Performance Analysis of Opportunistic Routing Algorithms in Highly Heterogeneous Environment

A Project report
submitted in partial fulfillment for the award of the degree of

Bachelor of Technology

IN

INFORMATION TECHNOLOGY

by

ALANGKRITA SARKAR (GCECTB-R19-2004)
(Registration No. 020892 of 2019-20)

AMILY MALLICK CHOWDHURY (GCECTB-R19-2005)
(Registration No. 020817 of 2019-20)

PARTHA BHOWMICK (GCECTB-R19-2016)
(Registration No. 020827 of 2019-20)

SUMIT MANDAL (GCECTB-R19-2035)
(Registration No. 020683 of 2019-20)

Government College of Engineering & Ceramic Technology
(An Autonomous Institution under MAKUT)

73, Abinash Chandra Banerjee Lane, Kolkata- 700010

2023

BONAFIDE CERTIFICATE

Certified that this project report **“Performance Analysis of Opportunistic Routing Algorithms In Highly Heterogeneous Environment”** is the bonafide work of “**Alangkrita Sarkar**, **Amily Mallick Chowdhury, Partha Bhowmick, Sumit Mandal”** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. Mausumi Maitra **HEAD OF THE DEPARTMENT**	Mr. Ritwik Mondal **SUPERVISOR**
INFORMATION TECHNOLOGY	**ASST. PROFESSOR** **INFORMATION TECHNOLOGY**

SIGNATURE

External Examiner:

DECLARATION

We declare that this project report titled **Performance Analysis of Opportunistic Routing Algorithms in Highly Heterogeneous Environment** submitted in partial fulfillment of the degree of **B.Tech. in Information Technology** is a record of original work carried out by us under the supervision of **Mr. Ritwik Mondal**, and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Alangkrita Sarkar (GCECTB-R19-2004)

Amily Mallick Chowdhury (GCECTB-R19-2005)

Partha Bhowmick (GCECTB-R19-2016)

Sumit Mandal (GCECTB-R19-2035)

Kolkata - 700010
Date - 28/04/2023

ACKNOWLEDGEMENT

We would like to thank Mr. Ritwik Mondal, our supervisor-in-charge and our Head of the Department, Dr. Mausumi Maitra for their support and guidance in completing our project on the topic “Performance Analysis of Opportunistic Routing Algorithms in Highly Heterogeneous Environment”. It was a great learning experience.

We would like to express our gratitude and appreciation to Ananya Biswas, Sushmita Panda and Abhisek Bose for giving their full effort in guiding us to complete this project report. The project would not have been successful without their cooperation and inputs.

____

Alangkrita Sarkar (GCECTB-R19-2004)

____

Amily Mallick Chowdhury (GCECTB-R19-2005)

____

Partha Bhowmick (GCECTB-R19-2016)

____

Sumit Mandal (GCECTB-R19-2035)

Abstract

Opportunistic Network (OppNet) was first envisaged as a type of Delay/Disruption Tolerant Network (DTN) but very soon Mobile Ad-hoc Network (MANET) got its resemblance with OppNet. The area of research for OppNet expanded and the routing in OppNet emerged as the core research area. The central idea as well as the challenge of routing in OppNet is the forwarding of messages because the contact between mobile nodes are either very small in number or the contact duration is very little. So, establishment of source to destination path becomes very difficult and the conventional store and forward type of routing is not a viable solution. A solution for that problem can be store-carry-forward type of routing where the mobile node itself takes the custody of the message to be routed until a new forwarding node is found. This type of routing is known as Opportunistic routing as the message gets forwarded on opportunity. In the last two decades a substantial amount of articles on Opportunistic routing algorithms were published encompassing different techniques and philosophies. These algorithms were simulated, the performance were measured and analyzed for different metrics like average delivery probability, average hop count, average overhead ratio, average latency. But a common limitation was found, which had been the amount of variation in simulation environment. In this project we selected six such Opportunistic routing algorithms namely Direct Delivery, First Contact, Epidemic, Spray and Wait, MaxProp and PRoPHET as representatives for others and simulated in ONE simulator with high degree of heterogeneity in simulation environment. We gave emphasis on parameters like buffer size, movement speed, transmission range, transmission speed, interface. The performance found and analyzed was mostly as expected but some results was very intriguing in nature. Throughout this project report we have presented setting up the environment, running the simulation, analysis of the results and the comparative studies.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.

1. INTRODUCTION

In recent years, multi-hop wireless networks have already become very popular and are receiving an increasing amount of attention by the research community. Compared to wired networks, routing in these networks is specially challenging because of two fundamental differences. The first one is the heterogeneous characteristics of the wireless links: due to the strong dependency of radio transmission impediments between the nodes with their distance and the environmental elements influencing the radio waves propagation. As a consequence, packet delivery probabilities may be significantly different for every link of a multi-hop wireless network. The second one is the broadcast nature of wireless transmissions: unlike wired networks, where links are typically point to point, when a node transmits a packet in a wireless network, this can simultaneously be received by several neighbouring nodes.

Traditional routing protocols proposed for wireless networks perform best path routing, i.e., preselect one fixed route before transmissions starts. Each node in a route uses a fixed neighbour to forward to. Doing this way, in the routing table of every node participating in the routing between a source and a destination, there is a forwarding entry which points to a neighbour (referred to as next-hop), over which packets addressed to the destination will be sent. Note that once all next-hops have been chosen, all packets between a source and destination follow the same path. This motivates the name of uni-path routing for such type of protocols. These approaches borrowed from the routing protocols for wire-line networks, and do not adapt well to the dynamic wireless environment where transmission failures occur frequently.

Ad hoc network is a wireless network that allows easy connection establishment between wireless client devices in the same physical area without the use of an infrastructure device, such as an access point or a base station. A mobile ad hoc network (MANET) is a collection of mobile nodes that act as both routers and hosts in an ad hoc wireless network and that dynamically self-organize in a wireless network without using any pre-established infrastructure [1-10]. Nodes typically transmit in broadcast messages that reach only nearby nodes. One of the major challenges in MANETs is in efficiently routing the data packets from the source to the destination. Traditional topology-based protocols depend on predetermined routes between source and destination devices. With highly mobile nodes, it is impossible to maintain a deterministic route. Also, the discovery and recovery procedures are time and energy consuming.

Another field of opportunistic network, along with MANET branches towards delay tolerant network. A delay-tolerant network (DTN) is a network that is designed to operate effectively in extreme conditions and over very large distances, such as with space communications. In such environments, the conventional internet does not work; networks are also subject to frequent disruptions, high error rates and latency of hours or even days. A DTN can overcome such challenges and reliably transmit information, ensuring dependable internetworking.

If the mobility is high and the best forwarder is unavailable, DTN mode is selected to avoid link disruption problem of MANET.

Opportunistic routing and forwarding have provided an efficient solution to this link disruption problem. Opportunistic network is a type of delay tolerant network (DTNs), where data will be routed with tolerant delay from source to destination. Opportunistic routing protocols [1,2] were proposed to offer reliable data delivery and excellent Quality of Service (QoS) to applications using MANETs for communication and resource sharing. Instead of (pre) selecting a specified relay node at each transmission, Opportunistic routing broadcasts a data packet to a set of relay candidates. Then, relay candidates, who have successfully received the data packet, run a coordination protocol to select the best relay to forward the packet.

The performance of such opportunistic networks may vary significantly, depending on how the mobile nodes move, how dense the node population is, and how far apart the sender and the receiver are. Delivery latency may vary from a few minutes to hours or days, and a significant fraction of the messages may not be delivered at all. The key factors are the routing and forwarding algorithms used and how well their design assumptions match the actual mobility patterns. No ideal routing scheme has been found so far.

Opportunistic Routing, also referred to as diversity forwarding [2], cooperative forwarding [3] or any-path routing [4], is being investigated to increase the performance of multi-hop wireless networks by taking advantage of its broadcast nature. In Opportunistic Routing, in contrast to traditional routing, instead of preselecting a single specific node to be the next-hop as a forwarder for a packet, an ordered set of nodes are selected as the potential next-hop forwarders. A benefit of Opportunistic Routing is that it increases the reliability of transmissions by combining weak physical links into one strong virtual link. In other words, it acts like Opportunistic Routing has additional backup links and the possibility of transmission failure is reduced [5].

In sparse mobile ad hoc networks, there may exist no contemporaneous end-to-end path between a pair of nodes. This violates the assumption on which many ad hoc routing protocols [5, 6] are based, thereby providing room for conducting research in identifying ways to provide connectivity in an environment that does not rely on the Internet's basic assumptions, such as presence of bidirectional end-to-end paths. Zebra net or wildlife monitoring, disaster recovery networks, and military applications are some of the examples of such environments. Opportunistic networking help to cope with this contrast by making use of transmission opportunities emerging from coarse-grained mobility within the network. Sporadic links appearing among nodes can be eventually construed over a period of time as presence of a complete path between a source and destination pair. Therefore, routing in these networks is performed in per-hop manner, in which every node opportunistically selects a next-hop relay to achieve eventual delivery, with store-carry-and-forward paradigm [7].

However, due to intrinsic nature of opportunistic networks, based on sporadic connectivity, high packet loss ratios are inevitable. Intuitively, many routing protocols [8–11], for opportunistic environments, are accustomed to spread multiple redundant copies of a message in order to achieve throughput and efficiency in end-to-end latency. These replication based schemes can be categorized into greedy-based [8, 9] and utility-based approaches. Greedy-based replication assumes homogeneous set of nodes [11] and spread replicas on every proximity encounter to increase the probability of successful delivery. However, this assumption reveals itself unrealistic in many scenarios where nodes exhibit heterogeneous characteristics, in terms of their mobility patterns and available resources, that might be deterrent for their participation in the delivery process. There has been a succession of utility-based routing schemes [10, 11]—choosing intermediate relays based on next-hop fitness of nodes—to avoid unnecessary overhead incurred by greedy replication in heterogeneous environments. In this case, number of extra copies of a message is directly proportional to the number of nodes presenting next-hop fitness above certain threshold value. Hence, limitations in determining next-hop utility of a node can ironically result in very high number of message replicas compared to those actually needed to achieve successful delivery at eventual destination. Each copy consumes energy to transmit along with extra computational resources, thereby leading to suboptimal performance in resource-stringent networks. Number of forwarding tokens, however, can also be assigned to limit message replication [12–14]. In this case, first encountering nodes meeting the given utility criteria are selected for distribution of tokens at each hop. Hence, correct determination of next-hop fitness of a node becomes more apparent to achieve eventual delivery.

Moreover, the intrinsic nature of utility-based schemes, based on pruning the epidemic distribution tree, makes it inevitable that the majority of the network traffic is carried by only the most suitable nodes, resulting in an unfair load distribution [15]. Intuitively, the utility heuristics based on any single attribute are not sustainable as the inefficiency of the algorithm to get itself cognizant of changing network characteristics can quickly deplete constraint resources in few better nodes.

In this project report, we present the studies of comparisons of various routing in the most heterogenous circumstances. By comparing six opportunistic routing schemes in a simulator, we can compare the nature of these routings under heterogenous conditions. Opportunistic Networking Environment (ONE) simulator, a Java- based tool offering a broad set of DTN protocol simulation capabilities. The ONE simulator offers an extensible simulation framework itself supporting mobility and event generation, message exchange, DTN routing and application protocols, a basic notion of energy consumption, visualization and analysis, interfaces for importing and exporting mobility traces, events, and entire messages. Using this framework, we implemented an extensive set of ready-to-use modules: six synthetic mobility models that can be parameterized and combined to approximate real-world mobility scenarios, six configurable well-known DTN routing schemes, a set of base primitives to design application protocols, a basic battery and energy consumption model, several input/output filters for interacting with other simulators, and a mechanism for the integration with real-world testbeds. The ONE simulator is designed in a modular fashion, allowing extensions of virtually all functions to be implemented using well-defined interfaces.

2. RELATED WORKS

Recent advances in wireless networks have enabled us to deploy and use mobile ad hoc networks for communication between the rescue officers in disaster recovery and reconstruction operations. This highly dynamic network does not require any infrastructure or centralized control. As the topology of the network remains dynamic, severe performance limitations occur with traditional routing strategies. Recently a new routing paradigm known as opportunistic routing protocols have been proposed to overcome these limitations and to provide efficient delivery of data in these highly dynamic ad hoc networks. In opportunistic networks, mobile nodes communicate even if a connecting route doesn't exist. In Traditional routing protocols fixed routes are selected for transmission and packets are forwarded for each hop. In a dynamic wireless environment, transmission failures occur frequently due to path breaks and demands retransmissions. This unreliable transmission causes additional traffic in the network and waste of network resources and bandwidth.

Opportunistic routing promises to be an efficient routing protocol to improve the performance of wireless networks providing increased network throughput and transmissionreli-ability. Selecting the forwarding nodes and coordinating them helps reduce duplicated retransmissions and increase network throughput and thus provide improved performance in wireless networks. Forwarding nodes can be selected dynamically for each packet and each hop, on the basis of the actual network performance. The paper [16] discusses the concept behind opportunistic routing, its classification, metrics used, efficiency of these protocols based on the different candidate selection algorithm and coordination method were examined. Using the broadcasting nature of the wireless medium, this latest routing technique tries to address two major issues of varying link quality and unpredictable node mobility in ad hoc networks. The paper [17] describes the meaning of Opportunistic Routing (OR) which has been introduced as a way of using the broadcast nature of multi-hop wireless networks. Although ETX is simpler to compute than EAX, it does not accurately compute the expected number of transmissions under OR. The other important issue of OR is the candidate selection. Four different candidate selection algorithms that have been proposed in the literature. They range from non-optimum, but simple, to optimum, but with a high computational cost. To show the differences between each algorithm under study a simple example running for each algorithm has been used. Regarding the different candidate coordination approaches they have described the four most used methods of coordination in

OR: acknowledgment-based, timer-based, network coding and RTS-CTS; and have explained the advantages and disadvantages of each approach. Applying the OR paradigm to multicast routing is another new research direction. The availability of multiple destinations can make the selection of CS and coordination among them complicated. There are few works that have tried to adapt OR to multicast settings. They have briefly described different protocols that apply OR to multicast routing. Most of them first create the shortest path tree to the destinations and then send the packet through the tree using OR. After the general overview and introduction of OR they have focused on its performance analysis. First, they have surveyed the existing performance studies that are based on analytical models. Then, they have introduced their own contribution, a discrete time Markov chain model to study the performance gain that may be achieved by using OR. In their model the nodes are represented by the states of the chain, and the state transitions model how the 33 packet progresses through the network. The only ingredients needed to build the transition probability matrix are the candidates of each node, and the delivery probabilities to reach them. As a consequence, the proposed model can be applied independently of the candidate selection algorithm that is employed. The model leads to a discrete phase-type representation for the distribution of the number of transmissions that are needed to reach the destination node. An important advantage of the phase-type representation is that there exist simple and closed-form expressions for its distribution and moments.

