

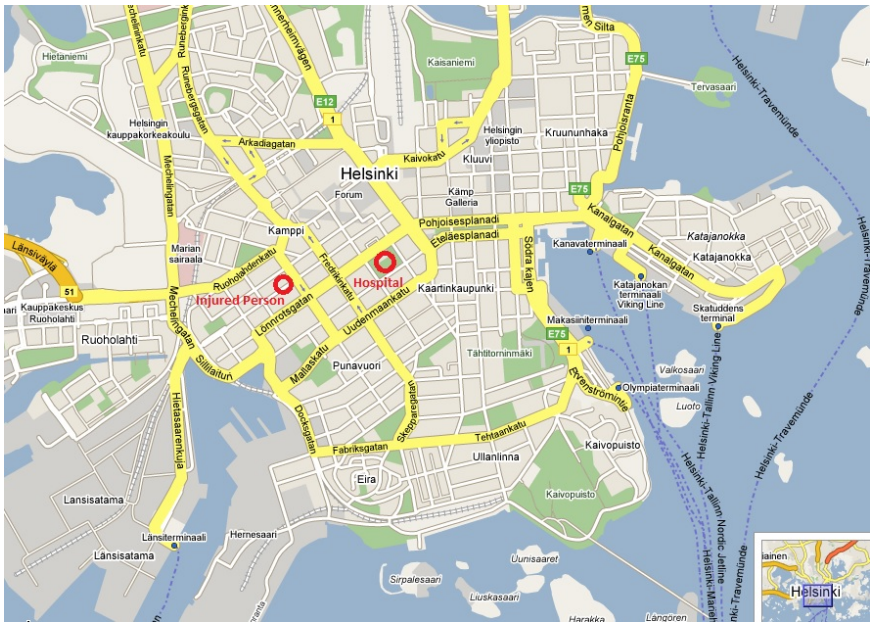


Figure 2: First Scenario with a more central injured person

4 Implementation

To implement the scenarios mentioned above, four settings files were created using the default file provided (one for each simulation to be run). Within the settings files much was kept the same, although two extra node groups were added each containing one node, these were made to represent the injured person and the hospital. As these nodes were to remain in their positions, their movement models were changed from the default to 'StationaryMovement' thereby making them static. The possible source and destination nodes within the settings files were also changed so that the node representing the injured person was the only node sending messages and only the hospital node was accepting them. Through trial and error, the locations of the static nodes within two of the files were moved until they aligned with Figure 1 and the other files with Figure 2. The last step was to change the routing protocols within the files so that one file of each scenario was set to use Epidemic and the others to use Spray and Focus. The tables below show the main settings used within all of the initial simulation files (Table 1) and a visualisation of the details given above (Table 2).

Parameters	Settings
Simulation time	12 hours
Number of nodes	127
Node Types	Pedestrians, Cars and Trams
Message size	500KB - 1MB
Message TTL	5 hours
Interface (pedestrian and cars)	Bluetooth
Buffer size	5MB

Table 1: The common settings used within each of the simulations to be run.

	Scenario 1	Scenario 2
Epidemic	• Filename = EpidemicBT • Group.Router = EpidemicRouter • Injured person set to top left corner of map • Hospital set to centre map	• Filename = EpidemicBT2 • Group.Router = EpidemicRouter • Injured person set to just left of centre • Hospital set to centre map
Spray and Focus	• Filename = SprayAndFocusBT • Group.Router = SprayAndFocusRouter • Injured person set to top left corner of map • Hospital set to centre map	• Filename = SprayAndFocusBT2 • Group.Router = SprayAndFocusRouter • Injured person set to just left of centre • Hospital set to centre map

Table 2: Breakdown of the initial files made.

5 Evaluation

Now that the initial simulation files had been set up, it was time to run the simulations. This section will discuss the results of the four initial simulations mentioned above (Section 4) and provide thoughts behind the results gathered. The changes made to each of the initial settings files, in the hopes of reaching a better outcome, will also be explored.

5.1 Scenario 1 - Injured Person on the Outskirts of the City

Figure 3: First Scenario visualised within the ONE simulator

5.1.1 Epidemic

The results of running the simulation file representing the first scenario using the Epidemic routing protocol (EpidemicBT) can be seen within the table below (Table 3, Column 1). By examining the results, it was found that the delivery rate was particularly low, while the delays and hop counts experienced within the simulation were quite high. It was hypothesised that this may be due to the network being sparse and only making use of one technology (Bluetooth) to spread the message between nodes.

Another settings file (EpidemicWF) was created by duplicating the previous settings used and changing the source and destination nodes to make use of both Bluetooth and Wi-Fi technology. This will allow the source and destination to connect to a wider range of nodes, such as trams, from further distances and increase the spread of the message over the network therefore improving the probability of delivery. The results of this experiment can be seen in Table 3 Column 2. Although the delivery rate did not increase by much, the average latency and hop count has improved more substantially. This experiment was deemed a success due to all the values moving in the correct direction, however it was obvious that the simulation could be improved further as the delays and hop count were still relatively high.

To try to make the experiment more realistic, it was decided that extra mobile destinations should be added to model emergency vehicles which would be found within the real world. To do so, another group was added to the configuration file (EpidemicWFEV), consisting of 5 nodes, which used the same settings as cars and were added to the “toHost” list to mark them as accepting nodes. This new simulation was then executed, the results of which can be seen in Table 3 Column

3. Although this method did greatly improve delays and hop counts, the amount of packets being dropped increased massively. It was thought that this was most likely due to the buffers becoming full and older messages being discarded.

	EpidemicBT	EpidemicWF	EpidemicWFEV
Packets Created	1460	1460	1455
Packets Started	35543	35543	161280
Packets Relayed	16036	16039	139400
Packets Aborted	19503	19502	21879
Packets Dropped	16548	16548	140320
Packets Removed	0	0	0
Packets Delivered	26	29	26
Delivery Probability	0.0178	0.0199	0.0179
Average Latency (ms)	4800.2692	3355.7552	2584.0692
Average Hop Count	7.7692	7.1379	5.4231

Table 3: The effect of changing network interfaces and number of destinations on message statics. Where EpidemicBT was run with static nodes only using Bluetooth, EpidemicWF with Bluetooth and Wi-Fi and EpidemicWFEV with both and extra mobile destinations.

Two more attempts at improving the simulation were made by first doubling (Epidemic-10M) the initial buffer size and then tripling it (Epidemic-15M). As expected, when increasing the buffer size the amount of packets dropped decreases significantly (Fig. 7). As packets are not getting dropped as quickly, more messages are making it to their destination therefore increasing the delivery rate and reducing the number of intermediary nodes needed to deliver a message.

	EpidemicWFEV	Epidemic-10M	Epidemic-15M
Packets Created	1455	1455	1455
Packets Started	161280	75368	25907
Packets Relayed	139400	56090	10203
Packets Aborted	21879	19276	15704
Packets Dropped	140320	56530	10201
Packets Removed	0	0	0
Packets Delivered	26	45	50
Delivery Probability	0.0179	0.0309	0.0344
Average Latency (ms)	2584.0692	3056.7200	2869.4440
Average Hop Count	5.4231	5.6444	5.0400

Table 4: The effect of changing the nodes' buffer size. Where EpidemicWFEV has the default of 5Mb (megabytes), Epidemic-10M has 10Mb and Epidemic-15M has 15Mb.

Overall, the graphs below (Figures 4, 5, 6 and 7) can be used to draw conclusions about the performance of Epidemic within a scenario in which nodes are sparse, the distance between the source and destination is large and a message could mean the difference between life and death. In such a situation, Epidemic is not a viable routing protocol as, even with the improvements made, a message would only be delivered up to 3% of the time. That being said, if such a protocol had to

be used it should make use of Wi-Fi technology and an increased buffer size to allow it to reach its maximum potential.

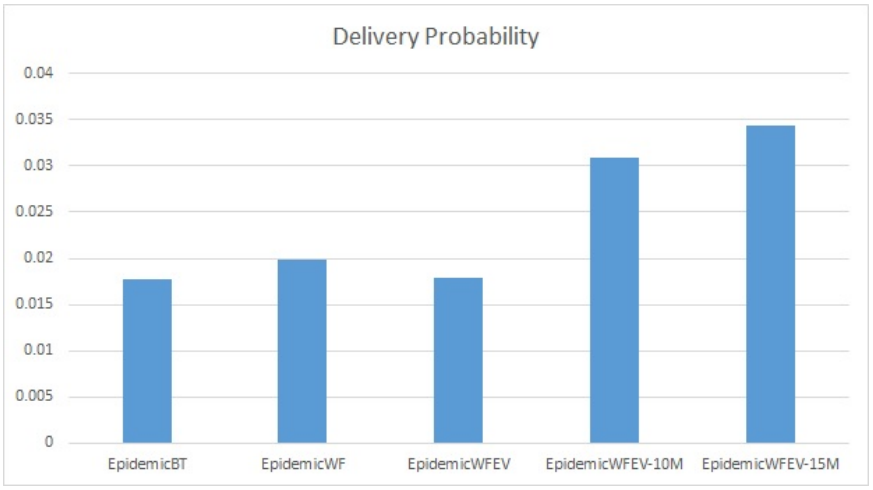


Figure 4: Graph showing how the delivery rate changed over the simulations

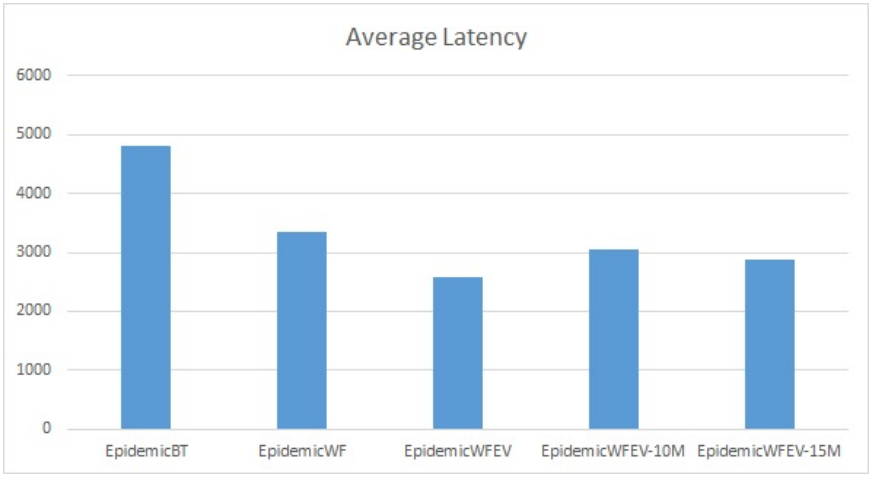


Figure 5: Graph showing how the average latency changed over the simulations

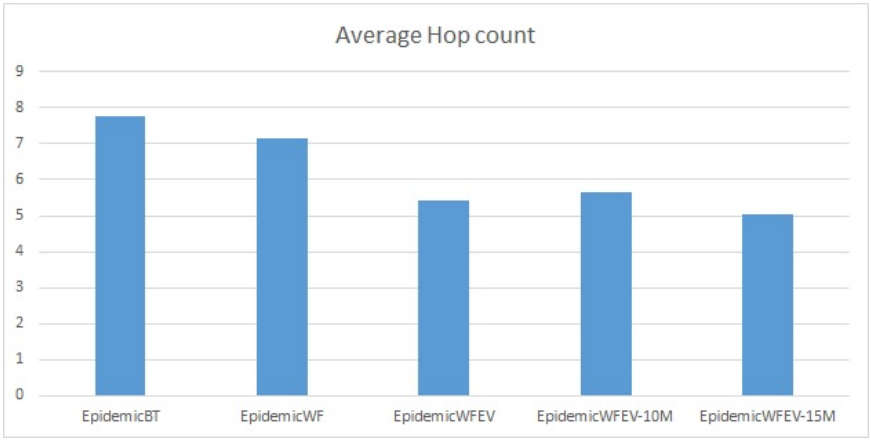


Figure 6: Graph showing how the average hop count changed over the simulations

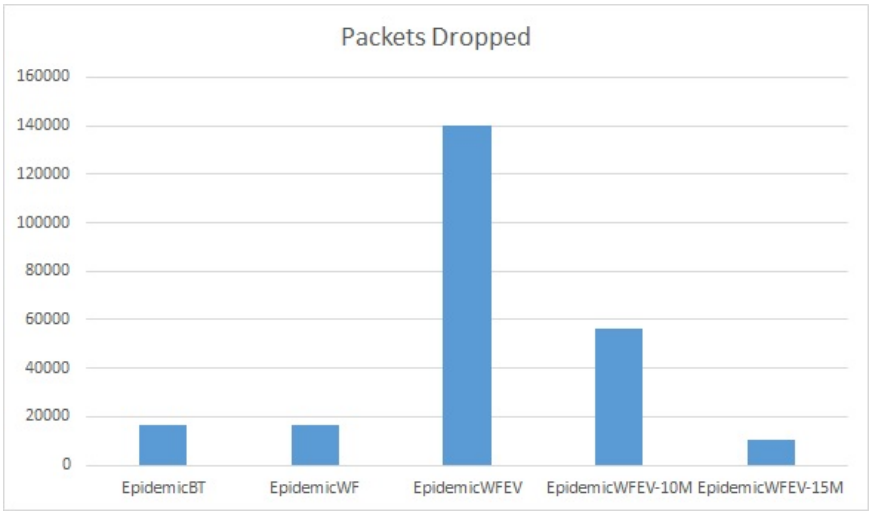


Figure 7: Graph showing how the number of packets dropped changed over the simulations

5.1.2 Spray and Focus

With the first simulation completed and improved, it was time to continue with the same scenario except using the Spray and Focus routing protocol (Section 2.6.2). The results of the experiment can be seen in Table 5 Column 1. Without needing to change the file at all, the performance of Spray and Focus is a colossal improvement over that of Epidemic. The delivery rate is up at 97.6% and the delays and hop count have decreased drastically. However, that does not mean that this can not be improved upon. As such, it was decided that, as with normal experiments, the steps taken to improve the results within the Epidemic simulation should be replicated to see the difference it makes when using Spray and Focus.

As with the Epidemic experiment the Spray and Focus setting file was first changed to make use of Wi-Fi (SprayAndFocusWF) and then also to include five extra emergency vehicle nodes (SprayAndFocusWFEV), the results of which can be seen within Table 5 Column 2 and 3 respectively. The changes seen within the simulation results are very minor compared to those when using Epidemic. However, the delivery rate does improve over the two experiments.

	SprayAndFocusBT	SprayAndFocusWF	SprayAndFocusWFEV
Packets Created	63084	63312	73562
Packets Started	69799	70607	80414
Packets Relayed	64490	65024	74900
Packets Aborted	5307	5582	5514
Packets Dropped	1625	1619	1478
Packets Removed	64209	64738	74761
Packets Delivered	61589	61846	72065
Delivery Probability	0.9763	0.9768	0.9796
Average Latency (ms)	14.2934	22.8447	17.6325
Average Hop Count	1.0020	1.0091	1.0039

Table 5: The effect of changing network interfaces and number of destinations on message statics. Where SprayAndFocusBT was run with static nodes only using Bluetooth, SprayAndFocusWF with Bluetooth and Wi-Fi and SprayAndFocusWFEV with both and extra mobile destinations.

The next step was to see if changing the buffer size of the nodes has any effect on the number of packets dropped within the scenario. The comparison of the results for the three buffer sizes used can be seen within Table 6 and it shows that, like with Epidemic, changing the buffer size does decrease the packets lost but to a lesser extent (Figure 11).

	SprayAndFocusWFEV	SprayAndFocusWFEV-10M	SprayAndFocusWFEV-15M
Packets Created	73562	73562	73562
Packets Started	80414	82329	82554
Packets Relayed	74900	75692	75827
Packets Aborted	5514	6636	6726
Packets Dropped	1478	1451	1441
Packets Removed	74761	75509	75635
Packets Delivered	72065	72078	72062
Delivery Probability	0.9796	0.9798	0.9796
Average Latency (ms)	17.6325	19.0695	16.9419
Average Hop Count	1.0039	1.0059	1.0051

Table 6: The effect of changing the nodes' buffer size. Where SprayAndFocusWFEV has the default of 5Mb (megabytes), SprayAndFocusWFEV-10M has 10Mb and SprayAndFocusWFEV-15M has 15Mb.

Overall, by looking at the results of the initial and final permutations of the simulation, it can be concluded that Spray and Focus fits very well within a sparse opportunistic network with a wide distance between source and destination. The results also allow us to see that it is not worth

installing Wi-Fi technology into the nodes when using this routing protocol within this scenario. This is due to the delivery probability already being at an acceptable level, as well as, the maximum improvement being only 0.3%. Using this technology may also be more expensive to implement and seemingly adds unnecessary delays and transmissions which can be avoided by solely using Bluetooth.

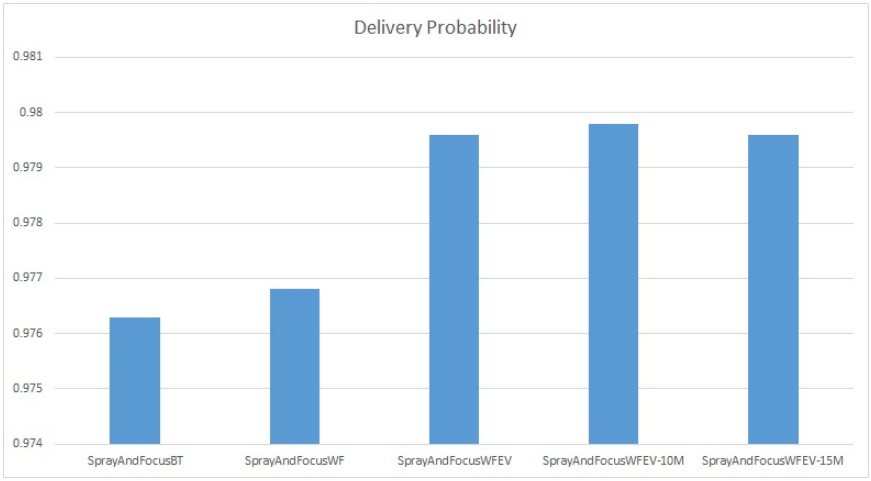


Figure 8: Graph showing how the delivery rate changed over the simulations

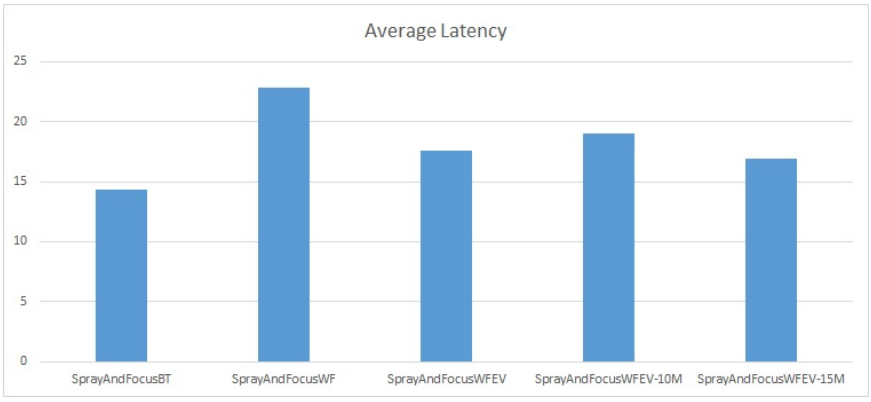


Figure 9: Graph showing how the average latency changed over the simulations

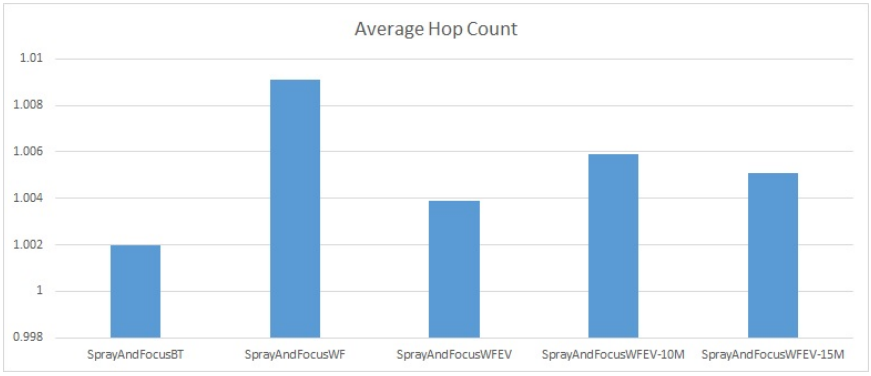


Figure 10: Graph showing how the average hop count changed over the simulations

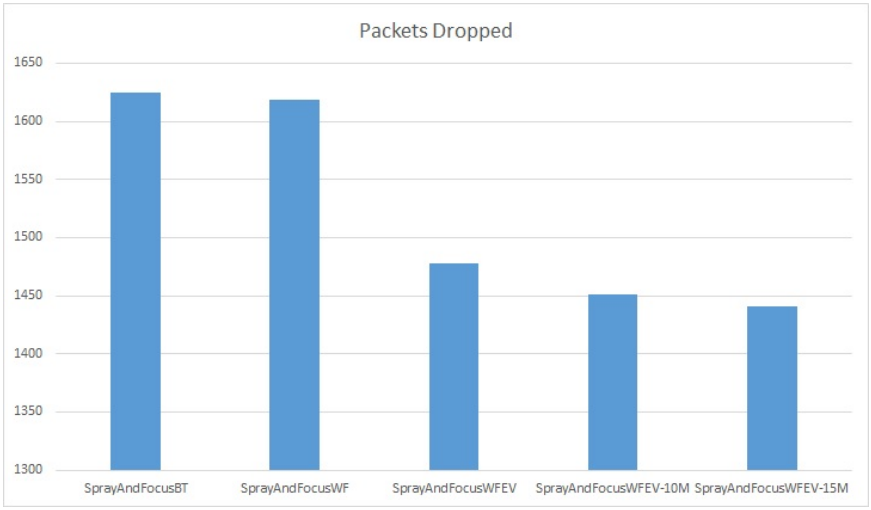


Figure 11: Graph showing how the number of packets dropped changed over the simulations

5.2 Scenario 2 - Injured Person within the City Centre

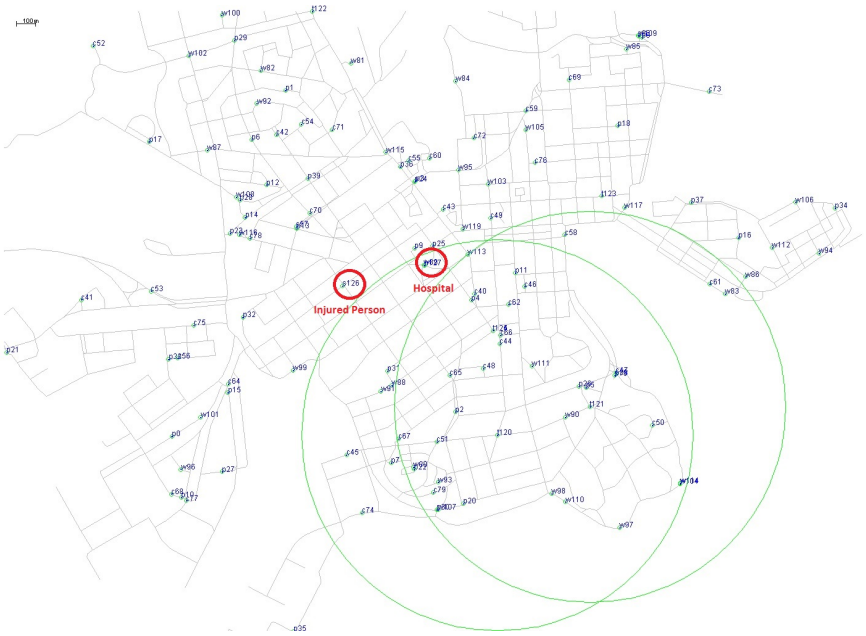


Figure 12: Second Scenario visualised within the ONE simulator

5.2.1 Epidemic

The next stage in the project was to run the initial file for the second scenario using Epidemic routing, the results of which can be seen within Table 7 Column 1. From the results, we can see that there is already an improvement in delivery rate over the previous scenario, showing that Epidemic is more likely to deliver over shorter distances. However, this is not saying much as it is still at a very low 3% and therefore changes should still be made to try to improve this.

As with the previous files (see Section 5.1), Wi-Fi technology (EpidemicWF2) and emergency vehicles (EpidemicWFEV2) were added to try to enhance the results of the simulation. Although improvements in delivery probability, delays and hop count (Figures 13, 14 and 15) were achieved through the use of Wi-Fi, the results took a step back when the extra destinations were added. This is most likely due to the source and destination being within range of each other when only using Wi-Fi, therefore allowing an end-to-end path to be formed. Whereas, when multiple destinations are used the destination that a message needs to be delivered to is randomly selected from the nodes supplied. This means that the other nodes may not always be in range of the source, especially given that the new destinations are mobile.

	EpidemicBT2	EpidemicWF2	EpidemicWFEV2
Packets Created	1460	1460	1455
Packets Started	54449	295472	615920
Packets Relayed	28502	269107	585721
Packets Aborted	25943	26360	30195
Packets Dropped	28950	268855	586011
Packets Removed	0	0	0
Packets Delivered	56	655	242
Delivery Probability	0.0384	0.4486	0.1663
Average Latency (ms)	7136.0696	0.1556	3300.8955
Average Hop Count	9.3750	1.0000	4.4008

Table 7: The effect of changing network interfaces and number of destinations on message statics. Where EpidemicBT2 was run with static nodes only using Bluetooth, EpidemicWF2 with Bluetooth and Wi-Fi and EpidemicWFEV2 with both and extra mobile destinations.

Although the use of Wi-Fi did improve some of the statistics, the amount of packets dropped increased massively (Figure 16). To try and offset the packet loss, the previous method of increasing the buffer size of all the nodes was experimented with. However, although small decreases were found, it was no way near that seen within the previous experiments.

	EpidemicWF2	EpidemicWF2-10M	EpidemicWF2-15M
Packets Created	1460	1460	1460
Packets Started	295472	293366	294652
Packets Relayed	269107	266371	267285
Packets Aborted	26360	26989	27361
Packets Dropped	268855	265485	265427
Packets Removed	0	0	0
Packets Delivered	655	687	690
Delivery Probability	0.4486	0.4705	0.4726
Average Latency (ms)	0.1556	0.2202	0.2113
Average Hop Count	1.0000	1.0000	1.0000

Table 8: The effect of changing the nodes' buffer size. Where EpidemicWF2 has the default of 5Mb (megabytes), EpidemicWF2-10M has 10Mb and EpidemicWF2-15M has 15Mb.

Overall, the performance of Epidemic within this scenario greatly surpassed that of the previous scenario. However, the maximum delivery rate achieved (47.3%) is still relatively low, compared to that seen earlier when using Spray and Focus, therefore it would still not be viable to use Epidemic routing within this situation.

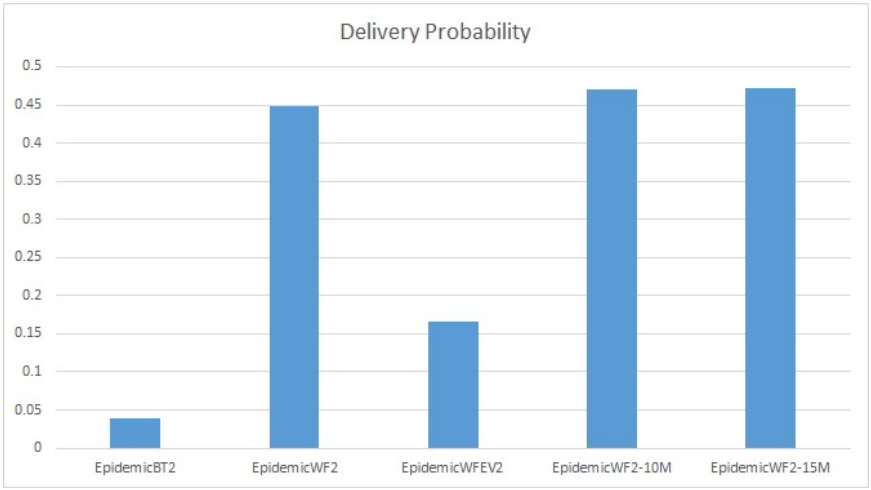


Figure 13: Graph showing how the delivery rate changed over the simulations

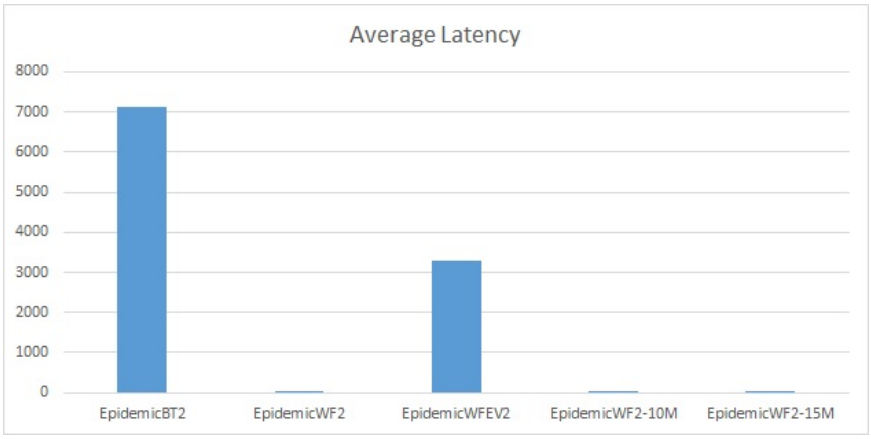


Figure 14: Graph showing how the average latency changed over the simulations

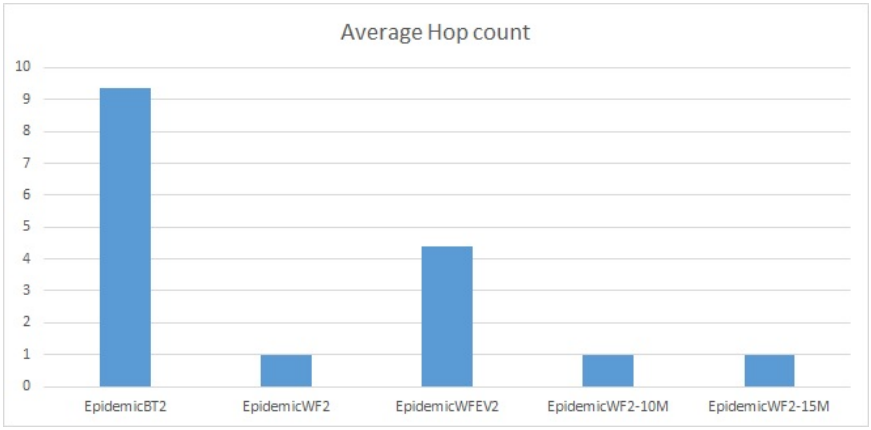


Figure 15: Graph showing how the average hop count changed over the simulations

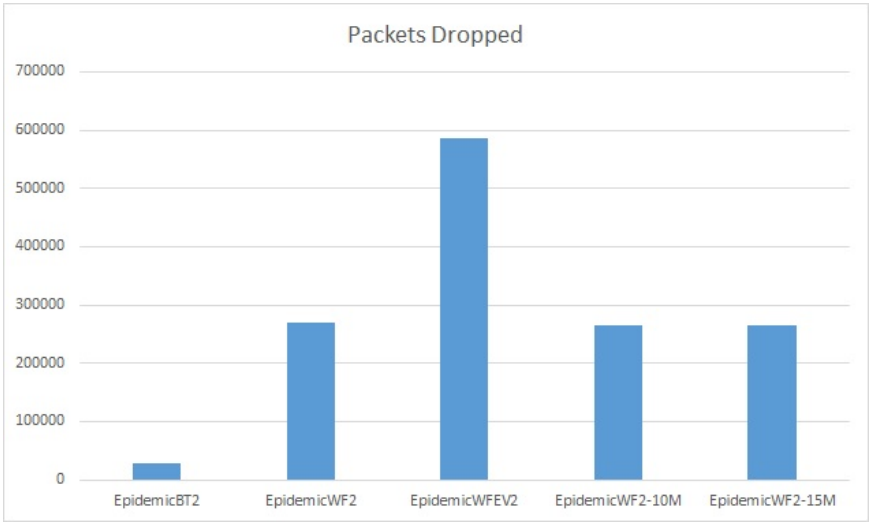


Figure 16: Graph showing how the number of packets dropped changed over the simulations

5.2.2 Spray and Focus

The final step of the project was to run the implementation of the second scenario using Spray and Focus. Again the results showed a massive improvement over those seen when using Epidemic (Table 9 Column 1). However, to keep the experiments consistent the additional steps used previously were also repeated to see how these affected the results, which are shown in the tables below.