Unlike conventional IP forwarding, where an intermediate node looks up a forwarding table for a suitable next hop, opportunistic routing brings in opportunistic data forwarding that allows multiple candidate nodes in the forwarding area to act on the broadcasted data packet. This increases the reliability of data delivery in the network with reduced delay. The research paper [18] analyses and compares the various advantages, disadvantages and the performance. In this paper the behaviour and performance of opportunistic routing protocols in highly mobile ad hoc networks are investigated. The working of opportunistic routing was discussed in detail along with the metrics used in each protocol. They have discussed the latest applications of opportunistic routing that include maintaining communication in disaster recovery operations. Opportunistic protocols are classified into five categories and they have discussed the working of the most popular and widely used protocol from each category. Various advantages and disadvantages of these protocols were discussed in detail. Simulations are used to analyse the performance of these protocols in highly mobile ad hoc networks. Results from simulations showed the variations in behaviour and performance of these opportunistic protocols. The performance analysis clearly showed that the opportunistic routing protocols working on link quality and positional information gives much better performance in fast changing MANETs. This study would further help in designing next level optimized opportunistic protocols that would guarantee very high quality of service in highly dynamic ad hoc networks.

A variety of applications and forwarding protocols have been proposed for opportunistic networks (OppNets) in the literature [19]. However, the methodology of evaluation, testing and comparing these forwarding protocols are not standardized yet, which leads to large levels of ambiguity in performance evaluation studies. Performance results depend largely on the evaluation environment, and on the used parameters and models. More comparability in evaluation scenarios and methodologies would largely also improve the availability of protocols and the repeatability of studies, and thus would accelerate the development of this research topic. In this survey paper, they focus their attention on how various OppNets data forwarding protocols are evaluated rather than what they actually achieve. They explore the models, parameters and the evaluation environments and make observations about their scalability, realism and comparability. Finally, they deduce some best practices on how to achieve the largest impact of future evaluation studies of OppNets data dissemination/forwarding protocols. In this paper [19], the focus is on the performance evaluation of opportunistic networking data dissemination/forwarding protocols. Differently from other surveys, there is a detailed discussion about how the protocols were evaluated instead of what they actually do or whether they perform well. This study has led to two main outcomes: a best practice evaluation process and a list of suggestions on further improving the process for the whole community.

Delay-tolerant Networking (DTN) makes successful communication in sparse mobile ad-hoc networks and other challenged environments where there is no end-to-end path established unlike traditional networking. PRoPHET routing protocol in DTN uses for delivery predictability of node combat to select and forward bundles to its neighbor node indifferent of the distance between them. The work proposed in the report [20] look into a number of “single-copy” routing schemes that use only one copy per message, and hence significantly reduce the resource requirements of flooding-based algorithms. In this paper [20], exploration of the single-copy routing space in order to identify efficient single-copy solutions that (i) can be employed when low resource usage is critical, and (ii) can help improve the design of general routing schemes that use multiple copies. Also focused on some of the multi copy case techniques to decrease delivery ratio and delay. The proposed algorithm would give better results in comparison of epidemic routing and direct delivery routing protocol. Most important factors are torus size, mobility model and routing algorithm. To prevent resource contention an algorithm is used and analyzed. Mobility models include only random walk and random way point but a new mobility model is to be implemented in existing model to improve this result. This algorithm only provides a mechanism to decrease resource contention but they also want to increase delivery ratio. PMA algorithm uses some main factor that forwarding decision made based on probability and movement of nodes. Though this algorithm also works on random walk and random waypoint algorithm and gives also some different results. PMA algorithm uses some main factors that forwarding decision made based on probability and movement of nodes. Though this algorithm also works on random walk and random waypoint algorithms and gives also some different results.

Recently the increasing number of sensors integrated in smartphones, especially the iPhone and Android phones, has motivated the development of routing algorithms for Opportunistic Mobile Sensor Networks (OppMSNs). Although there are many existing opportunistic routing algorithms, researchers still have an ambiguous understanding of how these schemes perform on OppMSNs with heterogeneous architecture, which comprises various kinds of devices. In this work, we investigate the performance of well-known routing algorithms in realistic scenarios. To this end, we propose a heterogeneous architecture including fixed infrastructure, mobile infrastructure, and mobile phones. The proposed architecture focuses on how to utilize the available, low-cost short-range radios of mobile phones for data gathering and dissemination. We also propose new realistic mobility models and metrics. Selected routing protocols are simulated and evaluated with the proposed heterogeneous architecture, mobility models, and transmission interfaces under various constraints, such as limited buffer size and time-to-live (TTL). Results show that some protocols suffer long TTL, while others suffer short TTL. We further study the benefit of fixed infrastructure in network performance, and learn that most of the opportunistic routing algorithms cannot benefit from the advantage of fixed infrastructure since they are designed for mobile nodes. Finally, we show that heterogeneous architecture needs heterogeneous routing algorithms, such as a combination of Epidemic, Spray and Wait, and context-based algorithms. In the paper [21], a heterogeneous architecture comprising fixed infrastructure, mobile infrastructure, and mobile nodes is proposed. In addition, a realistic mobility model and metrics are proposed. Several well-known opportunistic routing protocols are tested with this architecture under constraints of limited buffer size, message size, time-to-live, and unpredictable movement. Observation shows that none of the evaluated protocols performs well with a heterogenous scenario, such as the one described in this paper. It's also observed that most of the algorithms do not improve their performances when adding RSUs. Since a single simple routing algorithm does not suffice to improve the overall message delivery performance, a contribution of several algorithms should be considered: Road Side Units (RSU), as used in the backbone network, should not only carry received information to a central server but also disseminate information to nearby passing nodes. This communication shortcut leaves the base station out of the loop and contributes to a better delivery speed and delivery cost. The Epidemic routing protocol with a flooding control mechanism is best suitable for the RSU network if delivery cost is not the most important. Buses, which act as data mules or message ferries, have a mobility pattern based on fixed routes and time schedules. The Message Ferry routing protocol is most appropriate. Pedestrians and cars are best served by stochastic and context-based schemes. However, exchanging messages between nodes that use different routing protocols is a challenge. For example, nodes running PRoPHET fail to update the delivery predictability of nodes running Epidemic due to the unavailability of delivery predictability in Epidemic routers.

Wireless sensor networks (WSNs) have been used for environmental monitoring and reporting for many decades. Energy consumption is a significant research topic because wireless sensor nodes are battery-operated to be highly energy-constrained. Several strategies have been introduced in routing and MAC (Medium Access Control) layer protocols to facilitate energy saving. At the routing layer, an energy-efficient routing protocol, known as opportunistic routing (OR), has been designed to improve efficiency. OR achieves energy efficiency via load-balancing, which forwards packets along multiple routes over WSNs. At the MAC layer, an energy-efficient MAC protocol known as the asynchronous duty-cycled MAC (ADCM) protocol achieves energy saving by turning on and off a sensor node's transmitter and receiver to eliminate unnecessary energy wastage. These protocols each have their own advantages and disadvantages. OR achieves energy efficiency at the routing layer but it raises an issue at the MAC layer. ADCM achieves energy efficiency at the MAC layer, but it hinders the packet forwarding efficiency of the OR. To attain better energy efficiency, a combination of these two ideas led to the development of OR with asynchronous duty-cycled MAC (OR-ADCM). However, even with better energy efficiency, limitations still exist in combining load-balancing and duty-cycling due to conflicts in the inherent properties of OR and ADCM. In the paper [22], they present a survey of the evolution of OR-ADCM over WSNs to help the reader better understand and appreciate the details of this tradeoff, which they hope will lead to the development of better protocol designs.

One of the most challenging issues in the routing protocols for underwater wireless sensor networks (UWSNs) is the occurrence of void areas (communication void). That is, when void areas are present, the data packets could be trapped in a sensor node and cannot be sent further to reach the sink(s) due to the features of the UWSNs environment and/or the configuration of the network itself. Opportunistic routing (OR) is an innovative prototype in routing for UWSNs. In routing protocols employing the OR technique, the most suitable sensor node according to the criteria adopted by the protocol rules will be elected as a next-hop forwarder node to forward the data packets first. This routing method takes advantage of the broadcast nature of wireless sensor networks. OR has made a noticeable improvement in the sensor networks' performance in terms of efficiency, throughput, and reliability. Several routing protocols that utilize OR in UWSNs have been proposed to extend the lifetime of the network and maintain its connectivity by addressing void areas. In addition, several survey papers were presented in routing protocols with different points of approach. The paper [23] focuses on reviewing void avoiding OR protocols. In this paper, they briefly present the basic concept of OR and its building blocks. They also indicate the concept of the void area and list the reasons that could lead to its occurrence, as well as reviewing the state-of-the-art OR protocols proposed for this challenging area and presenting their strengths and weaknesses.

In the paper [24] with the use of Ant Colony Optimization an attempt is made to improve the throughput of a network. By using between-ness centrality the packet drop rate and overhead ratio is reduced. In routing where nodes are themselves carriers then nodes in the network are also data collectors. Nodes are following either arbitrary or predetermined routes around in the network area to move. Moving of nodes required to gather messages from the encountered nodes. These can be the only entities culpable for delivery of messages. This paper concludes that routing done through a mobility-based approach is not delivering better throughput as compared to social based. The packed dropped rate is also higher and it tends to increase w.r.t scalability in mobility based (Spray and Wait and PRoPHET). The ratio of overhead is also better in social based (SimBet and Bubble Rap). The work can also be enhanced by evaluating these protocols with respect to other parameters like buffer constraints, delay, and energy saving techniques.

In real-world applications, wearable devices, taxis, and buses might be the nodes of opportunistic networks to save-carry-forward a message from the source to the destination, particularly when facing a lack of infrastructure or sending an emergency alert from an injured involved in an accident to a hospital or rescue team. Albeit with various weights, NumMC, BuffS, MI, NumN, and NM influence the network's performance. To explore the real-world applications' restrictions and impact of resources on the network performance, they deployed three datasets that differed in features and structures to represent the characteristics of such networks in the paper [24]. Extensive simulations and analyses are required to show the impact of each factor on the network's performance. They found that the nodes and the respective features are the most significant influences on the network performance represented by MDP, DMP, and NetO. This means that NumN, NM, and NumMC have greater weights than BuffS and MI. In the real world, this is mapped to the type of node (taxi or bus), its speed (NM), and the route it follows. All these factors are impacted by the route and regulation features determined by the dataset represented by the algorithm used in the simulation.In a vast network, this requires specific configuration and tests. Thus, they used regressions techniques of machine learning to predict the optimized parameters for both networks with(out) resource restrictions. They showed that obtaining the optimized parameters in each scenario rather than general configuration improves the network performance under different routing algorithms (i.e., PPHB++, Prophet, and Epidemic). However, care must be taken to choose the appropriate means (wearables/taxis/bus) depending on the purpose and structure of the environment (city). Their study showed that buses with lower NetO are appropriate in OppNets if the message is time-tolerant. The results indicated that using taxis in modern texture cities with a higher network overhead (e.g., several taxis in the same area and direction), greater speed (NM), and fewer restrictions on driving are suitable. The datasets' features restrict their work in terms of data acquisition (two days) and node characterizations (limited by node density and mobility and 500 nodes). For future work, they plan to consider pedestrians included in their dataset. Therefore, they plan to implement an OppNet on several wearable devices deployed in real situations with true-to-life scenarios. In addition, they desired to compare these results with the simulation outcomes and improve the ML algorithms in terms of performance, speed, and time to train. The paper [25] introduces a methodology for the development of routing algorithms that takes into consideration opportunistic networking. The proposal focuses on the rationale behind the methodology, and highlights its most important stages and components. It also discusses the importance of two core elements in the process of protocol designing: the scenario selection, based on essential characteristics, and the choice of standard evaluation metrics.

As designing an optimal routing protocol with a delivery probability of 100% under all conditions is difficult, prioritizing messages becomes a necessity. Message prioritization relies on the importance of information, creation time, or source location. Priorities must be defined by a specific application, for instance, public safety applications define the priority based on the source location, creation time, and seriousness of detected events. One last point of concern is the security and privacy of information. A leading principle should be that the creator owns the data and decides how the data can be used by others. However, one may argue that in situations of emergency this principle may be overruled by authorities. These issues will be hopefully addressed in future research.

3. BACKGROUND

3.1. Opportunistic Network

An opportunistic network [27], also known as OppNets, is a set of mobile devices commonly called nodes that exchange information between them exploiting direct communication opportunities to perform an end-to-end transfer of data. Nodes communicate with each other even if an end-to-end route never exist. Furthermore, nodes are not supposed to possess or acquire any knowledge about the network topology. With the growth of the use of mobile devices in recent years, opportunistic networks have become a significant field of research. Opportunistic networks allow a flexible and highly dynamic connection between the nodes. Any node can join or leave the network at any time. The applications of opportunistic networks are cellular network offloading, communication in challenged areas, censorship circumvention and proximity-based applications and Internet of Things (IoT) [28], among others. Topology network is continuously changing due to the constant movement of the nodes, and the communication routes between senders and receivers are neither direct nor static. This communication capability allows the use of opportunistic networks in new applications. Before opportunistic networks, applications based their operation on an end-to-end connection path, however, when such connection is not possible, opportunistic networks present a solution, since the information is sent "opportunistically" hop by hop between source to destination using the "Store-Carry-and-Forward" approach [29].

3.1.1. Mobile Ad Hoc Networks Oriented Opportunistic Network

Mobile Ad hoc Network (MANET) is a group of nodes in a wireless communication network which dynamically forms temporary networks without the use of any existing infrastructure or centralized control. All the nodes are free to join or leave the network at any point of time. Most of the nodes in the network keep changing their position throughout. As the network is decentralized, the network organization and message delivery must be executed by the nodes themselves. Every node must be able to act as a host and as a router to forward the packets for other mobile nodes. As the nodes change their position continuously entering and leaving the network, the topology of the network remains dynamic. The topology of MANETs does not remain static (Fig.1) most of the time and keeps on changing with the movement of each mobile device in the network.