	SprayAndFocusBT2	SprayAndFocusWF2	SprayAndFocusWFEV2
Packets Created	63934	64392	74832
Packets Started	78328	77498	95471
Packets Relayed	68397	70897	87110
Packets Aborted	9929	6599	8360
Packets Dropped	2107	3737	4107
Packets Removed	67571	66931	82842
Packets Delivered	62412	64302	74737
Delivery Probability	0.9762	0.9986	0.9987
Average Latency (ms)	31.5293	19.5692	18.4511
Average Hop Count	1.0065	1.0000	1.0000

Table 9: The effect of changing network interfaces and number of destinations on message statics. Where SprayAndFocusBT2 was run with static nodes only using Bluetooth, SprayAndFocusWF2 with Bluetooth and Wi-Fi and SprayAndFocusWFEV2 with both and extra mobile destinations.

	SprayAndFocusWF2	SprayAndFocusWF2-10M	SprayAndFocusWF2-15M
Packets Created	64392	64392	64392
Packets Started	77498	79448	79391
Packets Relayed	70897	71959	71965
Packets Aborted	6599	7487	7424
Packets Dropped	3737	4062	4047
Packets Removed	66931	67584	67588
Packets Delivered	64302	64299	64306
Delivery Probability	0.9986	0.9986	0.9987
Average Latency (ms)	19.5692	20.6143	20.3949
Average Hop Count	1.0000	1.0000	1.0000

Table 10: The effect of changing the nodes' buffer size. Where SprayAndFocusWF2 has the default of 5Mb (megabytes), SprayAndFocusWF2-10M has 10Mb and SprayAndFocusWF2-15M has 15Mb.

Overall, the Spray and Focus protocol performs very well within a scenario in which the source and destination are quite close. Using Wi-Fi within the host nodes improved the results due to the fact that the nodes were then constantly within range of each other and therefore an end-to-end path could be formed. This also explains why increasing the buffer size had little effect on the results, as messages were mostly getting passed along before they were overwritten in the buffers. Therefore, if Spray and Focus were to be used within a real-world scenario of a similar kind it would be advised to use Wi-Fi and a buffer of around 5 Megabytes in size as this is when the balance of resources and performance seems to be at its best.

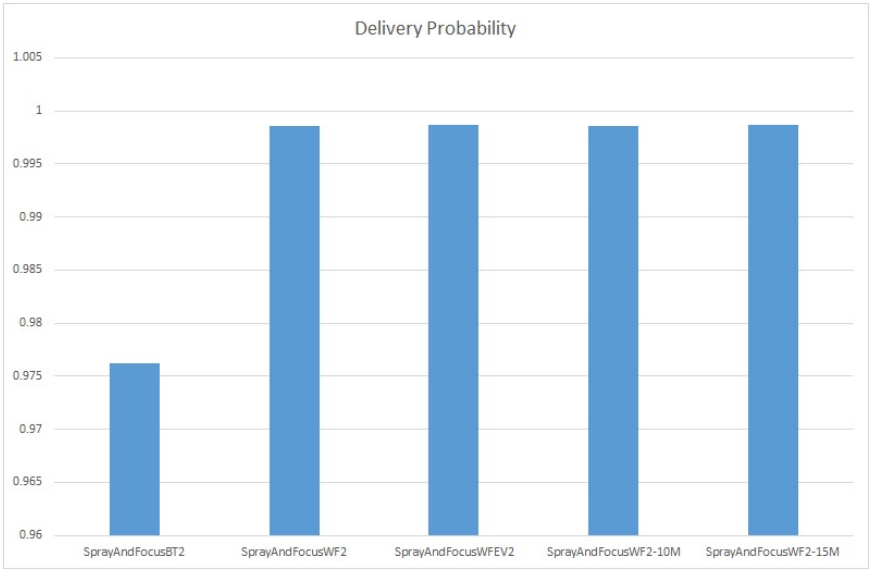


Figure 17: Graph showing how the delivery rate changed over the simulations

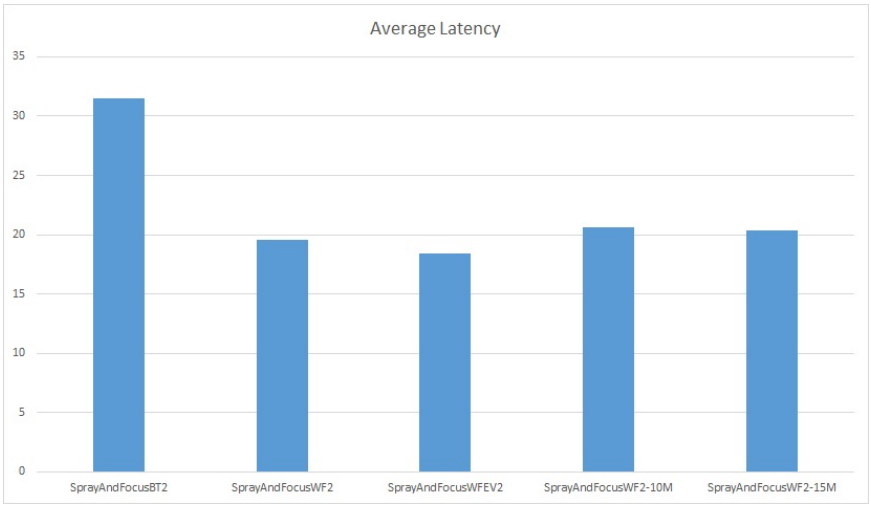


Figure 18: Graph showing how the average latency changed over the simulations

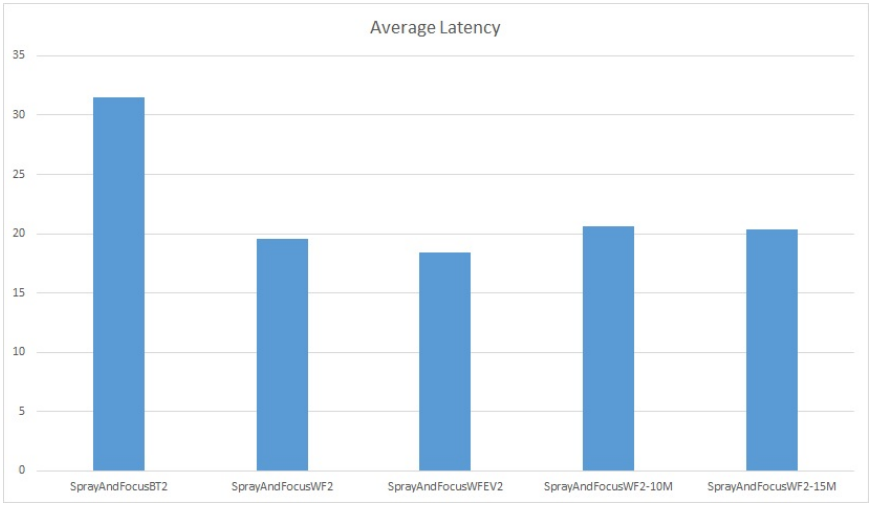


Figure 19: Graph showing how the average hop count changed over the simulations

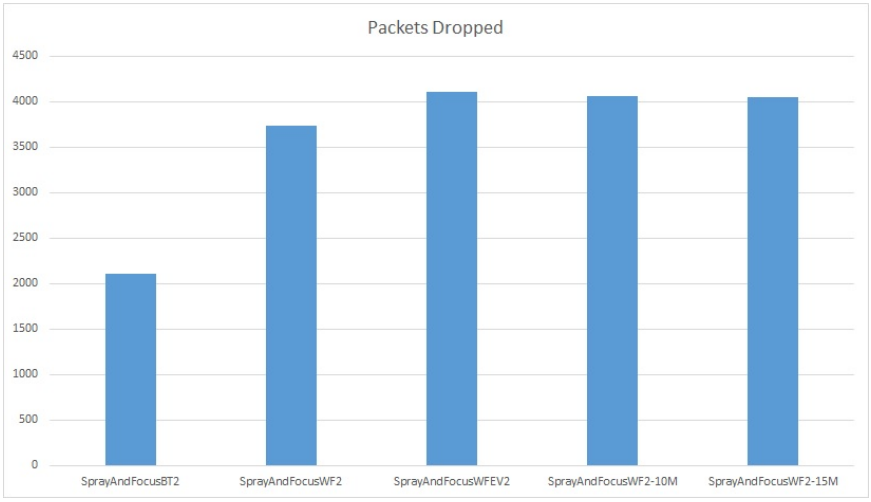


Figure 20: Graph showing how the number of packets dropped changed over the simulations

6 Conclusion

Opportunistic networks were used within this project to provide a way for communication between different types of nodes (such as mobile and vehicular), without the need of a central infrastructure. Due to the sparseness of nodes within the modelled scenarios the opportunistic networks mainly acted as delay tolerant networks, passing messages hop by hop between nodes until they reached their destination. The problem with this is that more resources can be used due to replication of messages and packets are often dropped due to buffers becoming full, which can cause additional

delays. These issues would not occur if using traditional MANETs and VANETs. However, these come with their own problems such as no guarantee that the node will ever be able to form an end-to-end path long enough for transmission to take place.

The performance of the DTN protocols within the opportunistic networks simulated above, especially that of Spray and Focus, show that they are a viable method for sensitive situations in which a high delivery rate is needed. For example, they could be used in disaster situations in which much of the network infrastructure may have been destroyed [3]. However, in these sorts of situations other aspects of the protocols would need to be considered, such as battery efficiency to keep the nodes alive as long as possible. Opportunistic networks could also be used to provide a form of internet access to rural areas, such as that done within the DarkNet project [6]. In this case, an important service, which most of the developed world has access to, can be provided to emergent areas through the use of the aforementioned techniques.

Taking everything into account, the experiments conducted within the opportunistic networks show that both protocols perform better over shorter distances. Spray and Focus, being a more intelligent approach to routing, operated at a much higher level to that of Epidemic during both scenarios. Therefore, this suggests that Spray and Focus should be the preferred protocol within similar emergency scenarios.

7 Bibliography

References

[1] S. Giordano et al. "Mobile ad hoc networks". In: *Handbook of wireless networks and mobile computing* (2002), pp. 325–346.

[2] A. Keränen, J. Ott, and T. Kärkkäinen. "The ONE Simulator for DTN Protocol Evaluation". In: *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*. Rome, Italy: ICST, 2009. ISBN: 978-963-9799-45-5.

[3] A. Martín-Campillo, J. Crowcroft, E. Yoneki, and R. Martí. "Evaluating opportunistic networks in disaster scenarios". In: *Journal of Network and Computer Applications* 36.2 (2013), pp. 870–880.

[4] A. Nasipuri. "Mobile ad hoc networks". In: *Handbook of RF and Wireless Technologies* (2004), p. 59.

[5] L. Pelusi, A. Passarella, and M. Conti. *Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks*. 2006.

[6] A. Pentland, R. Fletcher, and A. Hasson. "Daknet: Rethinking connectivity in developing nations". In: *Computer* 37.1 (2004), pp. 78–83.

[7] T. Spyropoulos, K. Psounis, and C.S. Raghavendra. "Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility". In: *Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops*. 2007.

[8] A. Vahdat and D. Becker. *Epidemic Routing for Partially-Connected Ad Hoc Networks*. 2000.

[9] S. Yousefi, M.S. Mousavi, and M. Fathy. "Vehicular ad hoc networks (VANETs): challenges and perspectives". In: *2006 6th International Conference on ITS Telecommunications*. IEEE. 2006, pp. 761–766.

Performance Characteristics of the Spray and Wait, and Spray and Focus Routing Protocols During an Urban Catastrophe

Natalia Sharon
4204656

G54ACN Report

Table of Contents

1. Introduction

The aim of this project is to compare performances of multiple criteria of two opportunistic routing protocols within The One Simulator for a specific pseudo realistic urban scenario that has been proposed, deployed and built. In this work I will compare multiple criteria simulation results in order to better understand what the most efficient protocol is for sending messages within a VANET is, for a scenario when a disaster has occurred at a certain location, and medical personal and emergency vehicles must be alerted as quickly as possible. This report will provide a detailed description and analysis of opportunistic routing protocols, how and why they can be used in our scenario, and discuss what a simulator can show us about their packet dissemination algorithms. A description of my experiment will be provided, in addition to a full performance evaluation and an analysis of my findings.

2. Scenario

My aim is to simulate two scenarios - one for when a bridge within a city were to collapse in the corner of the city, and one for a main road accident within the centre. These incidents would cause catastrophic repercussions for any person or vehicle on or nearby the bridge during this event. There would be a drastic need for medical personnel and emergency vehicles, such as ambulances and fire brigades. Alerts would need to be sent out at the instance of the disaster, and move throughout the city quickly in order to get the attention of these entities to alert them of what had happened and where, so they can proceed to get to the setting promptly to aid civilians. We will assume a pseudo realistic movement pattern of pedestrians, each having a mobile phone connected to Bluetooth and vehicles (cars, busses, trams) each having Wi-Fi connectivity – therefore being able to send and forward emergency messages and other packets. All of these nodes can communicate in a mobile Ad Hoc manner via different protocols. I have chosen two protocols to propagate messages throughout the network – Spray and Wait, and Spray and Focus as previous research has shown that they outperform other protocols for mobile social networks and for vehicular networks/connected cars [1], [2]. After the experiment, the protocols will be critically compared and assessed regarding delivered messages, dropped messages, and latency (delays) to see which would most efficiently deliver the emergency messages to the required entities. Performance of protocols will be assessed for both incidents to decide which protocol would be suited best for accidents in the centre of the city or in the outskirts.

3. Background

3.1. VANETs

Vehicular Ad-hoc Networks (VANET) are a special type of Ad Hoc Network. They are wireless, self-organised communication networks which enable vehicle-to-vehicle and vehicle-to-infrastructure communications [3]. It is a technology which creates a mobile network, where each vehicle is considered as a free-moving node within the network and information is constantly exchanged between any nodes close enough to form a connection. With the increase in need for constant streams of information to be passed between vehicles on the road – be it for safety reasons such as collision avoidance or accident detection, or simply for internet access - it has become essential to find the optimal and most efficient algorithm to decide the best route to pass data packets through the network. This is in the best interest of all travellers within the network – as could result in an increase in safety, security, comfort and convenience [3]. VANET's assume end to end connectivity/feedback, whereas DTN's do not – it is because of this that VANET's are more vulnerable to congestion and

disconnections than DTN's – and VANET's need to be extended with the DTN paradigm of communications, as explained below.