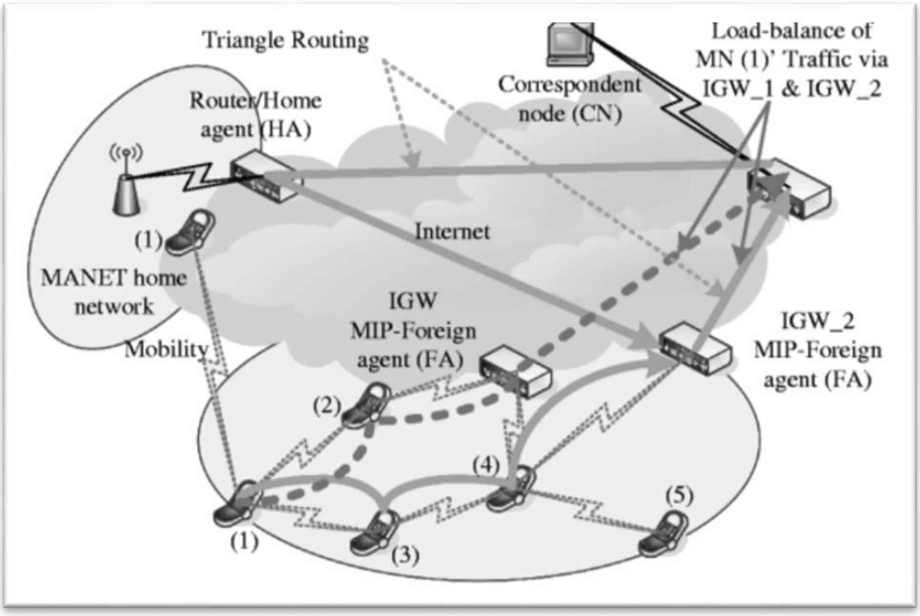


Figure 1. Mobile Adhoc Networks

Challenges in MANETs

Although MANETs have brought in a lot of advantages to set up new applications, a number of issues still remain to be addressed. Unreliable wireless links and the dynamic network topology have been the two most important challenges faced in MANETs. Moreover in highly dynamic and fast changing MANETs, routing of packets with reliable and efficient data delivery remains an open challenge. Over these years a number of protocols have been proposed to solve this routing problem in wireless networks. Recent studies shows that most of the routing protocols that follows traditional routing techniques like DSR, DSDV, AODV [13] and TORA [14-15] encounter numerous difficulties in handling with unpredictable wireless medium and random moving nodes. Most of these protocols relies on fixing a route before the data transmission and thus depends much more on the topology of the network. But in fast changing and reconfiguring MANETs, it is not possible to have a definite topology or definite route to send the packets. Once the path breaks, the data packet will be lost and considerable time and energy would be lost for the discovery and recovery procedures. So reliable and efficient delivery of data packets are not addressed by these protocols in fast changing mobile ad hoc networks.

The stringent challenges of MANET are mentioned below:

• Lack of centralized management: MANET doesn't have a centralized 3.3monitor server. The absence of management makes the detection of attacks difficult because it is difficult to monitor the traffic in a highly dynamic and large-scale ad-hoc network. Lack of centralized management will impede trust management for nodes.

- Routing in adhoc topology: In MANET, the mobility of nodes changes the connectivity between them frequently. Hence, an efficient routing protocol with link stability to be implemented to address this issue to ensure an efficient data packet delivery from source to destination node, thereby improving the throughput.

- Maintenance of routing: Regular update of information corresponding to the data packets and the control packets with dynamic links between the nodes in MANET is a major issue to be addressed.

- Scalability: In MANETs, the nodes that establish a network are battery powered. Hence, the throughput of the network depends on the parameters namely, input source, processing capability and storage capacity. As the density of nodes increases the battery drain, thereby affecting the lifetime of the network.

- Energy Efficiency: The nodes employed in MANETs are battery operated with limited lifetime. The battery of a node is crucial Also, whenever a node gets damaged; recharging or replacing the battery is hard. Further the nodes act as routers and also as an end device. Hence, a routing protocol to be designed to increase the throughput thereby utilizing minimum energy resource.

With all above mentioned constraints, mobility has been a major limitation to a smooth functioning of a MANET protocol. The increase in node mobility causes increased link distraction and consequently, the performance of the protocol deteriorates as the density of the nodes in the network increases. Further, there exist more control packets generation and the increased control overhead in the network. Furthermore, in a real scenario, mobility of the wireless node is tied to a path fading effect, which further deteriorates the network performance.

Applications of MANET

Military services: Military services are one of the critical areas of application, where deployment of any fixed infrastructure is difficult in the rival territories. In such situation, the end devices of the network are connected to the soldiers whereby the soldiers act as mobile nodes and establish connectivity within the radio range of communication. Also, coordination between the military object and the workforce in the battle field.

Sensor Network: It is a part of MANET where the nodes are stationary in a defined terrain. The limitation of the network is that the nodes are battery powered and hence, has constrained of energy to be addressed. Here, each sensor nodes are equipped with transceiver, a microcontroller and a battery as source. The sensor nodes relay the observed data corresponding to physical parameter to the access point which is a monitor through intermediate nodes or directly to the access point if it placed within the radio range of the sensor node. Some of the physical parameter that can be measured is temperature, humidity, pressure etc. Some of the sensors are also attached to the patients in biomedical applications.

Personal Area Network: The communication of data within a distance of 10 meters is established through PAN. The devices used to communicate can be a mobile phone or laptops etc. It can also be used for adhoc communication among the devices or for connection to a backbone network.

Education: Sensors are equipped for video conferencing. A network is established to monitor the campus of an educational institution or a university. A virtual class rooms can also be established using sensor nodes.

Emergency services: The natural disasters where entire communication collapses and in that scenario restoring the communication quickly is essential. By adopting adhoc network, one could establish the setup within few days required for wireless communication. Thus, a robust protocol capable of routing effectively within a highly mobile topology and without compromising its inherent attributes is important for efficient deployment of a MANET. In other words, an improved routing protocol is needed to be designed, in which the protocol keeps information about the speed of the intermediate nodes and adopts this information to establish a more stable routing path with the inclusion of minimal overhead.

3.1.2. DTN oriented opportunistic Network

Delay-tolerant networks (DTNs) [30] are partitioned wireless ad hoc networks with intermittent connectivity. Additional terminology in this family of dynamic networks includes disruption-tolerant networks, intermittently connected networks, and opportunistic networks.

DTNs are never fully connected at any point in time, but points of disconnection may be predictable as in vehicular networks following transportation schedules or networks with satellites traversing orbits [31]. In an intermittently connected network (ICMAN) or an opportunistic network, nodes rarely have information on the changing network topology Nodes may not know the availability of future encounters, but the network may benefit from learning such patterns over time. Thus, subsets of node s in transmission range leverage cooperation during pairwise contacts to forward data towards a destination. DTN evolved from the Mobile Ad hoc Networks (MANET) by relaxing some of the requirements of MANET and by allowing a high degree of mobility to the participating nodes (Fig. 1). MANET and DTN share many common aspects, such as, lack of infrastructure, resource constrains, node mobility resulting in frequent network partitioning, etc. However, they have multiple differences as well. While MANET nodes communicate using standard TCP/IP protocol suite, DTN nodes use application-layer Bundle protocols, designed to support the store-carry-forward communication paradigm of DTN, to communicate with peers. Bundle layer protocol span across transport and network layers: storing limited bundles inside the network and facilitating late binding of DTN address to subnet address (Franck, 2015). Network partition caused by node mobility results in packet loss in MANET whereas, it stretches the span of DTN as routing through frequent disconnection is an inherent property of DTN routing protocols. Finding the appropriate relay node may take long time in DTN which requires message carriers to store the message in their buffers for considerably long time.

Challenges of DTN oriented opportunistic Network

Routing in DTNs is non-trivial due to the following challenges. Firstly, DTNs has no pre-existing infrastructure like wired and wireless infrastructure-based networks. Even infrastructure-less MANETs computes end-to-end routing paths which are, instead, inherently non-existent in DTNs. So, traditional MANET protocols like AODV (Perkins et al., 2003), DSR (Johnson et al., 1996), etc. cannot be directly applied to DTN because they require the existence of a fully connected graph to route a message and fail otherwise. Secondly, DTN nodes store messages until they find a relay node to forward that message towards the destination. Finding the relay node is crucial to DTN routing as it directly affects performance in terms of delay in message delivery and may even result in message loss. Due to the intermittent connectivity and long delay in communications, it is often difficult to develop an optimal forwarding strategy for routing in DTN.

3.2. *Opportunistic Routing*

A major breakthrough in routing in mobile ad hoc networks was provided with opportunistic routing [20] and opportunistic data forwarding. Opportunistic routing makes use of the broadcasting property of the wireless networks to improve the efficiency of data delivery in the network. In wireless networks, when a data packet is send out, it is received by all other nodes within the transmission range on that channel. This property was often considered interference and a demerit of the wireless medium during these years. Opportunistic routing exploits the advantages brought about by this property. The concept of opportunistic routing [21] does not commit to a fixed route before data transmission. When a sender node wants to send a data packet to a particular destination device, it broadcasts the data packet into the network. This packet is received by every node within its transmission using MAC interception. The device that is closest to the destination is selected as the best forwarder device. This device forwards the data packet in a similar manner till it reaches the destination. Thus, data forwarding remains dynamic and there is no need to maintain predetermined routes. Thus, opportunistic routing provides an improved mechanism for the delivery of the data packet at the destination especially in ad hoc networks with highly mobile nodes. A number of opportunistic routing protocols have been proposed over these years. Every protocol has tried to address one or more issues involving data transfer in ad hoc networks. One of the most important issues that have not been studied so far is the varying performance of opportunistic routing protocols in wireless networks with highly mobile nodes. This project analyses and compares the various advantages, disadvantages and the performance of the latest opportunistic routing protocols in wireless ad hoc networks with highly mobile nodes.

We show comparisons of sample tests of routing protocols in the One simulator within an heterogenous environment to demonstrate the simulator's flexible support for opportunistic routing protocol evaluation. The following opportunistic routing protocols we work with are as follows:

Direct Delivery: The router is not used to forward the packet because the destination is on the same network (subnet or network segment) as the sending host. In Direct Delivery, the node carries messages until it meets their final destination. In widely used networks such as Internet or cellular networks, connectivity between a source node and a destination node should always be maintained in order to deliver data or message from a source node to a destination node. If connectivity fails due to any reason, however, a communication session between them is broken and message delivery is not possible any more.

First Contact: Direct Delivery and First Contact are single-copy routing protocols where only one copy of each message exists in the network. In Direct Delivery, the node carries messages until it meets their final destination. In First Contact routing the nodes forward messages to the first node they encounter, which results in a "random walk" search for the destination node.

Epidemic: The Epidemic protocol [32] is based on general broadcasting of messages: nodes freely replicate messages on each encounter until a message has reached a predefined maximum hop count. Messages are not exchanged if a copy is already present in the peer's buffer. Because it is essentially a flooding protocol, Epidemic was shown to have a good packet delivery ratio, but it suffers from very high overhead given the large number of packet copies flooding the network. Although buffer congestion issues have not been addressed in the protocol's design, the authors empirically investigate the impact of buffer size on successful delivery.

Spray-and-Wait: Spray-and-Wait [23,24] is an n-copy routing protocol that limits the number of message copies created to a configurable maximum and distributes ("sprays") these copies to contacts until the number of copies is exhausted. Both variants of Spray-and-Wait suggested by its authors are included: in normal mode, a node gives one copy to a contact, in binary mode half of the copies are forwarded. Once only a single copy is left, it is forwarded only to the final recipient. The Spray and Wait protocol outperforms all schemes discussed in including Epidemic for a large range of network connectivity scenarios. It is shown to perform close to the optimal oracle scheme (which has complete knowledge of future node encounters, i.e. future states of the wireless graph) for a random waypoint mobility model. The algorithm consists of two phases: spray and wait. During the spray phase, L packet copies are "sprayed" to relays in the network. Then these carriers enter the wait phase until they meet the destination and the message is delivered.

PRoPHET: In [33] the authors use historical throughput and historical contact time to calculate forward probability and a message is forwarded to a node with a higher forward probability. The forwarding probability has similar characteristics with delivery predictability in PRoPHET protocol, i.e., increase for each contact, decay after contact, and transitivity. In [26], contact history is managed in a table and a message forwarding is determined based on the contact existence with the destination node in the same time zone when the two nodes contact. In [27], in the early phase of message dissemination, a message is forwarded to a node with a higher chance of message dissemination. On the other hand, in the latter phase of message dissemination, message is forwarded to a node with higher chance of message delivery to the destination node. In [28], reachable probability is calculated based on the contact history with the destination node and a message is forwarded to a node with a higher probability value.

MaxProp: MaxProp floods the messages but explicitly clears them once a copy gets delivered to the destination. In addition, MaxProp sends messages to other hosts in specific order that takes into account message hop counts and message delivery probabilities based on previous encounters. In an effort to increase the delivery rate and reduce latency, the MaxProp protocol prioritizes buffered packets for retransmission. Packets with lower hop counts are given priority in order to facilitate quick propagation through the network. Once packets exceed the hop count threshold, packet prioritization is determined by the probability that two peers meet calculated using incremental averaging [30]. Acknowledgements are also utilized to delete replicated messages that have already been delivered. This prioritized delivery scheme has been shown to timely deliver packets at vehicular speeds and with tight constraints on buffer spaces. In this case another version of a "contact graph" is considered with edge weights given by the probability that two nodes meet.

3.3. Simulations

This project is highly focused on the process of network routing simulation and its key elements, real-time applications and through this article, we provide you the best simulation tools and our experience in tool implementation and the possible project titles in network simulation.

Simulation is a technique where a software program models the behaviour of a network either by calculating the interaction between the different network entities (hosts/packets, etc.) using mathematical formulae or capturing and playing back observations from a production network. It provides a virtual environment wherein the characteristic of the network is analysed before deploying in the real scenario or environment. The behaviour of the network and the various applications and services it supports can then be observed in a test lab [31].

Using a high performance, high fidelity discrete event simulation engine, extremely accurate virtual models of a communications environment can be created, and then analysed through a series of scenarios to identify where there are weak points or failure modes. This lab-based risk reduction methodology is repeatable, verifiable and highly cost effective.

With the help of efficient simulation models, Randomness and Probabilities can be introduced, which makes the simulation model more realistic. It is sometimes possible to design simulation models to find the optimal solution with a high level of assurance.

Advantages of simulation model

- Simulation results are reproducible compared to direct measurement results.
- All performance parameters are monitored.
- It provides a better approach and visualizing effects of performance evaluation.

Disadvantages of simulation model

- A long simulation time is required in order to achieve a desired statistical accuracy for some experiments; in some cases it is impossible to get meaningful results within reasonable times.
- To cover a meaningful fraction of the whole parameter space.
- Model setup may take a long time, also validation and verification time can be significant. Validation makes sure that the assumptions behind the model 33 suit the real system's behaviour and verification confirms that the actual code of the simulation model fits the claimed model assumptions.