Ad Hoc Network: a decentralized type of wireless network which does not rely on a pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. [4]

3.2. VDTNs and Opportunistic Routing Protocols

Vehicular Delay Tolerant Networks (VDTN) are inspired by Delay Tolerant Networks (DTN) - and are part of a family of opportunistic, self-organized and autonomous networks [5]. The network lacks consistent network connectivity due to its mobile (and potentially disconnection prone) nature; messages are stored by nodes as data bundles and moved in hops throughout the network until messages reach their destination, as opposed to waiting for the source node to encounter the destination. This increases the probability of message delivery, and opportunistic protocols dictate how nodes decide where to forward the message to next. There is no need to end-to-end connectivity between the source and destination node and thus VDTNs are more resilient to longer congestion or isolation periods.

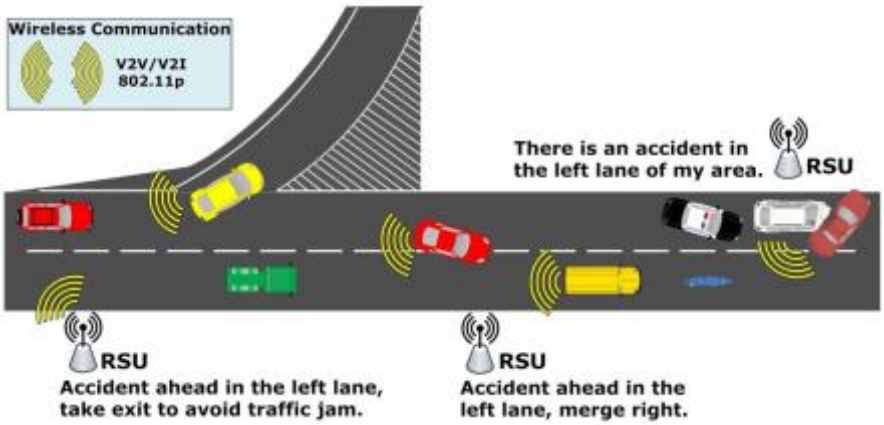


Figure 1 - An example of a VANET where cars send packets to notify other users, source: Spiro Projects [6]

Traditional packet routing approaches include calculating the shortest distance to the destination node, however this assumes that there is a consistent line of connection between the source and destination. In an Ad Hoc Network, connectivity is irregular and unpredictable. DTN network architecture utilises the store-carry-and-forward paradigm [7] which improves performance regarding packet routing [8] and has increased the efficiency, throughput and reliability of the network [9]. A variety of opportunistic routing protocols can be chosen, each using a distinctive algorithm to make a decision on the next node to be selected that would make the most optimal 'next-hop'. Opportunistic routing protocols defer route selection after transmissions – which can increase delivery rate and decrease dropped packets [10].

3.2. ONE Simulator

The Opportunistic Networking Environment (ONE) is a simulation environment which has many features for simulating opportunistic mobile social networks [11]. It includes a street

map of Helsinki within its user interface (please refer to Figure 2 below), and generates node movement based throughout the simulated network. The purpose of this software is to compare routing protocols and to mimic how these protocols respond to specific scenarios within the simulation – hence the best protocol can be determined for a situation. The simulator also enables the designing and building of new protocols and use of different maps and movement patterns. For this project, I will focus on performance evaluation across multiple criteria of two specific protocols in the pseudo realistic scenario that we design and build.

Configuration settings can be altered within files that are passed as parameters when executing the program, and various routing protocols can be set to route messages between nodes within specified scenarios and environments. These parameters may include world size, map type, nodes, network interfaces, movement patterns, event generators, routing algorithms and reports. When the simulation is run, nodes traverse throughout their available pathways, resembling an Ad Hoc Network. Messages are dispersed according to the protocol which has been defined as a parameter – if no protocol has been defined then One uses its default settings.

Analysis of a protocol can be made through examining the statistics report generated by ONE when the simulation has been stopped or is finished. Results displayed in the report include the number of packets created, dropped and delivered, delivery probability and latency average.

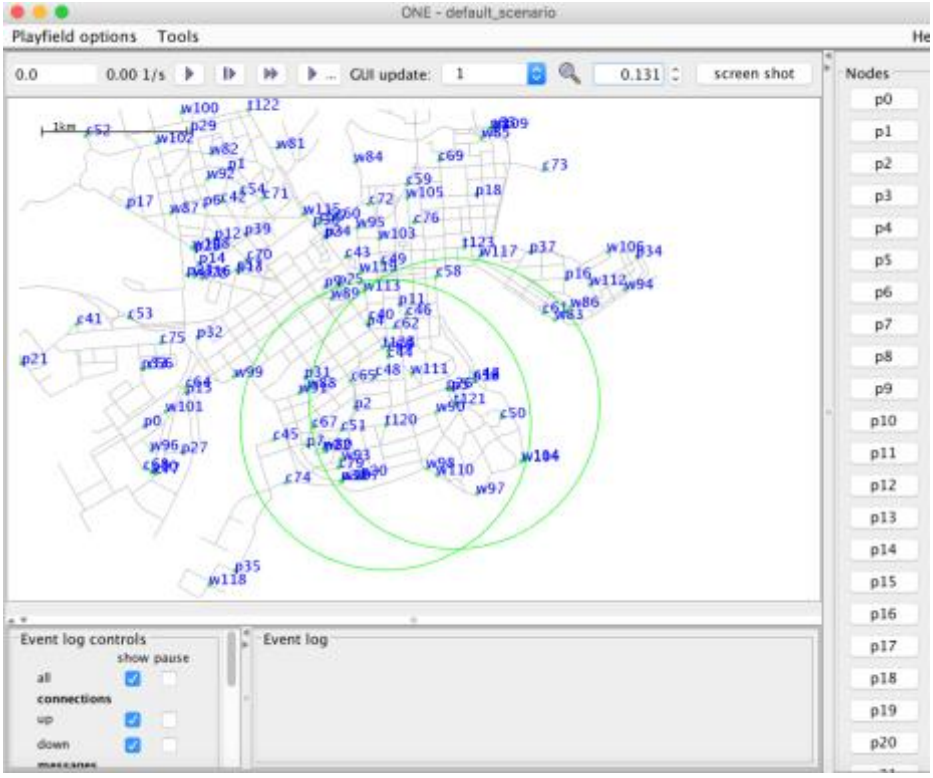


Figure 2 - Screenshot of One Simulator, Using Default Scenario Settings

4. Spray and Wait, and Spray and Focus Routing Protocols

The two protocols I will be using for my experiment are 'Spray and Wait' and 'Spray and Focus'. They are both examples of replication based protocols, however attempt to decrease the overhead produced by more traditional replication based networks [12] such as Epidemic.

Replication based protocols are more appropriate to use for this scenario than single copy protocols as the cost for any single lost message would be extremely high – this scenario relies on working with safety messages which all need to be delivered to the rescue teams. The two protocols are composed of two distinct phases – Spray and Wait/Focus.

Replication Based Protocol: *Intelligently forward a replicated message to another node in the network whilst still retaining own copy. Increases probability of messages reaching their destination and also decreases time of delivery, however results in large overhead regarding buffer space, bandwidth, energy and redundant copies of the message within the network.*

4.1. Routing Protocols and Jon Crowcroft

There are other routing protocols, such as MaxProp, Prophet and Epidemic. Jon Crowcroft used these protocols in a simulation to determine which protocol was the most efficient regarding energy as well as success of delivery specifically within disaster scenarios. I will briefly describe these alternative routing protocols.

MaxProp calculates the probability of future contact with nodes to decide if a message should be forwarded, and maintains a priority queue to discard messages which have little chance of being delivered to its destination, and keeps those which are most likely [13]. Prophet uses an algorithm to decide is each node to deliver a message to the destination, and passes on these calculations to other nodes. Nodes update their probabilities and only forward messages to nodes of higher probability of delivery. Epidemic replicates all its stored messages to all nodes which it comes in contact with. This results in a higher probability of delivering the message as more nodes have a copy of each message but it can also produce network congestion [13]. Mr Crowcroft found that MaxProp had the highest delivery probability, a low amount of overheard, and a low message cost in comparison to the other routing protocols.

I have chosen to concentrate on spray and wait, and spray and focus within this experiment in order to compliment his research analysis.

4.2. Spray and Wait

This protocol places an upper limit on the amount of times a message can be replicated, and hence there is a boundary on the amount of messages dispersed into the network [14]. This reduces wasted resources within the network and there is less overhead of redundant copies. The algorithm consists of two phases – 'Spray' which begins when an initial source node is created and spreads X copies to other nodes. Followed by the 'Wait' phase where the node stops spreading the message, and if the destination is not found during the spraying phase then each of the X nodes carrying a copy of the message will continue to carry until it encounters the destination node and can perform direct transmission [12].

Furthermore, there are two chief versions of how to execute the Spray and Wait protocol – 'Vanilla' and 'Binary'. The two versions only differ throughout the Spray phase, regarding the how packets copies are passed onto encountered nodes [15]. During the Spray phase, Vanilla forwards a single copy of the message to X-1 nodes, and decrements its number of local copies by 1 after each transmission. Once its number of copies reaches 1, the Wait phase is entered. Whereas Binary transfers $X/2$ of its message copies to the receiving node, continuing to do so until it has only one copy remaining, before entering the Wait phase. Again, when its number decrements to 1, it switches to direct transmission. [14]. Within this

experiment, the Binary method will be used as although in reality it is more complex to implement, messages are distributed away from the source node faster, and in a scenario where there is an accident, messages should be dispersed as soon as possible to reach the appropriate personnel sooner.

4.3. Spray and Focus

Spray and Focus works extremely similarly to the Spray and Wait protocol, by distributing a certain number of copies to only a few relays. It attempts to address inefficiency by replicating an allowable number of messages from the source during the spray phase [1]. This leads to the Focus phase, where each node can forward a copy of the message to other potential nodes. The message will only be forwarded if the possible receiving node has a higher potential to deliver the message to the destination node. These calculations are based on timers which record the times nodes come within communication range of each other. [1]

5. Experiment Method

The settings for the ONE simulation have been adjusted accordingly to the specification of my scenario. There are four configuration files – two running the Spray and Wait router, two running the Spray and Focus router, with one from each configured for an accident on the bridge in the corner of the city, and one within the centre. Please see the overlay map below for an illustration of where each accident will occur. Static nodes have been placed at areas of incident to cause congestion and send data out. Default settings were used, with the router type changed and the insertion of extra nodes. All settings - aside from router type and location of static nodes - across the four files are the same – with 35 normal pedestrians, 10 medical personnel, 35 cars, 10 emergency vehicles and 6 trams running through the city.

Below is a full description of the common settings within the files:

Parameters	**Settings**
Simulation Time	4 hours – 14400 seconds
Number of Nodes	96
Node Groups	Pedestrians (35) Cars (35) **Medical Personnel** (10) **Emergency Vehicles** (10) Trams (6) Static Nodes (4)
Number of Hosts (Sources)	4
Number of Destination Nodes	30
Number of Message Copies for Protocol	10
Interface	High Speed and Bluetooth
Mobility Pattern	Map Based Movement of Helsinki, Finland.

Table 1 - Experiment Setup

Figure 3 - Accident Scenarios, Places of Incident within One Simulator

The simulation time will be that of 4 hours, as realistically this is the roughly the window of hours which it may for the city to recover from the accident. The static nodes (of which there are four, to simulate four stationary vehicles) at the place of incident are hosts which initially start relaying messages (the source nodes) and the medical personnel/emergency vehicles are the destinations nodes. Each scenario will run and produce a report. The optimal router will be the one to disseminate the most packets, of which are most successful at notifying the greatest number of medical personnel/emergency vehicles of the accident as possible within the allocated time.

6. Results and Performance Evaluation

Below are the performance metrics of the routing protocols, taken from the reports produced by the experiments within the ONE Simulator. I have compared and evaluated statistics for the success ratio (delivery probability), hop count and latency.

	Spray and Wait		Spray and Focus	
	Bridge	**Center**	**Bridge**	**Center**
Packets Created	490	490	17882	17986
Packets Dropped	773	773	3246	3297
Packets Delivered	70	70	17387	17449
Delivery Probability	0.1429	0.1429	0.9723	0.9701
Latency Average	3320.4400	3320.4400	1.1704	2.9464
Hop Count Medium	3	3	1	1

Table 2 - Statistical Overview of Results

Packets Created: Number of originally created messages
Packets Dropped: Number of messages which have been lost
Packets Delivered: Total number of successfully delivered packets

Delivery Probability: Probability of packet delivery to medical personnel/emergency vehicles
Latency Average: Time it takes for a packet to get from one designated point to another [16]
Hop Count Medium: The medium number of hops to get to the destination.

6.1. Delivery Probability Rate

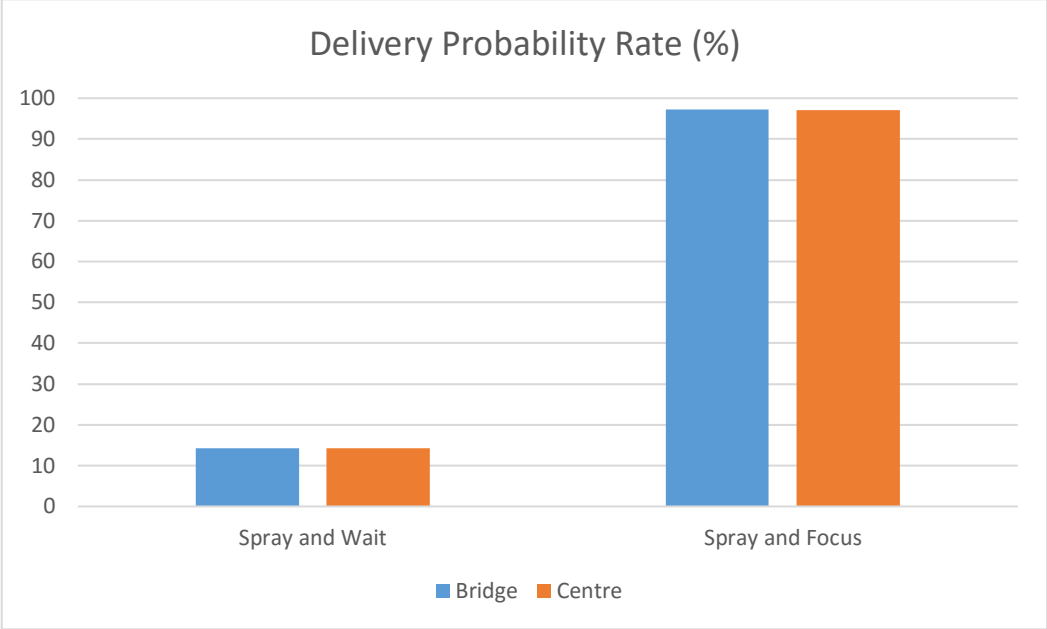


Figure 4 - Delivery Probability Graph Statistics

Figure 4 shows a drastic surge in delivery probability between the two protocols, with spray and focus producing a delivery probability rate roughly 83% higher than that of spray and wait. This is likely due to spray and focus' calculations to only forward the message if the possible receiving node has a higher potential to deliver the message to the destination node. It is also apparent that the location of the accident does not have a heavy effect on the success rate of the routing protocol – as within spray and wait there was no difference in delivery probability for either location, and with spray and focus the difference was negligible. This means that the nodes are able to disperse at the same rate and forward messages to the destination nodes regardless of whether the host nodes are within the centre of corner of the city. Additionally, a contribution to spray and waits low delivery probability, would be due to one of the host nodes unsuccessfully delivering messages within both scenarios. In the Bridge scenario S99 never comes in contact with any node so no messages are ever sent, and in the Centre it comes in contact with only few nodes but never long enough to deliver messages.

Below is a screenshot of the a few lines of the Connectivity ONE Report file and the Delivered Messages Report file generated during the spray and focus bridge scenario. The connectivity report displays in detail every time nodes come in contact, and the delivered messages report provide a complete list of delivered messages and their delivery route.

SNF_Bridge_DeliveredMessagesReport.txt

```
# time  ID  size  hopcount  deliveryTime  fromHost  toHost  remainingTtl  isResponse  path
0.2000 summary44 64 1 0.1000 c38 p19 n/a N c38->p19
0.2000 summary20 136 1 0.1000 t72 t73 n/a N t72->t73
0.2000 summary32 208 1 0.1000 MX82 MX83 n/a N MX82->MX83
0.2000 summary50 64 1 0.1000 p3 p24 n/a N p3->p24
0.2000 summary52 64 1 0.1000 MX77 MX79 n/a N MX77->MX79
0.2000 summary18 208 1 0.1000 t75 t74 n/a N t75->t74
0.2000 summary38 208 1 0.1000 s98 s99 n/a N s98->s99
0.2000 summary0 64 1 0.1000 s96 s97 n/a N s96->s97
0.2000 summary46 64 1 0.1000 p21 EX93 n/a N p21->EX93
0.2000 summary8 64 1 0.1000 t71 MX85 n/a N t71->MX85
0.2000 summary48 64 1 0.1000 MX84 MX81 n/a N MX84->MX81
0.2000 summary6 64 1 0.1000 MX76 MX80 n/a N MX76->MX80
0.3000 summary13 136 1 0.2000 MX76 MX78 n/a N MX76->MX78
0.3000 summary53 64 1 0.2000 MX79 MX77 n/a N MX79->MX77
0.3000 summary1 64 1 0.2000 s97 s96 n/a N s97->s96
0.3000 summary55 216 1 0.2000 t72 t73 n/a N t72->t73
0.3000 summary41 216 1 0.2000 t75 t74 n/a N t75->t74
0.3000 summary47 64 1 0.2000 EX93 p21 n/a N EX93->p21
0.3000 summary9 64 1 0.2000 MX85 t71 n/a N MX85->t71
0.3000 summary33 208 1 0.2000 MX83 MX82 n/a N MX83->MX82
0.3000 summary39 208 1 0.2000 s99 s98 n/a N s99->s98
0.3000 summary45 64 1 0.2000 p19 c38 n/a N p19->c38
0.3000 summary51 64 1 0.2000 p24 p3 n/a N p24->p3
0.4000 summary49 64 1 0.3000 MX81 MX84 n/a N MX81->MX84
0.4000 summary21 136 1 0.3000 t73 t72 n/a N t73->t72
0.4000 summary12 136 1 0.3000 MX78 MX76 n/a N MX78->MX76
0.4000 summary19 64 1 0.3000 t74 t75 n/a N t74->t75
0.4000 summary26 136 1 0.3000 MX83 t71 n/a N MX83->t71
0.4000 summary4 136 1 0.3000 s99 s96 n/a N s99->s96
```

SNF_Bridge_ConnectivityONEReport.txt

```
0.10 CONN 96 97 up
0.10 CONN 97 99 up
0.10 CONN 96 99 up
0.10 CONN 76 80 up
0.10 CONN 71 85 up
0.10 CONN 78 80 up
0.10 CONN 76 78 up
0.10 CONN 72 75 up
0.10 CONN 73 75 up
0.10 CONN 74 75 up
0.10 CONN 72 73 up
0.10 CONN 72 74 up
0.10 CONN 83 85 up
0.10 CONN 71 83 up
0.10 CONN 82 85 up
0.10 CONN 71 82 up
0.10 CONN 82 83 up
0.10 CONN 97 98 up
0.10 CONN 96 98 up
0.10 CONN 98 99 up
0.10 CONN 74 75 up
0.10 CONN 73 74 up
0.10 CONN 19 38 up
0.10 CONN 21 93 up
0.10 CONN 81 84 up
0.10 CONN 3 24 up
0.10 CONN 77 79 up
0.10 CONN 72 73 up
2.70 CONN 21 93 down
14.20 CONN 72 73 down
```

Figure 5 - Report Files

6.2. Hop Count

The hop count refers to the number of devices which the message between the source and destination passed through. Therefore, the medium hop count is the medium number of devices which were used to deliver between source and destination. Referring to figure 5 below, it took a medium of one device between source and destination for spray and focus, and a medium of three devices for spray and wait. It was much more efficient for spray and

focus to find the destination and deliver the packet. Again, the location of the scenario does not have an effect on the hop count.

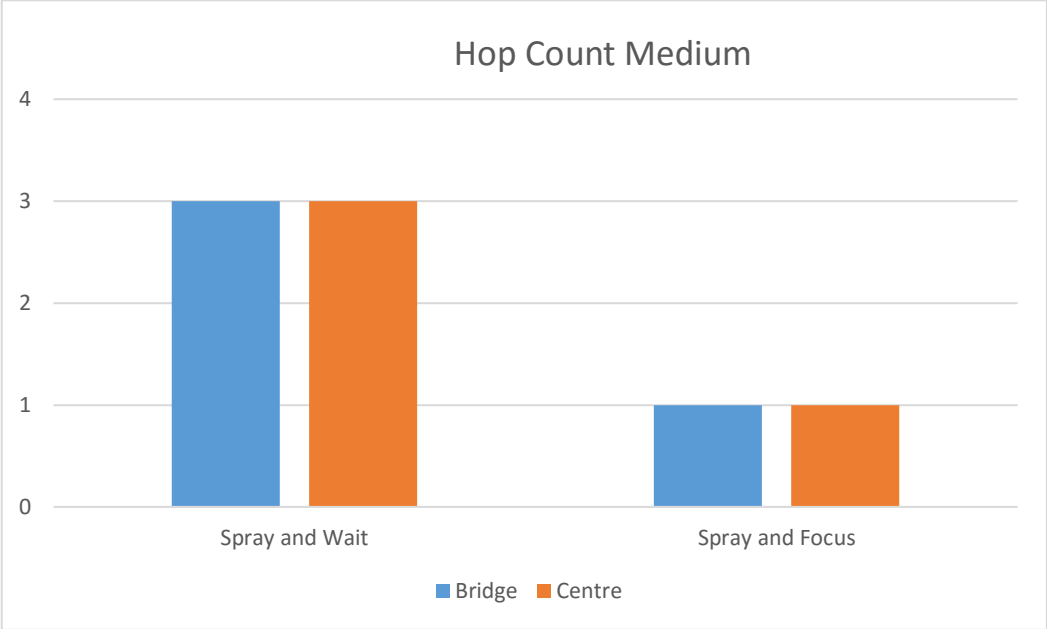


Figure 6 - Number of Hops Graph Statistics

Within this scenario, latency is an important consideration as a delay in packet transfer is a delay in message delivery to medical personnel/emergency vehicles, and in a life or death scenario a delay could mean the loss of someone's life. Below is a comparison of the average latency between routing protocols. The latency in seconds for Spray and Wait is 3.3 seconds, whereas the delay for spray and focus is negligible (0.001 seconds).

6.3. Latency Average

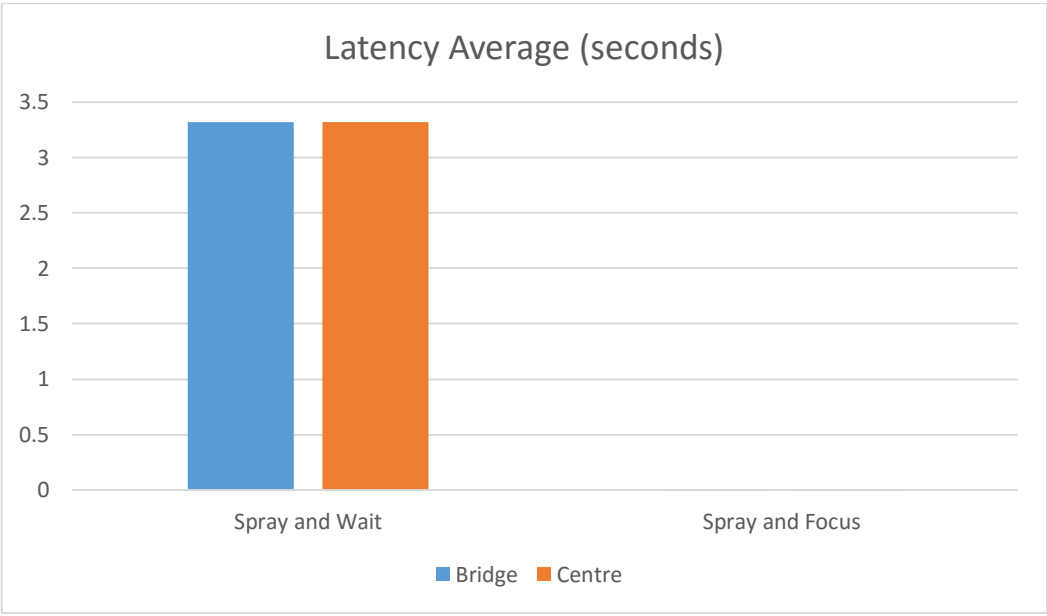


Figure 7 - Latency Average Graph Statistics

There is a large amount of packets dropped for either protocol – specifically from the spray and focus router (3246 from the bridge and 3297 from the centre). However, this router also created 17882 packets for the bridge scenario and 17986 for the centre scenario. Therefore, the reason which has the highest likelihood for causing this significant amount of message drops is that due to the high amount of messages within the network, the buffers of the nodes were highly congested and therefore had to drop the packets. Within this scenario of an accident occurring, a packet drop could signify a loss of life as the message is not delivered. However, spray and focus delivered roughly 17400 packets, whereas spray and wait only delivered 70, therefore it is safe to say that the loss of packets in this scenario does not affect the delivery probability, as the number of packets dropped correlates to the number of packets created, and if this number is incredibly high then it has little impact on the delivery probability. It is easy to see that the most efficient routing protocol for either of these scenarios is spray and focus, and would be the most appropriate routing protocol to be deployed for packet dispersion across a city.

7. Opportunistic Networks Being Deployed in Real World Scenarios

7.1. Opportunistic Networks

With the constantly increasing number of computers and mobile devices which are connected to the internet, implementing Opportunistic Networks would help support a world where we can be constantly surrounded by information. In the case of emergency scenarios (such as the scenario deployed within this experiment), a speedy and coordinated response must be given to improve the efficiency of rescue teams and save as many lives as possible [13]. The use of opportunistic networks is very appropriate for an emergency scenario, as mobile devices carried on by nearby medical personnel can be used to receive messages [13]. The store-carry-forward paradigm allows a dynamic route to be formed, thus minimising the delays and lost packets which a structure-less, intermittent network would produce. In the case of emergencies, it is crucial that the data reaches the destination from the disaster area as any latency or critical data loss could pose a serious risk to people's lives. It's also important to note that different types of emergency vehicles are relevant to different types of incidents. Also, message dispersion from the accident can be arbitrary as node movement cannot be predicted due to every accident being different. Regardless, constant streams of data regarding traffic monitoring could help reduce congestion and increase safety amongst the roads.

The problems introduced by DTN's are generally unpredictable – due to high node mobility and constant possible network partitions – a serious challenge is posed to protocol designers as this is limiting in performance in terms of latency and reliability [17]. Additionally, in scenarios where lots of packets must be dispersed as quickly as possible, for example in that of an emergency, the heavy use of mobile devices as well as overhead from other messages could drain battery power very quickly. If the mobile runs out of battery this is of inconvenience to the user, as well as the node being now redundant and the message unable to leave that node and therefore unable to be delivered. Henceforth It is incredibly important to take into consideration the preservation of battery.

In conclusion, the deployment of opportunistic networks within the real world must still undergo some design – there are many research projects currently underway to find the most optimal protocol in terms of performance (minimal latency and highest delivery probability) and efficiency (minimizing overhead and preserving battery power). One possibility of

increasing chances of message delivery whilst reducing energy usage is to deploy the opportunistic network for use in only emergency situations to minimize congestion and alert medical personal. This would mean it will not be used for passing messages regarding normal traffic congestion or other personal use such as games – thus freeing resources and reducing battery usage, whilst increasing chances of delivery.

7.2. Information-Centric VANET's

It may be more optimal to produce a different architecture to improve the problems discussed above regarding data dissemination addressed in VANET's. Information Centric Networks (ICN's) are network architectures which differ to the traditional host-based communication model and change the focus of content from being host-based to content-based [18]. Essentially, content can be collected from the network, processed in the network and stored in the network. Data is disseminated by being requested by name as opposed to connecting to requesting a host. Any node that overhears the request and has a valid copy of the data is able to respond with the data, regardless of whom the host it. The sent data its signed and optionally secured – the security secures the content rather than the hosts.

A study deployed a simulation to investigate the alternatives of Information-Centric VANETs and showed that the advantage to this over the current architecture is that it compensate for unreliable connectivity and loss rates in VANETs by pervasive caching (it assumes all nodes in the network cache incoming data chunks) [18]. ICN's can run over anything, including the IP stack, therefore it is easily implementable over the current system as well as widely universal.

Bibliography

[1] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Spray and focus: efficient mobility-assisted routing for heterogeneous and correlated mobility," *Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, 2007, PerCom Workshops' 06,* p. 79–85, 2007.

[2] M. Radenkovic and C. Oham , "Congestion Aware Spray and Wait Protocol: A Congestion Control Mechanism For The Vehicular Delay Tolerant Network," *International Journal of Computer Science & Informa tion Technology,* vol. 7, no. 6, 6 December 2015.

[3] "4.2.01. Vehicular ad-hoc network," Compass, [Online]. Available: http://81.47.175.201/compass/index.php?option=com_content&view=article&id=583:4201-vehicular-ad-hoc-network&catid=22:smart-cars.

[4] C. K. Toh, "Ad Hoc Mobile Wireless Networks," *Prentice Hall Publishers,* 2002.

[5] J. Rodrigues, J. Isento, J. Dias, V. Soares, B. Silva, N. Magaia, P. Pereira, A. Casaca, C. Pastor and J. Gallego, "The Vehicular Delay-Tolerant Networks (VDTN) Euro-NF Joint Research Project," 2011.

[6] "What is Vehicular Ad hoc Network (VANET)?," [Online]. Available: http://spiroprojects.com/blog/cat-view-more.php?blogname=What-is-Vehicular-Ad-hoc-Network-(VANET)?&id=29.

[7] V. Soares, F. Farahmand and J. Rodrigues, "VDTNsim: A simulation tool for vehicular delay-tolerant networks," December 2010.

[8] A. Boukerche and A. Darehshoorzadeh, "Opportunistic Routing in Wireless Networks: Models, Algorithms, and Classifications," *ACM Computing Surveys,* vol. 47, no. 2, pp. 1-36, 2014.

[9] P. Jadhav and R. Satao, "A Survey on Opportunistic Routing Protocols for Wireless Sensor Networks," *Procedia Computer Science,* vol. 79, pp. 603-609, 2016.

[10] E. Rozner, "SOAR: Simple Opportunistic Adaptive Routing Protocol for Wireless Mesh Networks," *EEE Transactions on Mobile Computing,* vol. 8, no. 2, pp. 1622-1635.

[11] A. Keränen, J. Ott and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proc. of 2nd International Conference on Simulation Tools and Techniques*, 2009.

[12] T. Spyropoulos, "Efficient Routing in Intermittently Connected Mobile Networks: The Single-Copy," *IEEE/ACM Transactions on Networking,* vol. 16, no. 1, pp. 63-76, 2008.

[13] M.-C. Abraham, J. Crowcroft, E. Yoneki and R. Marti, "Evaluating opportunistic networks in disaster scenarios," *Journal of Network and Computer Applications,* 16 April 2012.

[14] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking,* p. 252–259, 2005.

[15] S. R. Muhammad, F. Guerra, M. Zorzi and P. Casari, "On the Performance of Delay–Tolerant Routing Protocols in Underwater Networks".

[16] "What is latency? - Definition from WhatIs.com," [Online]. Available: http://whatis.techtarget.com/definition/latency.

[17] E. Monticelli, M. Arumaithurai, I. Psaras, X. Fu and K. K. Ramakrishnan, "Combining Opportunistic and Information Centric Networks in Real World Applications," 2015.