3.3.1. Simulators for Mobile Opportunistic Networks

MONICA : It provides a programmable interface to users, which is easy to learn and easy to use. It also integrates eight classic routing algorithms, six caching strategies and two types of mobility models, so that users can make a more comprehensive evaluation of the routing protocol. In addition, MONICA can trace abnormal behaviors with the network event function and observe the real simulation process by the visual module. All of these guarantee the efficiency of the simulator in evaluating the routing performance. MONICA based on the

Microsoft visual C++ platform, it integrates a variety of classic opportunistic protocols and mobility models. MONICA integrates 2 kinds of mobility models, 8 routing algorithms, 6 caching strategies, one energy consumption model, as well as the interface defined by the extension function. In addition, a dynamic visualization window permits users to observe the real process.

OMNeT++0: OMNeT++[32] is an open-source modular simulation platform that has primarily been used for simulating wired and wireless communication networks. It includes, and is continuously complemented by, multiple modeling frameworks like INET, INETMANET, MiXiM, etc. The idea of this work is based on the fact that OMNeT++ (including INETMANET) lacks support for some of the key features of opportunistic routing protocols.

Adyton: Adyton is an event-driven network simulator, written in C++, for Opportunistic Networks (a.k.a. Delay-Tolerant Networks) that is capable of processing contact traces. The Adyton simulator supports a plethora of routing protocols and real-world contact traces, while also providing several congestion control mechanisms and buffer management policies.

ONE: One is a Java based simulator targeted for research in Delay Tolerant Networks (DTNs). The ONE simulator has been developed in the SINDTN and CATDTN projects supported by Nokia Research Center (Finland). This simulation environment written in Java is completely configurable and is able to completely simulate the behavior of nodes in the simulations that includes movement of nodes, connections between nodes and provides complete routing information of each node. A more interesting feature, for our purposes, is the ability of emulating message routing using different routing protocols.

At its core, ONE is an agent-based discrete event simulation engine. At each simulation step the engine updates a number of modules that implement the main simulation functions.

The main functions of the ONE simulator are the modeling of node movement, inter-node contacts, routing and message handling. Result collection and analysis are done through visualization, reports and post-processing tools. The elements and their interactions are shown in figure 1. A detailed description of the simulator is available in [33] and the ONE simulator project page where the source code is also available.

Node movement is implemented by movement models. These are either synthetic models or existing movement traces. Connectivity between the nodes is based on their location, communication range and the bit-rate. The routing function is implemented by routing modules that decide which messages to forward over existing contacts. Finally, the messages themselves are generated through event generators. The messages are always unicast, having a single source and destination host inside the simulation world.

Simulation results are collected primarily through reports generated by report modules during the simulation run. Report modules receive events (e.g., message or connectivity events) from the simulation engine and generate results based on them. The results generated may be logs of events that are then further processed by the external post-processing tools, or they may be aggregate statistics calculated in the simulator. Secondarily, the graphical user interface (GUI) displays a visualization of the simulation state showing the locations, active contacts and messages carried by the nodes[33-39].

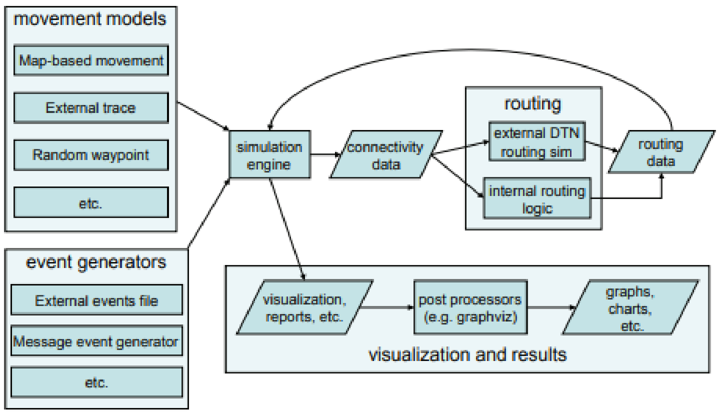


Figure 2. Overview of the ONE simulation environment

4. ONE SIMULATOR

4.1. Introduction

The Opportunistic Network Environment (ONE) is a Java based simulator targeted for research in Delay Tolerant Networks (DTNs) and its variants such as, Opportunistic Mobile Networks (OMNs). Apart from letting users simulate different scenarios quickly and in a flexible manner, the ONE also provides an easy way to generate statistics from the simulation(s) performed. The ONE simulator can be run on Linux, Windows, or any other platform supporting Java.

4.2. General Steps

Below are the steps that we will be following to perform our simulation. In fact, these are some steps that every simulation would involve.

1. Specify the general settings for the scenario to be simulated
2. Specify the network interface(s) of the nodes
3. Create a group of nodes
4. Specify the motion patterns
5. Specify the traffic pattern
6. Specify a set of reports to be generated

4.3. Configuration File(s)

Configuration file(s) define various settings for our simulation, in this(these) file(s) we would specify the number of nodes that our simulation would consist of, dimensions of the bounding region, how long the simulation would run, and so on.

ONE comes with a default_settings.txt file. It is important to note here that this file is read (used) for every simulation. Thus, we would, in general, be modifying this file to suit our purpose. However, when one goes for simulating a very complex scenario, it is better to take a modular approach. In that case we can have multiple configuration files. To reflect on this, consider a few group of nodes with each group using different routing algorithms. To make life simple, we can store parameters for each such algorithms in different files. We just need to specify names of these files while running the simulation.

4.4. Experiment Configuration of ONE Simulator

There are several different configurations based upon different scenario, let's look at one such specific scenario.

4.4.1. Scenario with 30 Nodes

A simulation of opportunistic network with total 30 nodes (20 groups) with high heterogeneity using EPIDEMIC algorithm.

4.4.2. Specifying Scenario Settings

The first step of a simulation is to setup the scenario. For example, how long would the simulation run, how many groups of nodes would it consist of, a name for the current scenario, traffic pattern, mobility pattern, types of interfaces, communication range and so on. The listing below shows some of the specifications together with their explanations as comment lines.

Note: In this project report all the configuration has been written in the default_settings.txt file.

```
## Scenario settings
Scenario.name = default_scenario
Scenario.simulateConnections = true
Scenario.updateInterval = 0.1
# 43200s == 12h
Scenario.endTime = 10000000
Scenario.endTime = 43200
```

We can create multiple groups of nodes, with each group possibly having it's own characteristics. Even if such a scenario is not required, we would create a single logical group and put all the nodes under it.

```
# Define 20 different node groups
Scenario.nrofHostGroups = 20
```

4.4.3. *Specifying Network Interfaces for Nodes*

We have to instruct the simulator about how the different nodes will communicate. For example, two computers can communicate over, say, Ethernet or P2P links. In our case, all the nodes would exchange messages using Bluetooth and/or Wifi as shown below.

```
## Interface-specific settings:
# type : which interface class the interface belongs to
# For different types, the sub-parameters are interface-specific
# For SimpleBroadcastInterface, the parameters are:
# transmitSpeed : transmit speed of the interface (bytes per second)
# transmitRange : range of the interface (meters)

# "Bluetooth1" interface for all nodes
btInterface1.type = SimpleBroadcastInterface
# Transmit speed of 2 Mbps = 250kBps
btInterface1.transmitSpeed = 500k
btInterface1.transmitRange = 20

# "Bluetooth2" interface for all nodes
btInterface2.type = SimpleBroadcastInterface
# Transmit speed of 2 Mbps = 250kBps
btInterface2.transmitSpeed = 1M
btInterface2.transmitRange = 15
# "Bluetooth3" interface for all nodes
btInterface3.type = SimpleBroadcastInterface
# Transmit speed of 2 Mbps = 250kBps
btInterface3.transmitSpeed = 2M
btInterface3.transmitRange = 10
```

```
wlanInterface1.type = DistanceCapacityInterface

# values from http://www.xirrus.com/cdn/pdf/wifi-demystified/documents_posters_range_plotter
# 0-50ft:54Mbps, 75-100ft:48 Mbps, 125ft:36Mbps, 150ft:24 Mbps, 175ft:18Mbps,
# 200ft:12Mbps, 225ft:9Mbps, 250ft:6Mbps, 275ft:2Mbps, 300ft:1Mbps

wlanInterface1.transmitSpeeds = 6000k, 6000k, 6000k, 4500k, 4500k, 3000k, 2250k, 1500k, 1125k, 750k, 250k, 125k, 70k
wlanInterface1.transmitRange = 120

# dummy speed
wlanInterface1.transmitSpeed = 0

wlanInterface2.type = DistanceCapacityInterface

# values from http://www.xirrus.com/cdn/pdf/wifi-demystified/documents_posters_range_plotter
# 0-50ft:54Mbps, 75-100ft:48 Mbps, 125ft:36Mbps, 150ft:24 Mbps, 175ft:18Mbps,
# 200ft:12Mbps, 225ft:9Mbps, 250ft:6Mbps, 275ft:2Mbps, 300ft:1Mbps

wlanInterface2.transmitSpeeds = 6750k, 6750k, 6750k, 6000k, 6000k, 4500k, 3000k, 2250k, 1500k, 1125k, 750k, 250k, 125k
wlanInterface2.transmitRange = 90
# dummy speed
wlanInterface2.transmitSpeed = 0
```

4.4.4. Creating the Groups of Nodes

As mentioned earlier, we can have multiple logical groups of nodes in ONE. For example, consider a rescue operation (after a disaster) performed by close coordination of army, health, fire brigade, and civil organizations. We can represent this situation with four groups, and define individual behavior of a group. However, in our case, to achieve extreme heterogeneity we have created 20 groups, each group differ from each other.

```
## Group-specific settings:

# groupID : Group's identifier. Used as the prefix of host names

# nrofHosts: number of hosts in the group

# movementModel: movement model of the hosts (valid class name from movement package)

# waitTime: minimum and maximum wait times (seconds) after reaching destination

# speed: minimum and maximum speeds (m/s) when moving on a path

# bufferSize: size of the message buffer (bytes)

# router: router used to route messages (valid class name from routing package)

# activeTimes: Time intervals when the nodes in the group are active (start1, end1, start2, end2, ...)

# msgTtl : TTL (minutes) of the messages created by this host group, default=infinite


## Group and movement model specific settings

# pois: Points Of Interest indexes and probabilities (poiIndex1, poiProb1, poiIndex2, poiProb2, ... )

#       for ShortestPathMapBasedMovement

# okMaps : which map nodes are OK for the group (map file indexes), default=all

#          for all MapBasedMovent models

# routeFile: route's file path - for MapRouteMovement

# routeType: route's type - for MapRouteMovement


# Common settings for all groups

Group.movementModel = ShortestPathMapBasedMovement

Group.router = EpidemicRouter

Group.bufferSize = 5M
```

```
Group.waitTime = 0, 120

# All nodes have the bluetooth interface

Group.nrofInterfaces = 1

Group.interface1 = btInterface1

# Walking speeds

Group.speed = 0.5, 1.5

# Message TTL of 300 minutes (5 hours)

Group.msgTtl = 300


Group.nrofHosts = 1


# The Pedestrian groups

# group1 (pedestrians) specific settings

Group1.groupID = p1-

Group1.speed = 0.5, 0.55

Group1.bufferSize = 4M

Group1.nrofInterfaces = 1

Group1.interface1 = btInterface1


# group2 (pedestrians) specific settings

Group2.groupID = p2-

Group2.speed = 0.55, 0.6

Group2.bufferSize = 6M

Group2.nrofInterfaces = 1

Group2.interface1 = btInterface2

Group2.nrofHosts = 2


# Group3 (pedestrians) specific settings

Group3.groupID = p3-

Group3.speed = 0.6, 0.65
```

```
Group3.bufferSize = 12M
Group3.nrofInterfaces = 1
Group3.interface1 = btInterface2

# Group4 (pedestrians) specific settings
Group4.groupID = p4-
Group4.speed = 0.65, 0.7
Group4.bufferSize = 10M
Group4.nrofInterfaces = 1
Group4.interface1 = btInterface2
Group4.nrofHosts = 2

# Group5 (pedestrians) specific settings
Group5.groupID = p5-
Group5.speed = 0.7, 0.75
Group5.bufferSize = 8M
Group5.nrofInterfaces = 1
Group5.interface1 = btInterface2

# Group6 (pedestrians) specific settings
Group6.groupID = p6-
Group6.speed = 0.75, 0.8
Group6.bufferSize = 12M
Group6.nrofInterfaces = 1
Group6.interface1 = btInterface2
Group6.nrofHosts = 2

# Group7 (pedestrians) specific settings
Group7.groupID = p7-
Group7.speed = 0.8, 0.85
```

```
Group7.bufferSize = 6M

Group7.nrofInterfaces = 1

Group7.interface1 = btInterface1


# Group8 (pedestrians) specific settings

Group8.groupID = p8-

Group8.speed = 0.8, 0.85

Group8.bufferSize = 10M

Group8.nrofInterfaces = 1

Group8.interface1 = btInterface3

Group8.nrofHosts = 2


# Group9 (pedestrians) specific settings

Group9.groupID = p9-

Group9.speed = 0.8, 1

Group9.bufferSize = 8M

Group9.nrofInterfaces = 1

Group9.interface1 = btInterface2


# Group10 (pedestrians) specific settings

Group10.groupID = p10-

Group10.speed = 1, 1.15

Group10.bufferSize = 8M

Group10.nrofInterfaces = 1

Group10.interface1 = btInterface3

Group10.nrofHosts = 2


# Group11 (pedestrians) specific settings

Group11.groupID = p11-

Group11.speed = 1.15, 1.35
```

```
Group11.bufferSize = 10M

Group11.nrofInterfaces = 1

Group11.interface1 = btInterface3


# The Cars groups

# Group12 (Cars) specific settings

Group12.groupID = c1-

Group12.speed = 8.1, 12

Group12.okMaps = 1

Group12.bufferSize = 34M

Group12.nrofInterfaces = 1

Group12.interface1 = btInterface1

Group12.nrofHosts = 2


# Group13 (Cars) specific settings

Group13.groupID = c2-

Group13.speed = 12, 14

Group13.okMaps = 1

Group13.bufferSize = 36M

Group13.nrofInterfaces = 1

Group13.interface1 = btInterface2


# Group14 (Cars) specific settings

Group14.groupID = c3-

Group14.speed = 14, 17

Group14.okMaps = 1

Group14.bufferSize = 42M

Group14.nrofInterfaces = 1

Group14.interface1 = btInterface3

Group14.nrofHosts = 2
```

```
# Group15 (Cars) specific settings
Group15.groupID = c4-
Group15.speed = 17, 18.9
Group15.okMaps = 1
Group15.bufferSize = 44M
Group15.nrofInterfaces = 2
Group15.interface1 = btInterface3
Group15.interface2 = wlanInterface1

# The Bus groups
# Group16 (Bus) specific settings
Group16.groupID = b1-
Group16.speed = 5.4, 9.5
Group16.okMaps = 1
Group16.bufferSize = 28M
Group16.nrofInterfaces = 1
Group16.interface1 = btInterface3
Group16.nrofHosts = 2

# Group17 (Bus) specific settings
Group17.groupID = b2-
Group17.speed = 9.5, 12.15
Group17.okMaps = 1
Group17.bufferSize = 32M
Group17.nrofInterfaces = 2
Group17.interface1 = btInterface3
Group17.interface2 = wlanInterface1

# The Tram groups
```

```
# Group18 (Tram) specific settings

Group18.groupID = t1-

Group18.bufferSize = 48M

Group18.movementModel = MapRouteMovement

Group18.routeFile = data/tram10.wkt

Group18.routeType = 2

Group18.waitTime = 10, 30

Group18.speed = 6, 7.5

Group18.nrofInterfaces = 2

Group18.interface1 = btInterface3

Group18.interface2 = wlanInterface2

Group18.nrofHosts = 2


# Group19 (Tram) specific settings

Group19.groupID = t2-

Group19.bufferSize = 46M

Group19.movementModel = MapRouteMovement

Group19.routeFile = data/tram4.wkt

Group19.routeType = 2

Group19.waitTime = 10, 30

Group19.speed = 7.5, 9

Group19.nrofInterfaces = 2

Group19.interface1 = btInterface2

Group19.interface2 = wlanInterface2


# group20 (Tram) specific settings

Group20.groupID = t3-

Group20.bufferSize = 48M

Group20.movementModel = MapRouteMovement

Group20.routeFile = data/tram3.wkt
```

```
Group20.routeType = 1
Group20.waitTime = 10, 30
Group20.speed = 9, 10.5
Group20.nrofInterfaces = 2
Group20.interface1 = btInterface1
Group20.interface2 = wlanInterface1
Group20.nrofHosts = 2
```