[18] Y.-T. Yu and M. Gerla, "Information-Centric VANETs: A Study Of Content Routing Design Alternatives," in *International Conference on Computing, Networking and Communications, Mobile Computing and Vehicle Communications*, 2016.

The University of Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Evaluating Performance Characteristic of Opportunistic Routing Protocols in ONE for the Emergency Scenario of the Collapse of Töölö Towers in Helsinki

Nora Abdullah Almania

Student ID: 4243507

E-mail: psxnaal@nottingham.ac.uk

MSc. In Computer Science

Coursework of Advanced Computer Networks (G54ACN)

Submission Date **12 December 2016**

School of Computer Science

University of Nottingham

Abstract:

Opportunistic Networks (OPPNETs) are a network that lacks persistent an end-to-end path connection amongst the end nodes due to the intermittent connectivity of nodes, restricted power resources and storage spaces. However, this network can overcome the disruption connectivity by following the specific methodology for the nodes which is store and carry the data packets within the communication range. Moreover, disseminate several number copies of the same data packet in order to increase the delivery likelihood. Recently, more than a few routing protocols have evolved especially for DTN which is known as an example of OPPNET. This paper describes OPPNET in detail and discusses the overview and main functions of the Opportunistic Network Environment (ONE) simulator. We focused on two DTN routing protocols Epidemic and Spray and Wait (SnW) and studied the performance of each of them across a range of metrics for a chosen emergency scenario. We designed and set up the experiment of the emergency scenario of the Collapse of Töölö Towers in Helsinki-Finland for our Simulation. Our extensive experiments show that SnW performs better than Epidemic protocol. We also show that our chosen protocols still suffer from some challenges in the network world lie in limited resources and the topology of network.

General Terms: Performance Evaluation, Experimentation, Simulations.

Keywords: Opportunistic Network (OPPNET), Delay Tolerant Network (DTN), Opportunistic Network Environment (ONE), Routing Protocols.

Table of Contents:

List of Figures:

List of Tables:

Introduction:

Nowadays, Opportunistic Networks (OPPNETs) are gaining more attention and becoming with great popularity in the community of research especially with those who are interested in this area of research. The main idea of this kind of network is based on exploiting the exciting of wireless communication abilities of any portable devices in order to build a large system of networking. Users who are carrying these devices with enabled a wireless would come inside the communication range, the devices will artlessly discover each other. Therefore, the devices will exchange the data and information which is stored in these devices and all of that will happen according to the desires or preferences of users [2].

This paper is organized to illustrate the general concept of opportunistic network and to explain the nature of its nodes and the types of applications. This research gives an overview of ONE simulator which is considered as a java-based tool that affords capabilities of DTN routing protocol simulation in a simple framework. It will focus on evaluation the performance of two DTN protocols which are Epidemic and Spray and Wait (SnW) in a frame of chosen scenario that is designed by the researcher with defined parameters. After running the setting of the scenario in ONE, this paper will indicate to some statistics and numbers as result by using some graphs. The final section in this paper will provide advantages and disadvantages sides of OPPNET and some challenges might be faced.

1 Opportunistic Network (OPPNET)

The opportunistic network changed the conventional concept of network. Nodes in that concept are positioned together by using end -to- end path for completing message forwarding. On the contrary, in the opportunistic network, the message is forwarded from the source node to the destination node without a steady path due to the full freedom mobility of the nodes in this network [4].

1.1 Opportunistic Network Definition

The opportunistic network in the communication world is defined as a network of nodes that are connected by using wireless concept. Nodes in this kind of networks could be either fixed or mobile. The distance between two nodes that are connected inside the communication range is estimated approximately among 100 to 300 meters. The topology of this network might be changed due to many reasons such the mobility of the node itself, the activation or deactivation of the node. The nodes are characterized in this network by providing several functions including the following**:**

- **The detection of node:** inside the direct communication range, a node can detect other network nodes.
- **One-hop interchange of message:** inside the direct communication range, a node can send and receive a random message to or from other network nodes.

Yogi, M. Kumar and chinthala, V. [10] said that inside the opportunistic networks there is an absence of relation among the nodes and there is no common aim between the nodes as well, because of that they merely support the one-hop interchange of the message.

This description above refers that the nodes 'mobile nodes' or 'devices' inside the opportunistic network have an opportunity to identify other nodes in the communication range and likewise to talk to them.

1.2 Opportunistic Network Nodes

The nodes in the opportunistic network are considered as devices with a short range wireless communication abilities. These devices that operate an opportunistic network application normally uses protocol for data dissemination and message interchange machinery. The nodes of an opportunistic network could be mobile devices which are carried by persons or fixed devices such as Information Sprinkler (IS).

1.2.1 Mobile Node Definition

A mobile node is defined as a mobile device that is carried by a user, for example, mobile phone, laptop or handheld device. This device acts as an opportunistic network node [10]. In addition, this mobile node or device is referred as an internet connected device which has site and point of correlation to the internet might change repeatedly.

1.2.2 Information Sprinkler (IS) Definition

An information sprinkler (IS) is defined as a fixed and static opportunistic network node inside the network. This kind of node is a device which is located at a dedicated place. The most important characteristics of this type of node are first, it is not mobile, it is fixed and secondly, it is not under the control of the user. Furthermore, IS uses the identical data distribution protocol same as other nodes within an opportunistic network. In addition, IS can also collect data from users and pass this data to other users. As an extension to this concept, information sprinklers that are located at different locations could be connected as well by using backbone link [2].

1.3 Opportunistic Network Application

The applications of the opportunistic network are divided into two general terms:

1- Active Collaboration (AC) 'Physical user interaction'.

2- Passive Collaboration (PC) 'Multi-hop information dissemination'.

1.3.1 Active Collaboration (AC)

An active collaboration utilizes the physical vicinity of users. Additionally, it is effective to exchange digital data and information with near users. For example, of that through the user notification the vibration of dexterous devices, users can directly be aware of each other. This way might lead to face-to-face collaboration such as making conversation or keeping track of a common goal in the real world.

1.3.2 Passive Collaboration (PC)

Within communication range, a passive collaboration gathers and passes data and information from and to other users. It is called passive due to that happens without any interaction from users. The mechanism of dissemination data within PC is an autonomous dissemination.

2 The ONE Simulator

Opportunistic Network Environment is defined as an engine of discrete event simulation. The role of this engine is to update many of modules which implement the main functions at each step of the simulation.

2.1 Overview of the ONE Simulator

The ONE simulator provides a good environment for simulation due to it has main functions which are contact of inter node, node movement modeling, routing modeling and message handling. In addition, the result after the simulation is collected and analyzed via reports, visualization and the tools of post- processing. As shown in Figure 1 ONE simulator has five basic components which are movement models, event generators, routing models, simulation engine, reports and visualization [3].

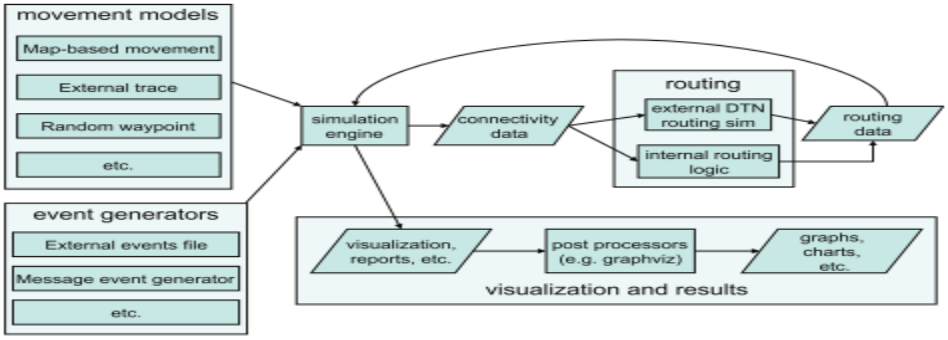


Figure 1: Overview of the ONE Simulator

The movement of node is employed by movement models. The communication of the nodes with each other is based on their range of communication, their location and the bit rate. Simply, the function of routing which is employed by routing modules, is to decide which messages that wanted to forward over the available contacts. The role of events generator is to generate the messages themselves. Inside the world of simulation, the messages are always having a single source and destination. During the simulation run, the results of that simulation are mainly collected by reports generator. The role of the report module is to receive the message the events generator and then generate the result based on that message. Finally, the Graphical User Interface (GUI), Figure 2 depicts this interface which shows a visualization of the state of simulation that showing the locations, contacts and messages carried by nodes and the benefit of that is to give the user more sense and realistic of visualization [3].

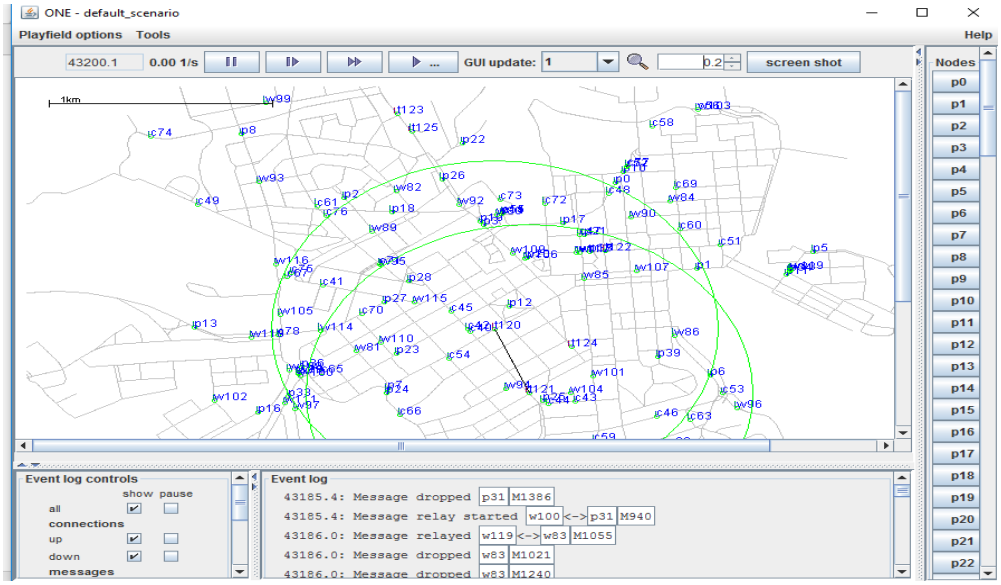


Figure 2: The Graphical User Interface (GUI) of the ONE Simulator

2.1.1 Node Capabilities

The main factors in the simulator are called nodes. The model of node is a mobile endpoint which capable of representing as a store – carry forward router. For example of these nodes are cars, trams, buses, trains and pedestrian with the hardware requirement.

In the simulation world, groups of nodes are created the simulation scenarios. Each group in the simulation is configured with different node capabilities and each node has a different set of primary capabilities.

These capabilities are movement, radio interface, energy consumption, persistent storage and message routing. The capabilities of the nodes such as persistent storage and radio interface are configured via parametrization for example, the communication range, storage capacity and bit-rate. The other complex capabilities such as routing model and movement model are configured via dedicated modules which perform a specific performance for the capability [3].

2.1.2 Movement Modules

The mobility paths of node are implemented by opportunistic network environment via the mobility models which already exists in the simulator or by loading the external movement data. The most popular two movement models are Random Walk (RW) and Random Waypoint (RWP) were involved in ONE. The reason for their popularity due to the simplicity however,

both have various of limitations. The movement of these models is in a random trend and unlimited manner.

ONE can provide a real-world mobility to make the performance of DTN routing protocols more realistic by restricting the mobility of node to predetermined paths. These movements are called The Map-Based Model (MBM), Shortest Path Map-Based Model (SPMBM) and Routed Map-Based model (RMBM). Moreover, the ONE contains the Helsinki town map data like the roads and the pedestrian sidewalks which the MBM can use.

The MBM is considered as the easiest and simplest model of movement that applied in ONE. In MBM the nodes are moving randomly, however, they always follow the predetermined paths by the map data. In addition, these nodes might not be very exact and accurate as the real mobility of human. The SPMBM is considered as an advancement of MBM and is more realistic than MBM, the nodes in this movement model choose random points instead of entirely random walks on the map and then follow the shortest path to reach that point which was selected by them from their current location. These random points might be selected entirely randomly or maybe from a list of Point of Interest (POI). Finally, the RMBM model, nodes in this model have predetermined paths or routes that they follow it. RMBM is very valuable and beneficial for simulating nodes like buses, trams, and trains.

2.1.3 Routing Modules

The capability of routing message is employed likewise to the capability of movement. The ONE simulator comes with a basic framework for defining the rules and algorithms in routing and with prepared applications of very known DTN routing protocols. The ONE provides six popular of DTN routing protocols which are called: Direct Delivery (DD), First Contact (FC), Spray and Wait (SnW), Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET), MaxProp and Epidemic [3]. These routing protocols conduct similar by following the same process which by permit the contact nodes to deal with message to exchange it until the message reach to the destination. Nevertheless, these routing protocols conduct in a different way when it deals with forwarding and replication the messages.

2.1.4 Events Generation

Any events for the simulation are generated by event generators. The ONE has two different methods to import the events into it. Firstly, trace files which is defined as a simple text file which has timestamped of events for example, establishing a new link between nodes, creating message or removing message from the message storage location which is named the buffer. Secondly, event generator modules which is considered as normal java classes

that can exactly generate the same events as the trace files. Whereas the trace files are generally created with a special script or by transformed from the other output of program, the event generator modules can be configured by using the setting of the simulator. The basic feature of the ONE simulator is supporting several simultaneous event generators and the events itself are mechanically inserted inside it. Whereas the external events offer input for the simulation in the other side the output is created by the report modules.

2.1.5 Report Modules

The role of report modules is to write the information that related to the event which will be existing in the output file, or to store that information into an internal data structure for producing a summary after the simulation running is completed and finished. Several reporting modules are already available in ONE simulator such as message delivery probability, average of latency 'delays' and message dropped. The ONE simulator uses a *MessageStatsReport* file to store the result for each simulation inside the reports folder that holds a summary of message situations such generated, dropped, delivered etc. Furthermore, this report is valuable to use it for evaluating the performance characteristics of opportunistic routing protocols in DTN.

2.2 Running Simulations

The ONE simulator provides two modes for running the simulation which are: The Graphical User Interface GUI mode and the batch mode. The GUI is particularly valuable for testing, processing and demonstration purpose. Another mode which is called the batch mode, this is very beneficial for running large volume of simulations with dissimilar collection of parameters. These two modes create reports which could be analyzed by post processing tools to produce a variety set of summaries as graphs or plots.

2.2.1 GUI mode

The simulation inside Graphical User Interface (GUI) mode is visualized in a real time. The GUI mode of the ONE simulation consists of three fundamental sections. The biggest section is the playfield of the geographical simulation area. In the playfield, the location of nodes, the nodes radio range, the paths and the number of messages are clearly visualized. Inside the playfield there are some buttons which allow the user who use the simulator for controlling the simulation for example, play button  , pause button  , step forward button , a zoom button  , a drop-down menu for many selections of GUI update speed and button for screenshot capture. In the middle of GUI model exactly beneath these buttons the

geographical simulation area is existing. The role of this area is to show the nodes locations on the map, the range of communication, the movement and the number of messages. The event log section which keeps follow and track the events then displays and shows them in the event log panel. In addition, it gives more information about the nodes or messages. The list of the nodes on the right side of the GUI mode of simulation. Nodes in this list are assigned by the first letter of the group names such c=car, p=pedestrian etc. The number of the nodes in this list is the same number on the setting in the simulator. If the user clicks on any node in the list, the user will get the path of this node in the map.

2.2.2 Batch mode

The batch mode does not have GUI, the running of the simulation through this mode is done by using the command prompt. In addition, the batch mode runs the simulation without the real-time visualization. In the beginning of the simulation, it will show the name of the scenario and the number of the scenarios left in this batch. The result of the simulation is collected from report modules. The most important feature of the batch mode is the running simulation speed which is considered faster than GUI due to it does not produce graphics. In addition, another feature of this mode which is named run indexing this feature allows the easy configuration of settings for various simulation with a same single configuration file.

2.2.3 Post-processing

The essential feature of post-processing in ONE simulator that allows the files of report to be processed by other programs. Two modules of report are used to create the input files for Graphyiz. The first module is called *AdjacencyGraphvizReport* that is used to generate adjacency graphs merely to present nodes which have been related to each other. The second module is called *MessageGraphvizReport* that used to generate graph of delivery path of message.

3 Delay Tolerant Network (DTN)

The general concept of DTN relies on the connected networks that might suffer from frequent segmentations and that might be included one or more than one of different group of protocols or protocols families. The DTN are applied in several functioning environments including the environments of networks that subject to disconnection and disruption cases [9].

3.1 DTN Routing Protocols

The routing protocol of DTN embraces the paradigm of store-carry-and-forward. The main role of this paradigm is based on if there is no node in the communication range then the existing and current node will store and carry the information or data until it encounters another node in the same communication range [5]. This paper aims to evaluate the performance characteristics of merely two assigned DTN routing protocol which are Epidemic and Spray and Wait (Snw). The following section will provide an expanded description of these two protocols.

3.1.1 Epidemic

Epidemic routing protocol is defined as an unlimited-copy routing protocol and utilize flooding-based pattern [5]. Epidemic is considered as the first routing protocol that is suggested for dispersed network. Becker and Vahadat suggested Epidemic protocol [8] to positive influence on performance of message delivery through the interrupted connecting and likewise increasing the ratio of message delivered to the destination. Epidemic offers a unique ID number for each packet that is generated and associated with it. The name of the list that contains all the packets IDs is summary vector. When couples of nodes encounter, they will exchange their summary vector [1]. By using this protocol the message is dispersed to all the nodes are existed in the network. In addition, nodes in this protocol deal with avoiding connecting with each other several times by keeping and storing the data or the information on the other nodes they recently connected to. The reason for a good percentage of the delivery probability of messages when using Epidemic protocol due to the flooding-based technique. Nevertheless, Epidemic suffers from high overhead ratio due to high consumption of buffer, bandwidth and energy which hence at the end to network congestion.

3.1.2 Spray and Wait (SnW)

Spray and Wait routing protocol is considered as another flooding-based protocol that was suggested by Spyropoulos, Psounis and Raghavendra [7]. The general idea of this protocol is based on bound and limit the number of packet copies in the network. The SnW is classified as two phases which are spray phase and wait phase. Firstly, in the spray phase exactly at the source node the message will be generated, number of copies of this message are dispersed via the source node and probably the intermediate nodes will receive as well. Secondly, in the wait phase this phase is started when the message state is not delivered to the destination node through the spray phase. The nodes that are carried copies of the message will only forward the message to their destinations [1]. The main feature of this

protocol is that compile the performance of Epidemic with high rate of delivery probability with direct transmission to achieve the aim which increases the quality of performance. Nevertheless, this protocol to be performed in a good way, the sizable number of mobile nodes must be existing in the network in order to deliver the message to their destination.

4 Designing and Setting up the Emergency Scenario of Collapse of Töölö Towers

This paper aims to design and set up the emergency scenario which allows the DTN network to be applied to it. By using mobile devices which are carried by the member of the rescue team that have Bluetooth and the cars of the rescue team. This scenario is based on existing one source node and many of destination nodes. The source node will be the Töölö Towers that located in Helsinki city in Finland. This building is considered as a residential building and many people are living there. The coordinate of this building is 60.1772° N, 24.9206° E on the map of Helsinki. This building collapsed due to the earthquake that effected this city. This static point is constant and during this incident, it will send a message to get a response from the rescue team for helping victims. The destination nodes will receive the message and then forwarded it. These nodes are about groups of car and groups of person. The groups of car are about three kinds of rescue car 1-Fire Engine, 2-Ambulance, 3-Police Car. In the other hand, the groups of person are about three kinds of person 1-Fireman, 2-Medic, 3-policeman. The bright blue circle has represented the nodes that are involved in the communication range. The emergency area is illustrated more at the bottom in Figure 3.

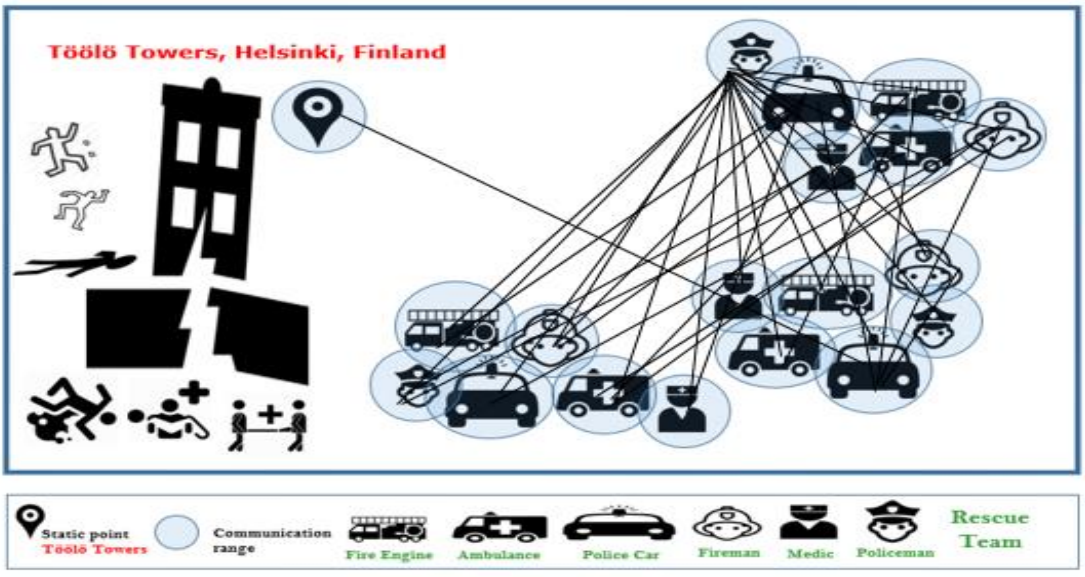


Figure 3: The Emergency Scenario of Collapse of Töölö Towers for the Simulation

4.1 Experiment Set up

This paper aims to evaluate the performance of two DTN routing protocols which are Epidemic and Spray and Wait by using ONE simulator in a simulation time of 1 hours (3600s) defining a different kind of groups firstly, the static point group (Töölö Towers) which has a specific setting for example as shown in the code the ID, the kind of movement and the location of the node on the map:

Group1.groupID = T

Group1.movementModel = StationaryMovement

Group1.nodeLocation = 60,24

Group1.interface1 = highspeedInterface

Group1.nrofInterfaces = 1

Secondly, the car groups which are consisted from three groups each group has special type of car with different ID and another special setting:

Group2.groupID = FE (Fire Engine)

Group3.groupID = PC (Police Car)

Group4.groupID = A (Ambulance)

A Common setting for all car groups:

BufferSize = 120M, waitTime = 10, 30, speed = 0.5, 1.5, nrofHosts = 10, interface1 = highspeedInterface, nrofInterfaces = 1.

Finally, the person groups which are consisted from three groups each group has special type of person with different ID and another special setting:

Group5.groupID = FM (Fireman)

Group6.groupID = PM (Policeman)

Group7.groupID = M (Medic)

A Common setting for all person groups:

BufferSize = 12M, nrofHosts = 20, interface1 = btInterface, nrofInterfaces = 1.

4.2 Parameters Defined in the Simulator

The hole setting and parameters for the entire scenario by using ONE simulator for Epidemic and SnW are shown in Table 1:

Parameter	Value
Scenario name	**The_Collapse_of_Töölö_Tower_EpidemicRouter The_Collapse_of_Töölö_Tower_SprayandWaitRouter**
Scenario run time	**1h (3600s)**
Movement	**Shortest Path Map Based Movement**
Number of node groups	**7**
Number of nodes	**97**
Node speed	**0.5 to 1.5 m/s**
Time to Live (TTL)	**5h (300 mins)**
Map data	**Helsinki map**
Types of interface	**Bluetooth and high speed data**
Interface represented as:	**Static point, Cars and persons**
Node buffer size	**120 MB**

Table 1: Defined Experiment Settings

5 Running the Emergency Scenario of Collapse of Töölö Towers in ONE

The next expected step after designing, setting up and defining the parameter of the scenario is running the scenario by the configuration file. Figure 4 is shown the emergency scenario during the running time when the nodes are connecting for completing sending and receiving the messages. After the simulation is done by the time that is selected before the result of that scenario will be shown in the reports files as txt. format file for each routing protocols. These files will give numbers and statistics about different states of messages such message started, delivered, relayed, dropped etc. that happened during the simulation.

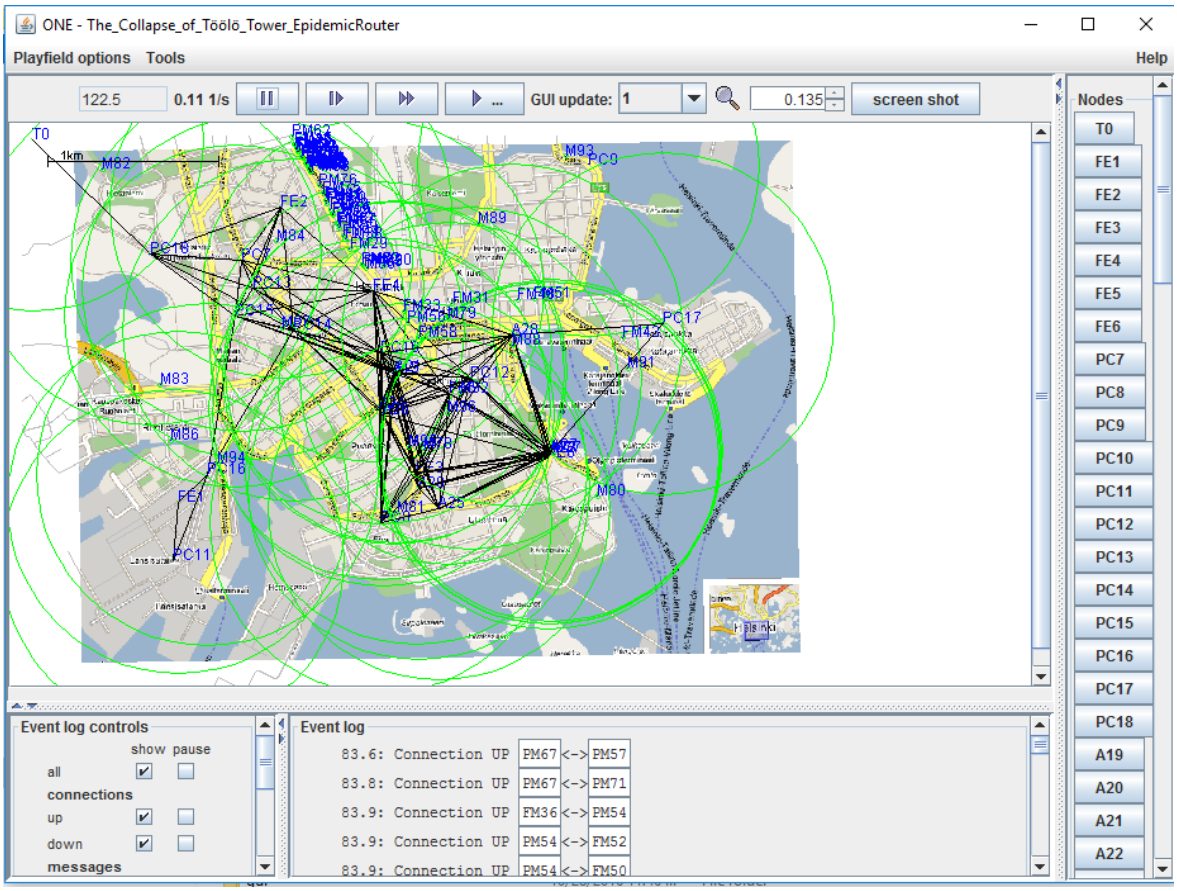


Figure 4: Running the Emergency Scenario of Collapse of Töölö Towers in ONE

6 The Evaluation of performance of the Emergency Scenario of Collapse of Töölö Towers

After running the emergency scenario in ONE simulator, firstly with Epidemic and secondly with SnW the results of the scenario are shown in Table 2.

Metrics	Epidemic	SnW
Created	120	120
Started	12169	1237
Relayed	7082	534
Delivered	32	42
Dropped	5513	0
Latency average	504.6250	500.7429
Overhead ratio	220.3125	11.7143

Table 2: Message Statistics Report for selected DTN routing protocols

These results are created by the *MessageStateReport* and this paper shows these statistics by using charts to visualize the result which indicates the performance of Epidemic and SnW and generally this result based on different metrics such message started, message relayed, message delivered, message dropped, latency average and finally overhead ratio.

Firstly, Started Message: A message is started that means move and transfer from node to another node after it is created.

Secondly, Relayed Message: A message is relayed that means when it transfers between the nodes.

Thirdly, Delivered Message: A message is delivered when it reaches to the goal which is the destination node.

Fourthly, Dropped Message: A message is dropped when it considers and understands that the buffer of the node has restriction and cannot receive it.

Fifthly, Latency Average: The latency average of message is measured and calculated by the time from the message is created at source node and when it is received at the destination node.

Finally, Overhead Ratio: The overhead ratio of message is referred to the amount of time that spent to transmit data in OPPNET.

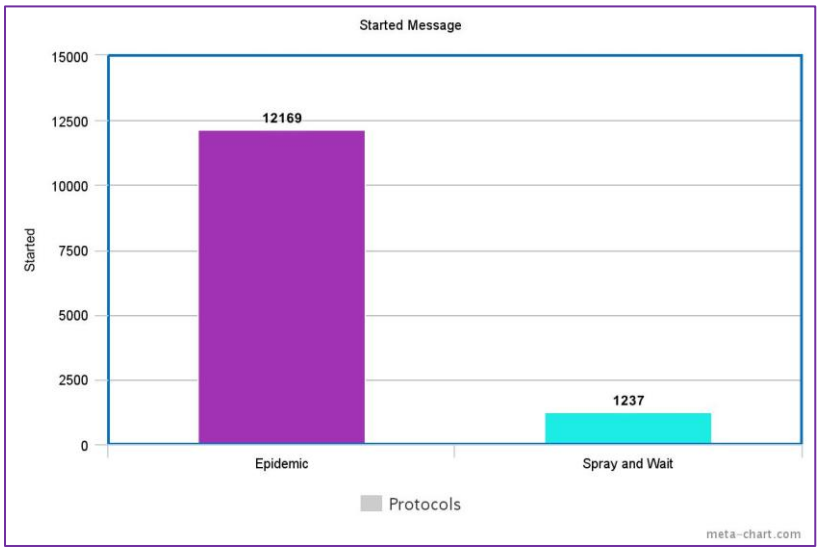


Figure 5: Started Message

Figure 5 shows started message graph which indicates the number of messages that are replicated using Epidemic and SnW. Epidemic tries to replicate the message on each node that it encounters within the communication range by giving the high number of replicated messages which is 12169. However, SnW gives the low number which is 1237, compared with Epidemic due to the efficiency work of SnW and high-ability of controlling the concept of flooding in the network. In addition, as a known SnW has two phases Spray and Wait which effects on the number of started messages due to the fact, that is related to the wait phase this phase will merely copy a message if a close node is its destination.