4.4.5. Specifying Mobility Patterns

Individual groups can have their own mobility model. We have already specified in the previous section that all nodes would move randomly (RandomWaypoint model). However, we also need to specify the system boundary i.e. area of the (2D) space within which motion of nodes would remain confined to. We do that in the following way.

```
## Movement model settings
# seed for movement models' pseudo random number generator (default = 0)
MovementModel.rngSeed = 1
# World's size for Movement Models without implicit size (width, height; meters)
MovementModel.worldSize = 4500, 3400
# How long time to move hosts in the world before real simulation
MovementModel.warmup = 1000

## Map based movement -movement model specific settings
MapBasedMovement.nrofMapFiles = 4

MapBasedMovement.mapFile1 = data/roads.wkt
MapBasedMovement.mapFile2 = data/main_roads.wkt
MapBasedMovement.mapFile3 = data/pedestrian_paths.wkt
MapBasedMovement.mapFile4 = data/shops.wkt
```

4.4.6. Specifying Traffic Pattern

A network simulation is not of any practical use unless nodes exchange messages. ONE allows us to do this in two possible ways:

- Let ONE itself generate messages
- Load an external file containing a list of messages to be created (with timestamp, source node #, destination node #)

We use the first method here. The second method is quite useful if we wish to import data generated by any other application. ONE, however, also provides a PERL script to generate such traffics.

```
## Message creation parameters

# How many event generators

Events.nrof = 1

# Class of the first event generator

Events1.class = MessageEventGenerator

# (following settings are specific for the MessageEventGenerator
class)

# Creation interval in seconds (one new message every 25 to 35
seconds)

Events1.interval = 25,35

# Message sizes (500kB - 2MB)

Events1.size = 500k,2M

# range of message source/destination addresses

Events1.hosts = 1,2

Events1.tohosts = 5,6

# Message ID prefix

Events1.prefix = M
```

4.4.7. *Specifying Reports to be Generated*

ONE makes our work of network analysis simplified by generating a set of reports. For example, from MessageStatsReport we can find out the number of messages created, delivered, dropped, average buffer time, and so on. We will generate some of the important report for our simulation, as shown below.

```
## Reports - all report names have to be valid report classes

# how many reports to load

Report.nrofReports = 5

# length of the warm up period (simulated seconds)

Report.warmup = 0

# default directory of reports (can be overridden per Report with output setting)

Report.reportDir = reports/

# Report classes to load

Report.report1 = ContactTimesReport

Report.report2 = ConnectivityONEReport

Report.report3 = MessageStatsReport

Report.report4 = MessageDeliveryReport

Report.report5 = DeliveredMessagesReport
```

4.4.8. Running the Simulation

Once our settings file is ready, we can go ahead and run the simulation. If it doesn't throw any error, report files can be found in the '*reports/*' directory. Alternatively, we can simply execute one.bat without the *-b* argument. That will open the visualizer of ONE, where we can view motion of nodes', data exchange, and other different related events. A screenshot of ONE's GUI, can be seen in figure 3.

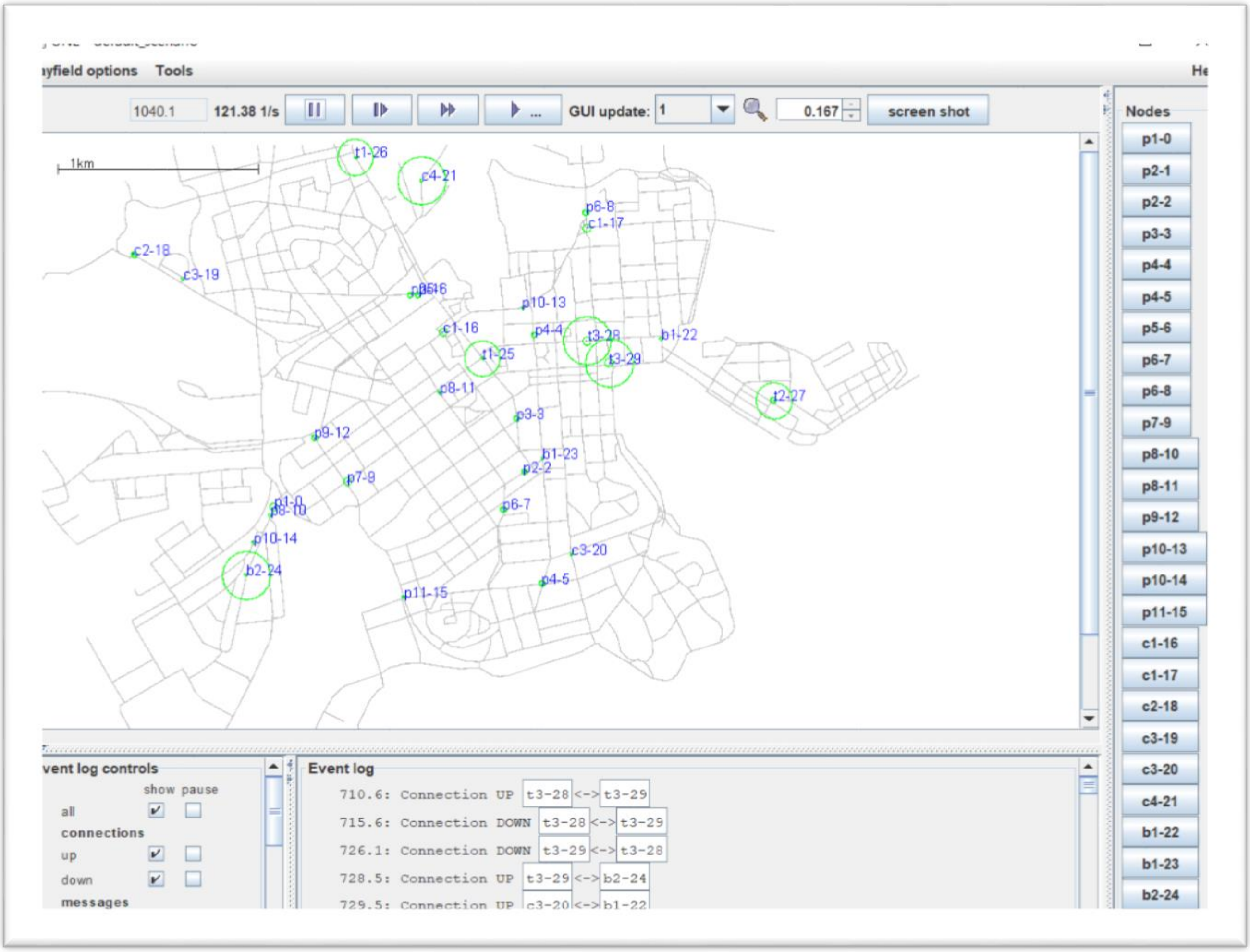
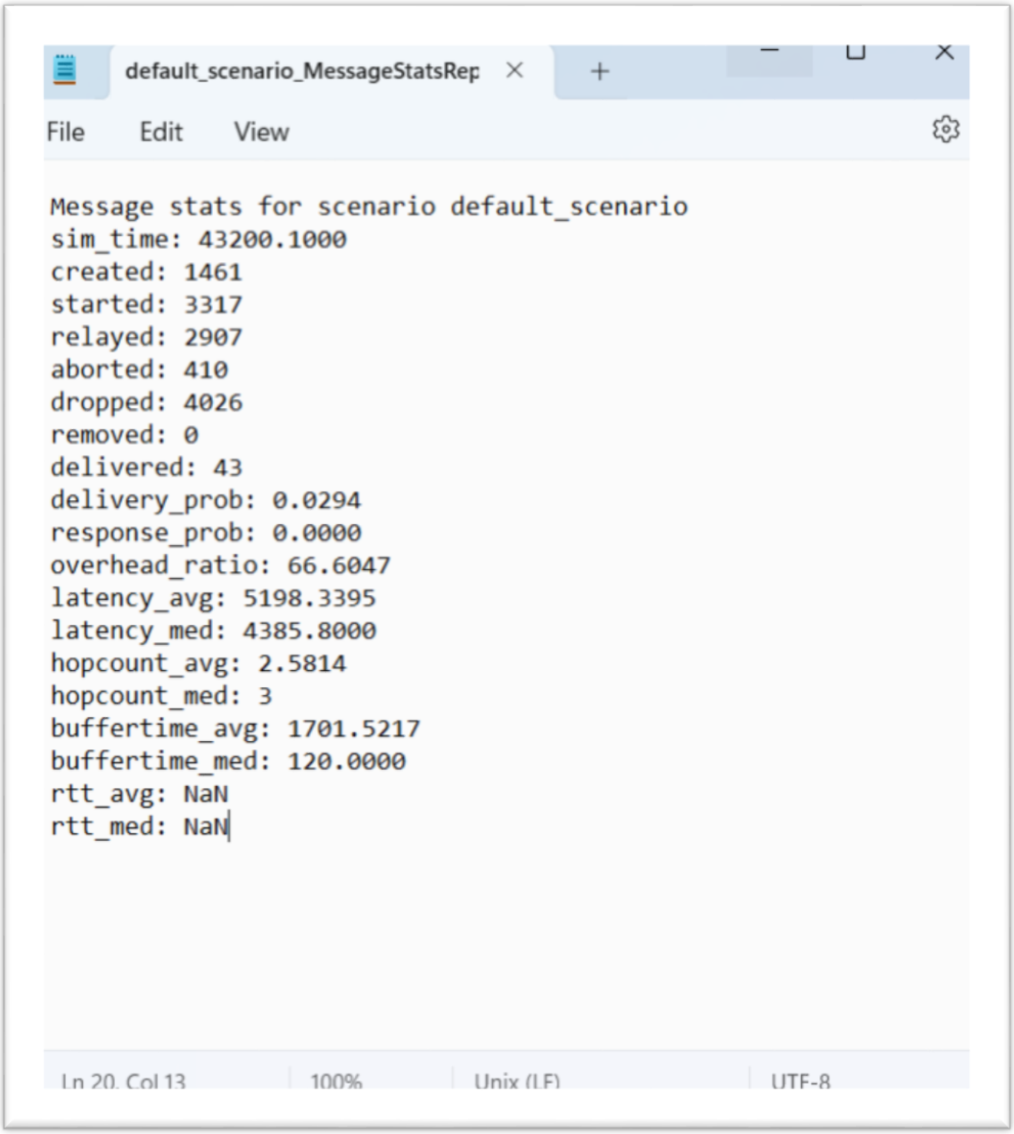


Figure 3. ONE Simulation GUI.

The report file obtained by running this simulation is attached below.




```
Message stats for scenario default_scenario
sim_time: 43200.1000
created: 1461
started: 3317
relayed: 2907
aborted: 410
dropped: 4026
removed: 0
delivered: 43
delivery_prob: 0.0294
response_prob: 0.0000
overhead_ratio: 66.6047
latency_avg: 5198.3395
latency_med: 4385.8000
hopcount_avg: 2.5814
hopcount_med: 3
buffertime_avg: 1701.5217
buffertime_med: 120.0000
rtt_avg: NaN
rtt_med: NaN
```


Figure 4. Report Generated by Simulation.

Similar to the 30 nodes (20 groups) scenario we have, 20 nodes (20 groups), 40 nodes (30 groups), 50 nodes (20 groups), 60 nodes (30 groups), 80 nodes (30 groups), 100 nodes (40 groups), 125 nodes (40 groups), 150 nodes (40 groups) scenarios.

We have run all the scenarios in Epidemic, Spray and Wait, MaxProp, First Contact and Direct Delivery, PRoPHET algorithm.

5. SIMULATION RESULTS AND ANALYSIS

We have carried out specific simulations to assess the impact of selected aspects of mobility modeling on the simulation results. These simulations are intended to compare or otherwise assess the performance of different protocols for the given scenario.

We compared the average delivery probability, average latency, average hop-count and overhead ratio in Epidemic, Spray and Wait, MaxProp, Direct Delivery, First Contact and PRoPHET routing to our approach. The main reasons for choosing these routing algorithms are following:

i. These are the most common and mostly cited algorithms used in opportunistic network.

ii. We consider them as representative routing protocols for opportunistic network because they encompass the basic techniques of routing like uncontrolled and controlled flooding, basic forwarding, and utility-based forwarding.