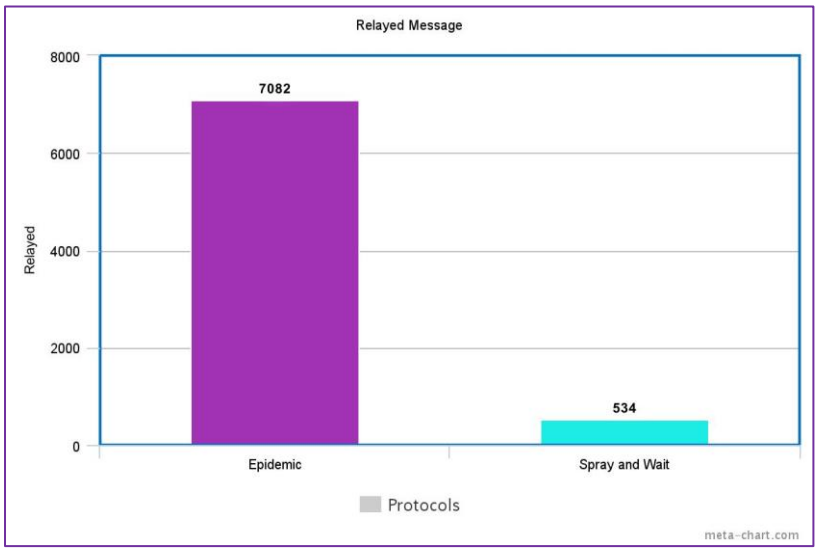


Figure 6: Relayed Message

Figure 6 shows relayed message graph which indicates to finding probably association and correlation between started messages and relayed messages among these two protocols. When the protocol gives a high number of started messages that means these messages will be relayed between the nodes and this principle is obvious and clear with the statistics that given by the Epidemic protocol. Epidemic protocol has the highest number of relayed 7082 compared to SnW which has merely 534.

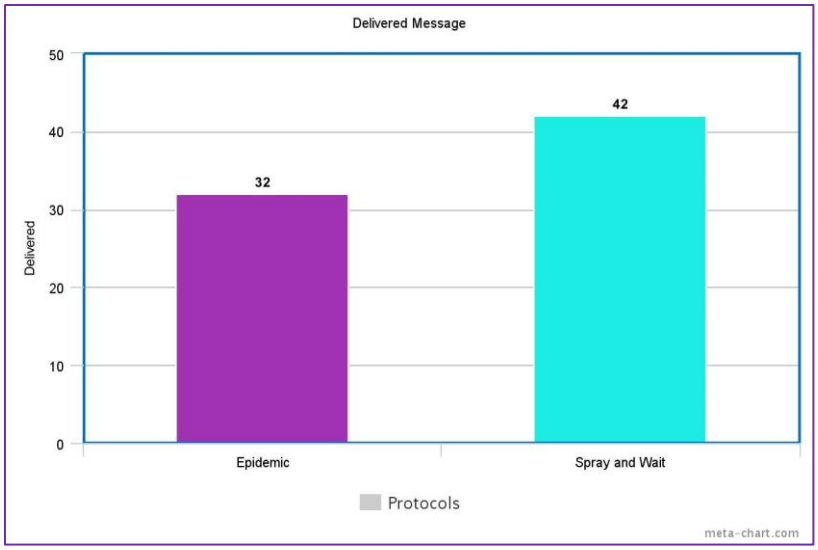


Figure 7: Delivered Message

Figure 7 shows delivered message graph which indicates the number of messages that reached the goal which is the destination. From the graph above it could be saying that Epidemic has the low number of delivered messages due to the congestion that happened during the experiment. The congestion is considered as the most effective factor of causing the low number of delivered message and that is what happened with Epidemic merely 32 messages have reached to the destination. Nevertheless, SnW records the largest number compared with Epidemic by further 10 of delivered messages. in addition, SnW protocol based on the concept of increasing the probability of delivery by decreasing the number of replicating messages.

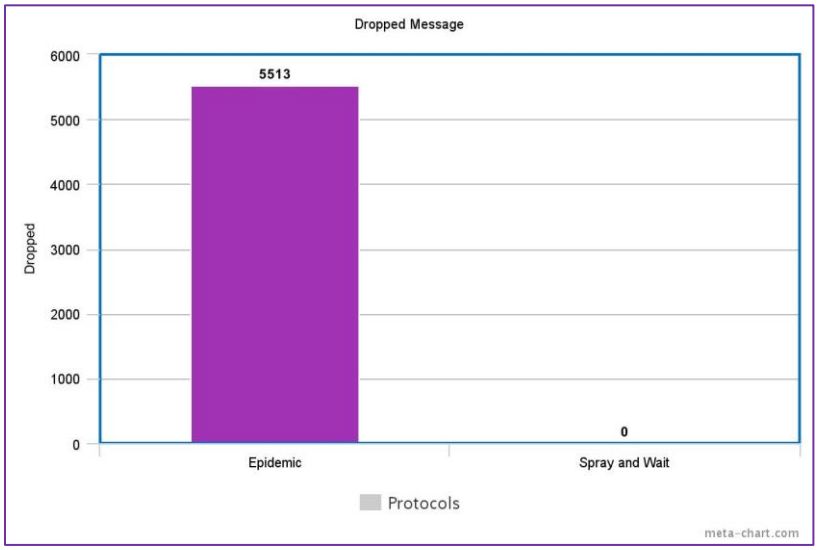


Figure 8: Dropped Message

Figure 8 shows dropped message graph which refers to the number of the packets that are dropped and failed to be delivered during the experiment due to some limitations such the buffer size of the node. From the Figure 8 above it is obvious that SnW records a good result which is zero of dropped messages due to the limit number of copied messages that are stored in buffer and sent directly to the destination in the network. In the other hand, the Epidemic protocol records the high rate of dropped message by giving 5513 due to the flooding approach that it used. Epidemic floods a several numbers of copied messages to the destination inside the network. By using this approach the buffer space of node will be filled and hence it will suffer from congestion and a high number of dropped messages.

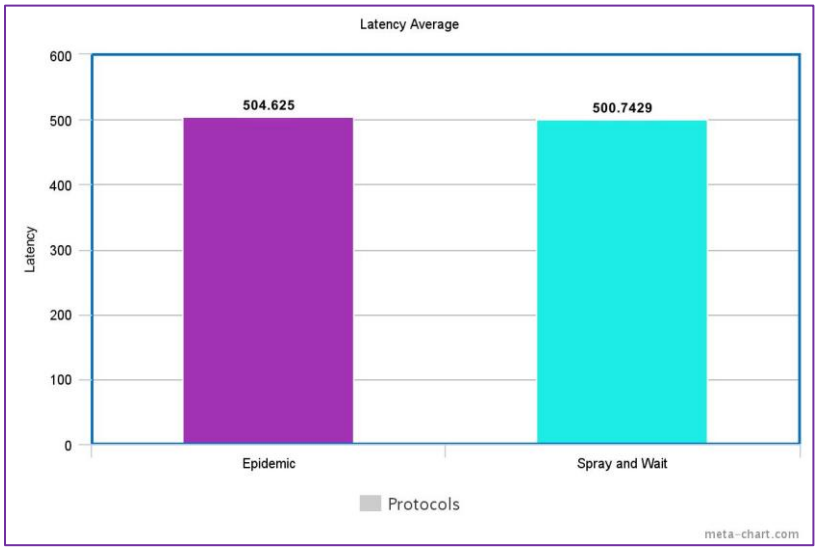


Figure 9: Latency Average

Figure 9 shows latency average graph which mentions to the time it takes between message generation at source node and delivery to the destination nodes. SnW protocol experienced the minimum delay in the generation and receiving messages due to it limits the number of message copies that are sent in the network. As with preceding graphs, congestion levels of Epidemic protocol are higher than SnW, hence it could be concluded that the latency average respectively higher with a large volume of nodes used.

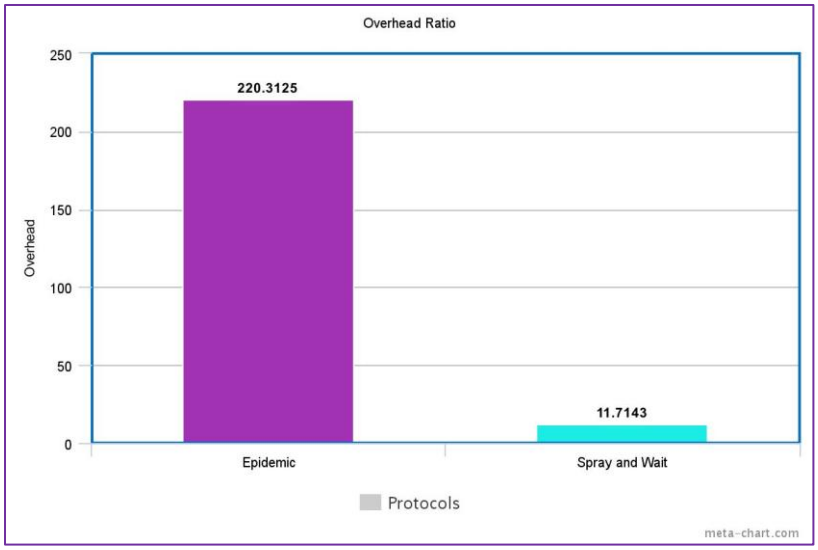


Figure 10: Overhead Ratio

Figure 10 shows overhead ratio graph which compares the ratio of overhead between Epidemic and SnW. It can be seen clearly that SnW has the least number of overhead 11.7143 due to this routing protocol controls the number of message copies sent and forwarded within the network. In the other hand, the Epidemic protocol has the highest number 220.3125 of overhead ratio because it attempts to increase the delivery probability by replicating the messages.

After finishing the analysis that is made above, SnW proves to be better in performance than Epidemic in terms of defined experiment settings and designed the emergency scenario. SnW performed basically better than Epidemic due to it had a high opportunity of delivering the messages without dropping any of them to the destination nodes. In addition, SnW had the least amount of overhead ratio also had the least delay in getting a message to the destination.

7 Opportunistic Network: Pros and Cons

The general concept of the defined emergency scenario is based on sending the message and request help at source point in the incident location then get a response from the rescue team when they receive the message will come and help the victims then making a control on the incident area with less infrastructure network. There are some challenges that will face any of DTN routing protocols that are using with the chosen scenario due to the absent of end-to-end connectivity between source and destination node hence the topology of the OPPNET might change randomly and dynamically. In addition, the source node at the edges of the city

where not many cars pass very often and this is a fundamental challenge problem for our scenario. We envisage that having roadside unit within the coverage of the tower and close to the main road would significantly improve the performance. The most prominent difference between OPPNET and other networks is that there is no guarantee of end-to-end connectivity between the source and destination node, hence the high latency average is inevitable and in fact that is what happened with the two chosen routing protocols Epidemic and SnW during our experiments and showed that they can deliver the messages despite the challenging conditions. The nodes in OPPNET are mobile and battery which might suffer from limited resources such when the data is forwarded to the next node this data should be stored in a safe way until the connectivity is available to the next node. However, the other new data can be received and then might take another portion of the buffer size due to increases the buffer sizes of this network. Therefore, another limited resource like limited memory capability will constraint the buffering of data [6]. The main goal of OPPNET when it is applied to any scenario firstly, with routing protocol of DTNs to increase the number and maximize the probability of messages deliver; and secondly, with resources of nodes is to decrease the number and minimize the buffer size, the bandwidth of network and the battery energy consumption which is merely considered as important for routing. However, maximizing the delivery of messages and minimizing the resource consumption at the same time will lead for conflicting with each other. The best way is to allocate the resources, such as if it can be ensured that the data will be effectively, surely and undoubtedly delivered to the destination then no need to store the copies of data in all the hosts of network. On the other hand, if merely necessary for maximizing the prospect of delivering a certain message then a good way is increasing the number of copies of that message at several hosts. By using this way, the performance of routing protocols will be at the high level of quality and efficiency. The reliability of OPPNET is usually doubtable however, with any routing protocol of DTNs as an example should have some acknowledge that can be ensured an effective and steady of messages delivery. The security of OPPNET, in the networks world the security of data is still issue not only for the OPPNET but merely for conventional network, the message may disconnect before reaching destination. With end-to-end routing using the technique of cryptographic might be valuable for security due to the receiver will know whether the message incurred to untrusted hosts [6]. Generally, the security of OPPNET is still an open, wide and broad issue which available to be under the circle debate. OPPNET like DTN is efficiently used in various and different environments that are subjected to delay, disconnection or disruption. OPPNET will overcome these circumstances and more specifically it could be applied to many applications such a commercial application with long delay and

intermittent connectivity, terrestrial wireless networks with no end-to-end connectivity and satellite networks with long delay or periodic connectivity [6]. Nowadays, there has been a rising attention in emergency controlling systems and the organization of rescue teams are main characteristics of these systems. Many of disasters scenario as a real emergency scenario such Tsunami, hurricane Sandy or Earthquake that are relied on network infrastructure, after the incident is happened the network might be unobtainable. There are many methods to deal with these problems one of them using OPPNET like DTN to offer network which is suitable for these cases with not fully connected nodes.

Conclusion:

In conclusion, opportunistic network such DTNs is considered as an updated network that changed and turned the traditional concept of an end-to-end path, in order to give the messages a full freedom mobility of moving and forwarding without a steady path. Despite the idea of routing in DTNs is a new part of research but it can be seen that the results of research have grown rapidly. This paper has shown the worthy description of OPPNET and ONE simulator with deep details. With ONE simulator, the researcher has could create a complex mobility scenario that comes closer to the reality. All the routing protocols have the same aim which is attempting to increase the delivery probability ratio of messages with decreasing the consumption of resources at the same time.

This paper has shown and presented a comparative in performance of two DTN routing protocol Epidemic and SnW with a designed emergency scenario. The results that are gotten have been discussed in this paper. Even though many routing protocols have been considered, there are still many challenges should be solved. Overall, all that can be said the researchers believe that it is an appropriate time to study and consider a wide range of network characteristics in expressing a new architecture of network.

List of References:

1. Abdelkader, T., Naik, K., Nayak, A., Goel, N. and Srivastava, V. 'A performance comparison of delay-tolerant network routing protocols', *IEEE Network*, 30(2), pp. 46–53. doi: 10.1109/mnet.2016.7437024, 2016.

2. Heinemann, A., Kangasharju, J. and Mühlhäuser, M. 'Opportunistic data dissemination using real-world user mobility traces', *International Conference on Advanced Information Networking and Applications, AINA*, pp. 1715–1720, 2008.

3. Keränen, A., Ott, J. and Kärkkäinen, T. 'The ONE simulator for DTN protocol evaluation', pp. 79–88. doi: 10.4108/ICST.SIMUTOOLS2009.5674, 2009.

4. Poonguzharselvi, B. and Vetriselvi, V. 'Trust framework for data forwarding in opportunistic networks using mobile traces', *International Journal of Wireless & Mobile Networks*, vol.4 (6), pp. 115–126. doi: 10.5121/ijwmn.2012.4609, 2012.

5. Samyal, V.K., Bamber, S.S. and Singh, N.'Performance Evaluation of Delay Tolerant Network Routing Protocols', *International Journal of Computer Applications*, , pp. 24–27, 2015.

6. Shen, J., Moh, S. and Chung, I. 'Routing protocols in delay tolerant networks: A comparative survey',pp. 1577–1580, 2008.

7. T Spyropoulos, K. Psounis, and C. S. Raghavendra, 'Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks', Proc. ACM SIGCOMM workshop on Delay-Tolerant Networking (WDTN "05) ACM, New York, NY, USA, pp.252-259, 2005.

8. A. Vahdat and D. Becker, 'Epidemic routing for partially connected ad hoc networks', in Technical Report CS-200006, Duke University, April 2000.

9. V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss, 'Delay Tolerant Network Architecture', [Online] available at https://www.rfc-editor.org/rfc/rfc4838.txt, 2007 [accessed at 08 November 2016].

10. Yogi, M. kumar and chinthala, V. 'A Study of Opportunistic Networks for Efficient Ubiquitous Computing', *International Journal of Advanced Research in Computer and Communication Engineering*, vol.3 (1), pp. 5187–5191, 2014.