In our experiment we have worked on highly heterogenous environment. Now to achieve heterogeneity we have created 20 to 40 groups and each group differ from other in various ways. The parameters that we have handled in all the groups to create heterogeneity are following:

i. Movement Speed
ii. Interface
iii. Buffer size
iv. Transmission speed
v. Transmission range

5.1. Groups and Node capabilities

For the 20 groups, we have taken 12 pedestrian, 4 car, 2 bus and 2 tram and when there are 30 groups, we have taken 18 pedestrian, 7 car, 3 bus and 2 tram. Similarly for the 40 groups, we have taken 24 pedestrian, 8 car, 5 bus and 3 tram.

There are no more than 5 nodes per group, so whenever we increase the nodes, we had to increase the number of groups as well, otherwise the network would not be heterogenous.

In our project work we have taken single source and single destination for all the scenarios.

5.2. Message delivery strategy of routing algorithms

5.2.1. Epidemic

Epidemic [32] routing is flooding-based in nature, as nodes continuously replicate and transmit messages to newly discovered contacts that do not already possess a copy of the message.

5.2.2. PRoPHET

Epidemic routing is particularly resource hungry because it deliberately makes no attempt to eliminate replications that would be unlikely to improve the delivery probability of messages. This strategy is effective if the opportunistic encounters between nodes are purely random, but in realistic situations, encounters are rarely totally random. The Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) [33] protocol uses an algorithm that attempts to exploit the non-randomness of real-world.

5.2.3. MaxProp

MaxProp [30] is flooding-based in nature, in that if a contact is discovered, all messages not held by the contact will attempt to be replicated and transferred. The intelligence of MaxProp comes in determining which messages should be transmitted first and which messages should be dropped first. In essence, MaxProp maintains an ordered-queue based on the destination of each message, ordered by the estimated likelihood of a future transitive path (cost) to that destination.

5.2.4. Direct Delivery

This is one of the earliest proposed routing protocols, in this routing the source node only transmits the message when the source is in the range of the destination node. In opportunistic routing that rarely happens, so the delivery probability of the message reaching its destination is very low.

5.2.5. First Contact

First Contact is a forwarding based algorithm. In order to lessen the weaknesses of epidemic approach, and to mitigate the spread of messages, the First Contact routing protocol follows a very simple and very quick approach, for every message received or generated, this algorithm forwards a single copy of the bundle to the first node it meets in its path then it removes it from the local memory.

5.2.6. Spray and Wait

Spray and Wait [23] is a routing protocol that attempts to gain the delivery ratio benefits of replication-based routing as well as the low resource utilization benefits of forwarding-based routing Spray and Wait achieves resource efficiency by setting a strict upper bound on the number of copies per message allowed in the network.

5.3. Performance Metrics

To compare network routing protocols, several parameters must be tested. These parameters can describe the simulation results in terms of the performance metrics, variables, or input data of simulation such as models of nodes mobility, nodes resources etc, in a surface of simulation where the network is set up.

Among these metrics we report the following:

i. Average Delivery Probability
ii. Average Overhead Ratio
iii. Average Hop-count
iv. Average latency

5.3.1. Average Delivery ratio (Delivery probability)

It is the ratio of the total number of messages delivered to the destination and the total number of messages created at the source node.

$$DeliveryRatio = D/C$$

Bearing in mind that: D: Number of messages delivered to the destination.

C: Number of messages created at the source.

The results in Figure-5 shows the comparison of average message delivery probability for the six routing schemes.

			AVERAGE DELIVERY PROBABILITY			
TOTAL NO. OF NODES	EPIDEMIC	SPRAY & WAIT	MAXPROP	DIRECT-DELIVERY	FIRST-CONTACT	PROPHET
20	0.0068	0.0027	0.0137	0	0.0062	0.0055
30	0.0294	0.0349	0.0315	0.0075	0.0151	0.0205
40	0.0376	0.0137	0.0486	0	0.0171	0.0287
50	0.0205	0.026	0.0267	0.0034	0.0103	0.0287
60	0.0185	0.0014	0.0301	0	0.0089	0.0123
80	0.0287	0.0363	0.0445	0.0082	0.0185	0.037
100	0.0287	0.011	0.0479	0	0.0123	0.0411
125	0.0363	0.0568	0.0684	0.0055	0.0226	0.0493
150	0.0479	0.0589	0.0821	0.0062	0.0253	0.0465

Table 1. Average Delivery Probability of Routing Protocols.

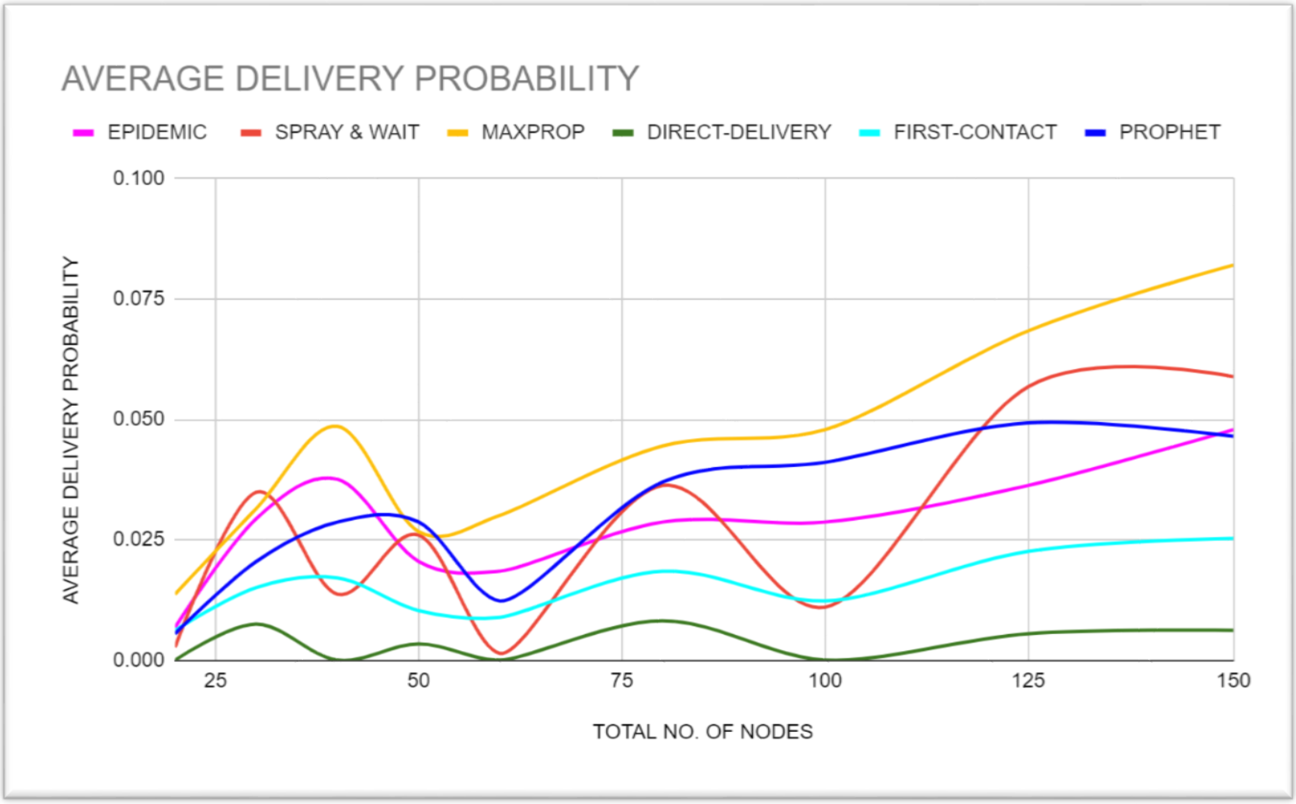


Figure 5. Average Delivery Probability vs No of Nodes of Routing Protocols.

From our simulation results we can verify that the MaxProp algorithm has the highest delivery probability followed by PRoPHET and Spray and Wait, also Direct Delivery have the lowest probability. If our main priority is delivering the message to the destination then we should use MaxProp over other five algorithms in highly heterogeneous environment.

It is evident by theory also that the MaxProp should have the highest probability and Direct Delivery should have the lowest probability. In MaxProp routing algorithm, these messages were forwarded using either one type of contacts or multiple contacts depending upon the priority of the message. Both scheduled and on-demand contacts provided very efficient message delivery mechanism and, therefore, MaxProp routing algorithm being capable of utilizing the existence of such contacts was better in delivering the messages. In Direct Delivery routing the source node only transmits the message when the source is in the range of the destination node. In opportunistic routing that rarely happens, so the delivery probability of the message reaching its destination is very low.

5.3.2. Average Overhead Ratio

This metric will allow us to evaluate the effectiveness of the bandwidth and interpret the number of copies created by a delivered message (it simply reflects the cost of transmission in a network). In other words, the number of replications required performing a successful delivery. For this purpose, we always look for algorithms that would minimize the value of overhead ratio.

$$OverheadRatio = (R - D)\ /D$$

Taking into account: R: Number of successful transmissions between nodes (This metric reflects the transmissions that took place).

D: Number of messages delivered to the destination.

We know that Overhead Ratio depends largely on the type of data diffusion techniques used for each opportunistic routing protocol. Theoretically the Epidemic algorithm should have the highest overhead ratio, as Epidemic routing is uncontrolled flooding-based in nature, nodes continuously replicate and transmit messages to newly discovered contacts that do not already possess a copy of the message. This means to reach one copy of the message to the destination Epidemic algorithm will create a very high number of copies. In Direct Delivery the message gets delivered only when it comes in contact with the destination node, there is no replication of that message so, overhead ratio is lowest.

			AVERAGE OVERHEAD RATIO			
TOTAL NO. OF NODES	EPIDEMIC	SPRAY & WAIT	MAXPROP	DIRECT-DELIVERY	FIRST-CONTACT	PROPHET
20	45.6	24.75	17.85	NaN	26.6667	18.625
30	66.6047	3.4706	22.3696	NaN	27.4545	7.4333
40	54.7091	20.35	25.8169	0	54.88	30.1667
50	104.6667	3.6842	27.7949	0	46.4667	15.5238
60	155.963	70.5	34.6591	NaN	48.4615	29.1111
80	204.6667	3.7736	72.7846	0	47.037	32.4444
100	484.7381	16.125	86.4714	NaN	114.5556	65.5
125	765.3774	3.7711	148.67	0	96.2424	158.0833
150	915.1143	4.9186	171.5417	0	139.8378	307.3235

Table 2. Average Overhead Ratio of Routing Protocols.

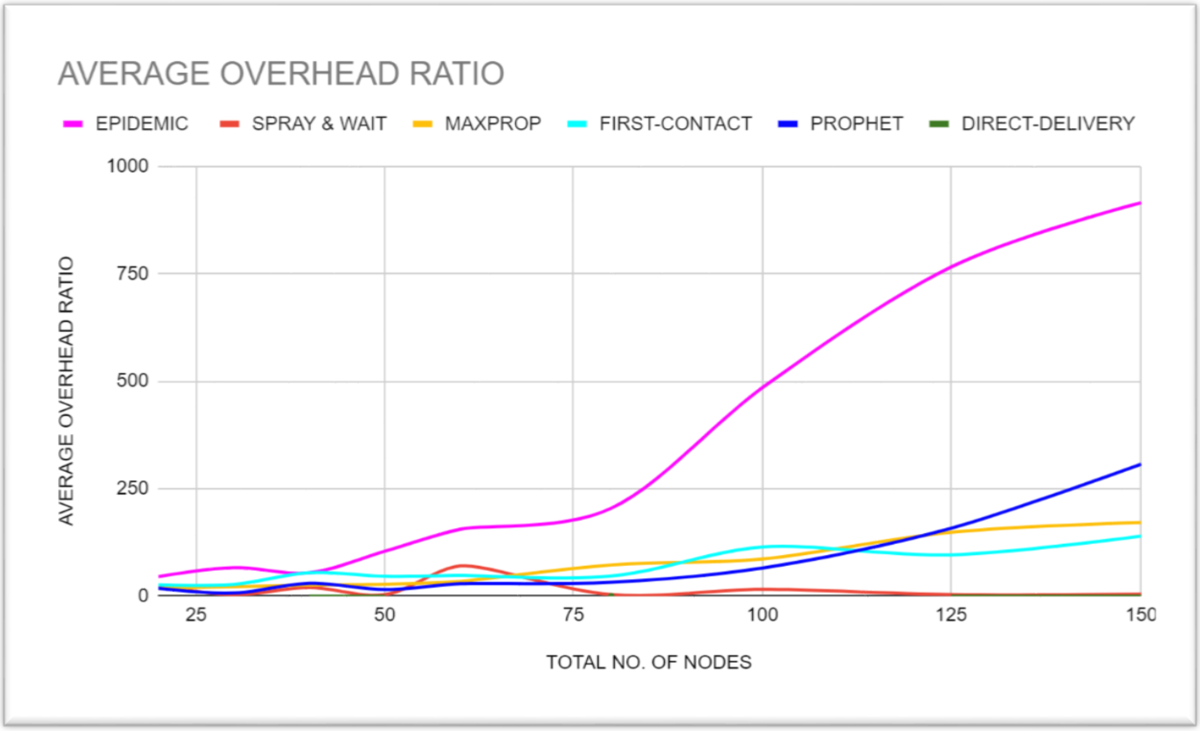


Figure 6. Average Overhead Ratio vs No of Nodes of Routing Protocols.

From our simulation results we can see that the Epidemic is the worst performing followed by ProPHET and Spray and Wait is the best performing among the five algorithms we have tested (excluding Direct Delivery) followed by First Contact and MaxProp. We can not see the Direct Delivery that well, because the delivery probability of Direct Delivery is very low in highly heterogeneous environment, so when Direct Delivery is not able to reach its destination, it is not possible to plot that algorithm properly.

5.3.3. Average Hop-count

Number of hops is a metric in DTN assessments which denotes the number of nodes by means of which the message must pass between the source and the destination node, it helps understanding how messages, along a path, must pass from the source to the destination or how the network resources have been used, etc. Thus, the information of the average number of hops tells us about the use of network resources.

The average hop count is an important metric for the interpretation and analysis of the routing performance according to the delivery rate and delivery time. Theoretically First

Contact should have the highest hop-count as it is a forwarding based algorithm so to reach the destination First Contact might require much numbers of forwarding than replication-based algorithms. Direct Delivery and Spray and Wait should have the lowest hop-count as they both have a limit to the maximum number of copies that can be present in the network.

			AVERAGE HOP COUNT			
TOTAL NO. OF NODES	EPIDEMIC	SPRAY & WAIT	MAXPROP	DIRECT-DELIVERY	FIRST-CONTACT	PROPHET
20	5.4	3.25	4.55	NaN	6.3333	4.625
30	2.5814	2.3137	2.587	1	2.2727	2.0333
40	5.2	3.4	5.0704	NaN	8.24	4.5
50	2.5	2.1842	2.9231	1	3.6667	2.6667
60	5.8148	4	5.9773	NaN	10.6923	4.6111
80	2.9286	2.0755	2.9538	1	4.0741	2.4444
100	5.9762	3.5625	5.7286	NaN	18.2222	4.9167
125	3.6792	2.6747	3.69	1	8.4848	3.125
150	3.9714	2.6395	3.9417	1	9.4865	3.1618

Table 3. Average Hop Count of Routing Protocols.

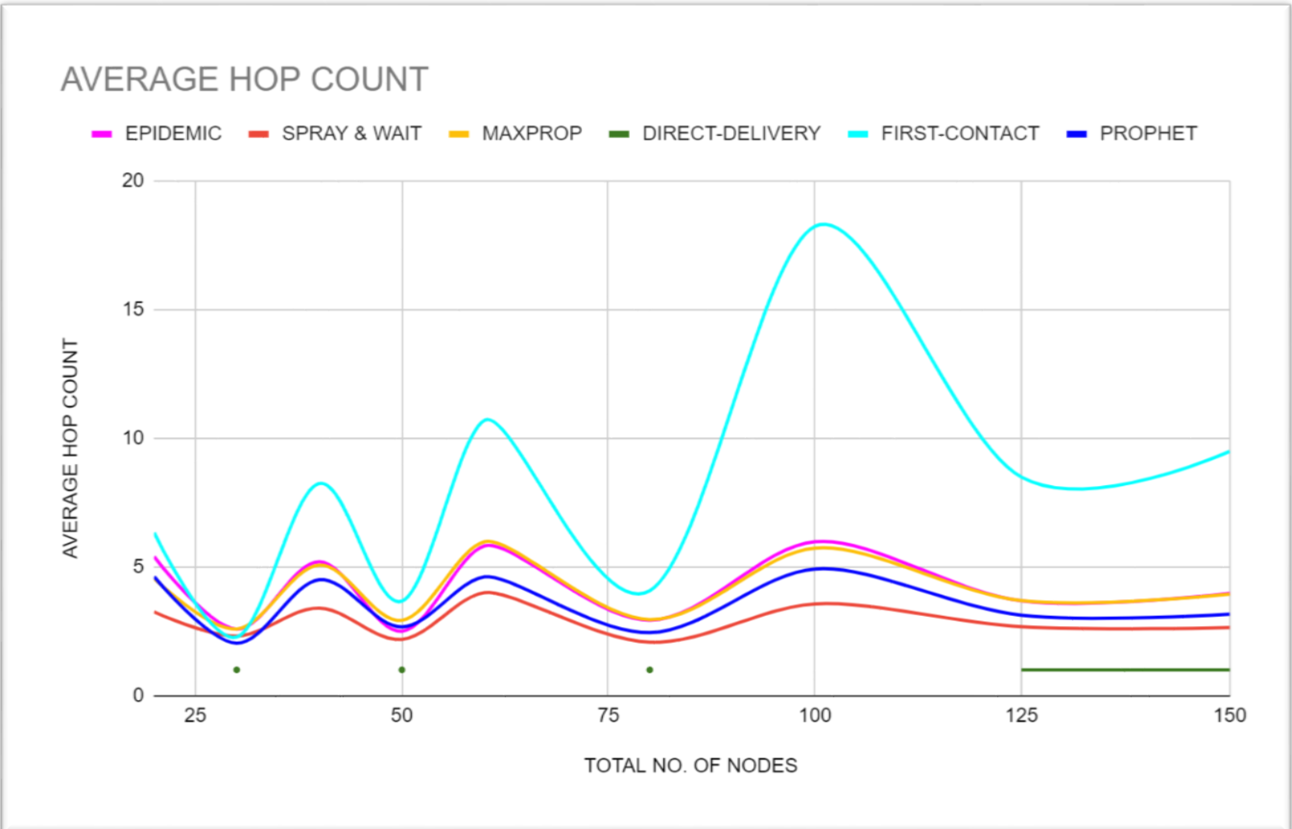


Figure 7. Average Hop Count of vs No of Nodes Routing Protocols.

From the figure 7, we can claim that the algorithms work as expected in highly heterogenous network as well with respect to average hop count against number of nodes.

5.3.4. Average latency

The latency measured here is the time that elapses between the creation of a message and its delivery at its destination.

			AVERAGE LATENCY			
TOTAL NO. OF NODES	EPIDEMIC	SPRAY & WAIT	MAXPROP	DIRECT-DELIVERY	FIRST-CONTACT	PROPHET
20	7629.81	7161.175	6597.325	NaN	8038.8222	7380.9375
30	5198.3395	4914.4902	6017.0848	NaN	3353.0364	2266.4667
40	6515.1909	4565.69	5473.3972	56.1273	6495.456	10385.8976
50	3478.6133	3795.5895	3939.4564	19.92	2787.3067	4665.9643
60	4477.0185	4434.55	6320.4455	NaN	6800.8846	7549.8278
80	4392.0571	6409.3962	5991.3723	33.4167	3285.2852	6248.0352
100	3932.1262	6610.9125	4446.4971	NaN	6268.5111	5321.5033
125	4360.8604	7136.8217	5075.624	34.4625	3699.0758	6807.8389
150	4033.1029	6073.6267	4291.4567	42.8111	4562.6568	5407.6706

Table 4. Average Latency of Routing Protocols.

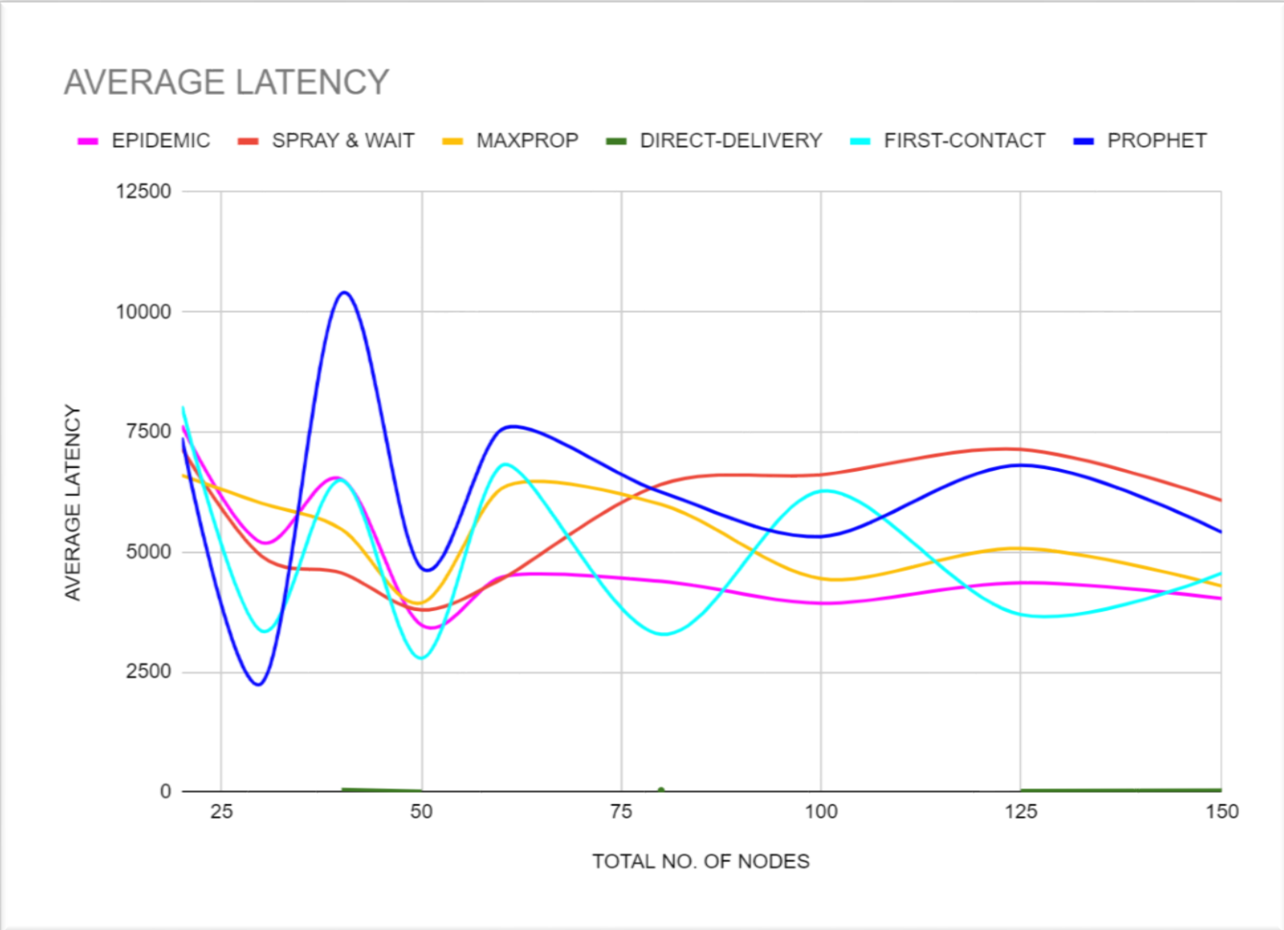


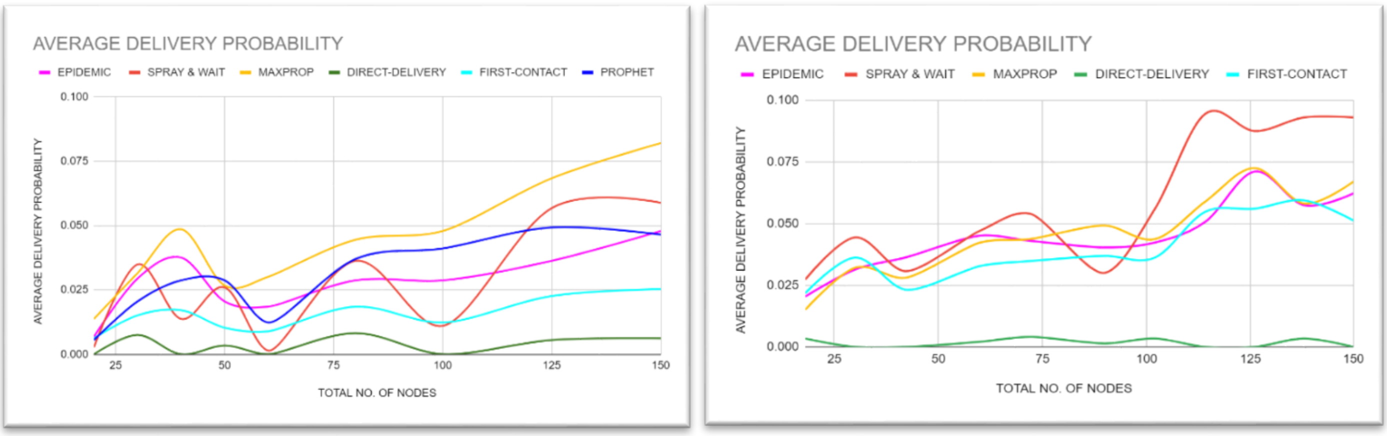
Figure 8. Average Latency of vs No of Nodes Routing Protocols.

As we can see from the fig that there is not much difference in the nature of plotting of output between all the algorithms, but a peak for PRoPHET is very much prominent in the range of 25 to 50 nodes that might be due to high heterogeneity of the configuration. If our priority is delivering the message as fast as possible, and we do not consider the wastage of resources then Epidemic algorithm is the best, and spray and wait is the worst, though the gap of average latency graph between them is not that much high.

5.4. Comparative Study

A comparative study has been presented with representative homogeneous scenario.

5.4.1. Average Delivery Probability

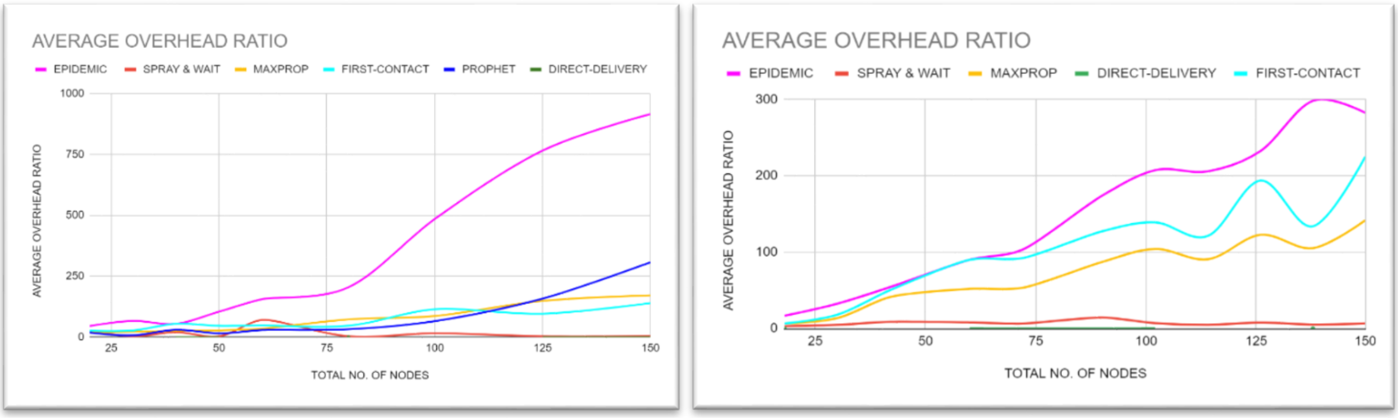


(A) (B)

Figure 9. Average Delivery Probability of Routing Protocols (A) Highly Heterogeneous vs (B) Minimum Heterogeneous.

As we can clearly see there is big difference between the minimum heterogeneous figure 9(A) and highly heterogeneous figure 9(B) result, in minimum heterogeneous Spray and Wait algorithm performed the best but in heterogeneous MaxProp is the best performing algorithm, in highly heterogeneous environment MaxProp shows steadily better performance from around 50 plus nodes, whereas in case of minimum heterogeneity up to 100 nodes all the five algorithm except Direct Delivery are performing at per but after 100 nodes Spray and Wait performs remarkably good. This is because, in minimum heterogeneous network the buffer each node has was very small, compared to highly heterogeneous network, that’s why algorithms like Epidemic and MaxProp suffered and Spray and Wait performed really well. But in highly heterogeneous network the buffer was more realistic, that’s why MaxProp performed the best, which follows the theory as well.

5.4.2. Average Overhead Ratio

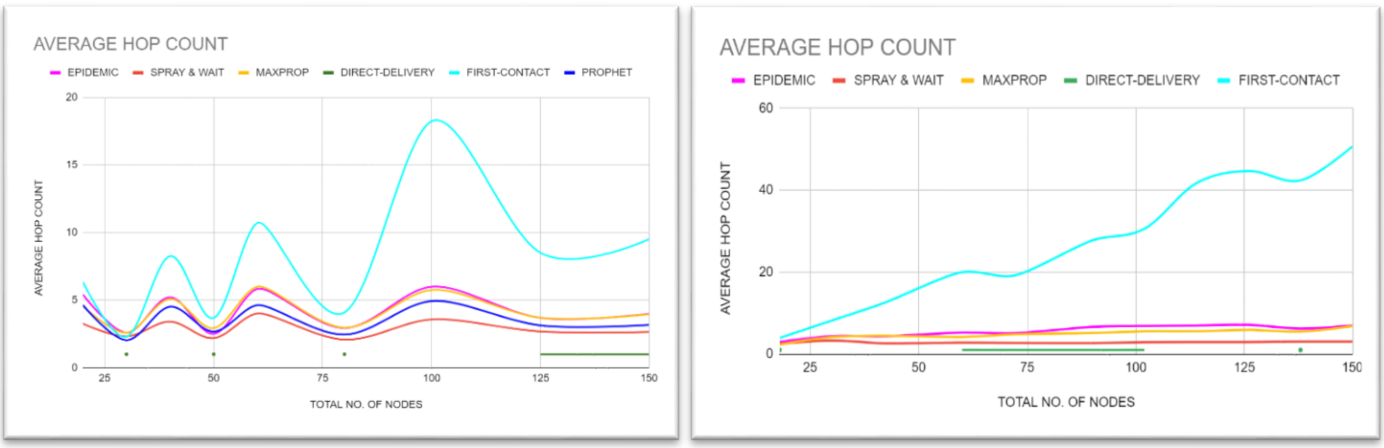


(A) (B)

Figure 10. Average Overhead Ratio of Routing Protocols (A) Highly Heterogeneous vs (B) Minimum Heterogeneous.

As we can see from both the figures 10(A) and 10(B) Epidemic is the worst performing algorithm and Spray and Wait is the best, which is as expected.

5.4.3. Average Hop Count

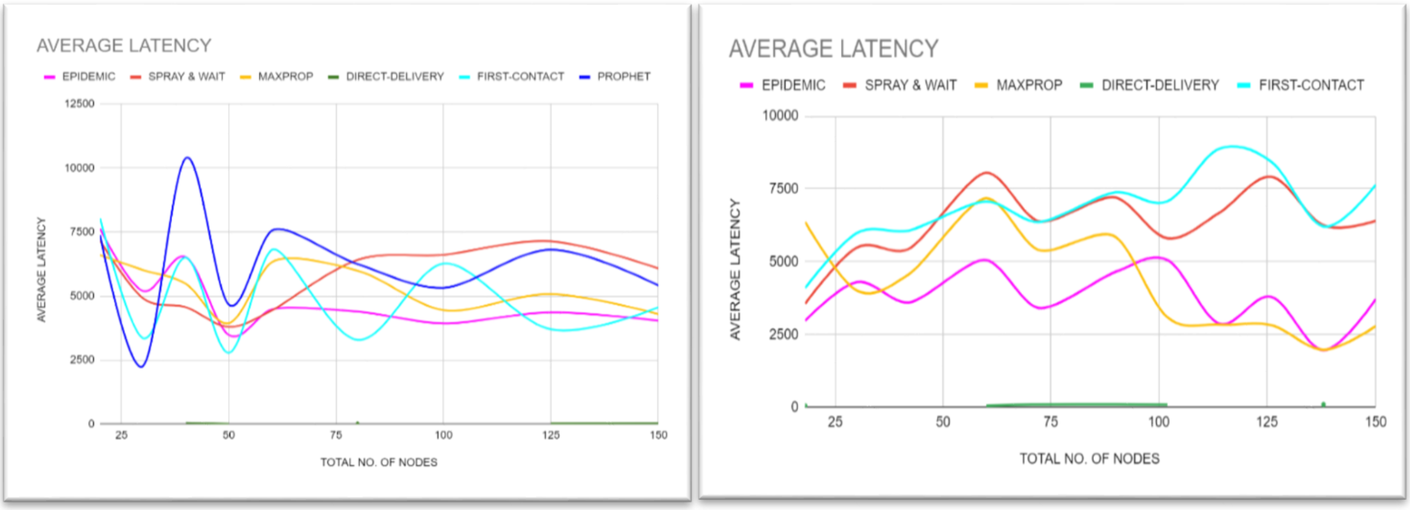


(A) (B)

Figure 11. Average Hop Count of Routing Protocols (A) Highly Heterogeneous vs (B) Minimum Heterogeneous.

From both the figure 11(A) and 11(B) we can clearly see that Spray and Wait delivers the messages with the lowest hop count and First Contact with the highest hop count, this follows the theory as well. It can be observed from the plot that the average hop count in case of highly heterogeneous is less with respect to minimum heterogeneous environment, the probable reason can be the more opportunities of contact is possible in highly heterogeneous due to various in speed of all the nodes and nodes come in the range of communication in more heterogeneity.

5.4.4. Average Latency



(A) (B)

Figure 12. Average Latency of Routing Protocols (A) Highly Heterogeneous vs (B) Minimum Heterogeneous.

In both the scenario Epidemic and MaxProp performed the best and Spray and Wait is the worst, which is as expected.

6. CONCLUSION

We have measured how routing algorithms differ when the environment is extremely heterogeneous. We have compared six highly cited algorithms, namely Epidemic, MaxProp, Prophet, Spray and Wait, First Contact and Direct Delivery. Our observation is that average delivery probability of MaxProp is the highest and Direct Delivery is the lowest, so Direct delivery is only suitable for nodes that need very low power consumption. The First Contact routing delivery probability may not differ much with changes in the buffer sizes. In minimum heterogeneous environment Spray and Wait performed the best. Epidemic routing passed most of the messages with huge values of the parameter because of packet duplication, moreover, it consumes more resources than available to portable devices. So, when it comes to overhead ratio Epidemic is the worst performing one and Spray and Wait is the best performing. In the First Contact, the average hop count is highest but in Spray and Wait it is stable. Epidemic algorithm has the lowest average latency though this routing protocol is not practical for real deployment of opportunistic networks because they either had an extremely low delivery ratio or had extremely high resource consumption. We have also noticed, the delivery probability of Direct Delivery is very low in highly heterogeneous environment, so when Direct Delivery is not able to reach its destination, it is not possible to plot that algorithm properly. For most of the cases MaxProp algorithm is found to perform better than others. A lot more analysis can be taken up as future work. One such area can be analysis of opportunistic routing algorithms based on social relationship which are very common nowadays and probably the opportunistic algorithm for future.

REFERENCES

1. S. Biswas and R. Morris, “ExOR: opportunistic multi-hop routing for wireless networks,” In Proc. of ACM SIGCOMM, Aug. 2005.
2. S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” In Proc. of ACM SIGCOMM, Aug. 2007.
3. Shuai Wang, Basalamah, A. ,Song Min Kim, Shuo Guo ,Tobe, Y.; Tian He, "Link-Correlation-Aware Opportunistic Routing in Wireless Networks," in Wireless Communications, IEEE Transactions on , vol. 14, no. 1, pp. 47-56, Jan. 2015.
4. Peter Larsson. Selection diversity forwarding in a multihop packet radio network with fading channel and capture. SIGMOBILE Mob. Comput. Commun. Rev., 5(4):47–54, 2001.
5. D. Fuste-Vilella, J. Garcia-Vidal, and J.D. Morillo-Pozo. Cooperative forwarding in IEEE 802.11-based MANETs. In Wireless Days, 2008. WD ’08. 1st IFIP, pages 1–5, nov. 2008.
6. H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Valuable detours: Least-cost anypath routing. IEEE/ACM Transactions on Networking, 19(2):333–346, April 2011.
7. Che-Jung Hsu, Huey-Ing Liu, and Winston Khoon Guan Seah. Opportunistic routing - a review and the challenges ahead. Computer Networks, 55(15):3592–3603, 2011.
8. Clausen T., Jaqcquet P.Optimized Link State Routing (OLSR) RFC 36262003IETF Networking Group.
9. Perkins C., Belding-Royer E., Das S., et al.Ad hoc on-demand distance vector (AODV) routingRFC20033561Internet RFCs.
10. Cerf V., Burleigh S., Hooke A., Torgerson L., Durst R., Scott K., Fall K., Weiss H.RFC 4838, Delay-Tolerant Networking Architecture20072IRTF DTN Research Group.
11. Vahdat A., Becker D.Epidemic routing for partially connected ad hoc networks2000CS-200006Duke University.
12. Spyropoulos T., Psounis K., Raghavendra C. S.Efficient routing in intermittently connected mobile networks: the multiple-copy caseIEEE/ACM Transactions on Networking200816177-90.
13. Grasic S., Davies E., Lindgren A., Doria A.The evolution of a DTN routing protocol—PRoPHETv2Proceedings of the 6th ACM Workshop on Challenged NetworksSeptember 2011Las Vegas, Nev, USAACM27-30.
14. Burgess J., Gallagher B., Jensen D., Levine B. N.MaxProp: routing for vehicle-based disruption-tolerant networks6Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM ‘06)April 2006Barcelona, SpainIEEE1-11.
15. Spyropoulos T., Turletti T., Obraczka K.Routing in delay-tolerant networks comprising heterogeneous node populationsIEEE Transactions on Mobile Computing2009881032-1047.

16. Elwhishi A., Ho P.-H.SARP-a novel multi-copy routing protocol for intermittently connected mobile networksProceedings of the IEEE Global Telecommunications Conference (GLOBECOM ‘09)December 2009Honolulu, Hawaii, USAIEEE1-7.
17. Nelson S. C., Bakht M., Kravets R.Encounter-based routing in DTNsProceedings of the 28th Conference on Computer Communications (INFOCOM ‘09)April 2009Rio de Janeiro, Brazil846-854.
18. Pujol J. M., Toledo A. L., Rodriguez P.Fair routing in delay tolerant networksProceedings of the IEEE Conference on Computer Communications (INFOCOM ‘09)April 2009Rio de Janeiro, Brazil837-845.
19. A. Arockia Abins and N. Duraipandian, opportunistic multi-hop routing for wireless SOAR: simple opportunistic adaptive routing networks, in: Proceedings of SIGCOMM, ACM, protocol for wireless mesh networksAmerican-Eurasian Journal of Scientific Research 10 (3): 148-153, 2015.
20. "Improved PROPHET Routing Protocol in DTN" by Dushyant Patel, Rakesh Shah student, Dept. Of Computer Engineering, Grow More Faculty of Engineering Himatnagar, Gujarat, India in International Research Journal of Engineering and Technology (IRJET).
21. Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level measurements from an 802.11b mesh network. SIGCOMM Comput. Commun. Rev., 34:121–132, August 2004.
22. Barry M. Leiner, Robert J. Ruth, and Ambatipudi R. Sastry, “Analysing the Behaviour and Performance of Opportunistic Routing Protocols in Highly Mobile Wireless Ad Hoc Networks ,” IEEE Personal Communications, 3(6):34–43, December 1996.
23. Article Evaluating Forwarding Protocols in Opportunistic Networks: Trends, Advances, Challenges and Best Practices Vishnupriya Kuppusamy , Udaya Miriya Thanthrige , Asanga Udugama and Anna Förster Sustainable Communication Networks, University of Bremen, 28359 Bremen, Germany 11 May 2019.
24. Improved PROPHET Routing Protocol in DTN Dushyant Patel, Rakesh Shah Student, Dept. Of Computer Engineering, Grow More Faculty of Engineering Himatnagar, Gujarat, IndiaAsst. Professor, Dept. of Computer Engineering Journal of MordenEngineering Research, July-Aug 2012.
25. Evaluation of Opportunistic Routing Algorithms on Opportunistic Mobile Sensor Networks with Infrastructure Assistance Viet-Duc Le, Hans Scholten, and Paul Havinga Pervasive Systems (PS), Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS) University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.
26. Review Void Avoiding Opportunistic Routing Protocols for Underwater Wireless Sensor Networks: A Survey Rogaia Mhemed 1, William Phillips 1, Frank Comeau 2 and Nauman Aslam 3.

27. Review A Survey on the Evolution of Opportunistic Routing with Asynchronous Duty Cycled MAC in Wireless Sensor Networks Ayesha Akter Lata and Moonsoo Kang Department of Computer Engineering, Chosun University, 375 Seosuk-Dong, Dong-Gu, Gwangju 501-759, Korea; lata_sust@yahoo.com Correspondence: mskang@chosun.ac.kr; Published: 23 July 2020.
28. Performance Comparison of Routing Protocols in Opportunistic Networks, Chander Prabha, Rakesh Kumar ,Associate Professor, Department of Computer Science and Engineering, Professor, Department of Computer Science and Engineering 2020.
29. Impact Of Message Size On Epidemic Routing In Mobile Opportunistic Networks Jasvir Singh, Raman Maini International Journal of Scientific & Technology research vol.9 issue 02, February 2020.
30. V.D. Le, H. Scholten, and P. Havinga, “Unified routing for data dissemination in smart city networks,” in Proc. of the 3rd International Conference on the Internet of Things (IoT2012), 2012.
31. S. Biswas and R. Morris, “ExOR: opportunistic multi-hop routing for wireless networks,” In Proc. of ACM SIGCOMM, Aug. 2005.
32. S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” In Proc. of ACM SIGCOMM, Aug. 2007.
33. Shuai Wang , Basalamah, A. ,Song Min Kim, Shuo Guo ,Tobe, Y.; Tian He, "Link-Correlation-Aware Opportunistic Routing in Wireless Networks," in Wireless Communications, IEEE Transactions on , vol. 14, no. 1, pp. 47-56, Jan. 2015.
34. V. G. Menon and P. M. Joe Prathap, “Opportunistic routing with virtual coordinates to handle communication voids in mobile ad hoc networks,” in Advances in Signal Processing and Intelligent Recognition Systems, vol. 425 of Advances in Intelligent Systems and Computing, pp. 323–334, Springer International, 2016.
35. J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva, “a performance comparison of multi-hop wireless ad hoc network routing protocols,” Proc. ACM MobiCom, pp. 85-97, 1998.
36. I. T. Haque, “On the overheads of ad hoc routing schemes,” IEEE Systems Journal, Vol. 9, No. 2, pp. 605-614, June 2015.
37. David B. Johnson. Routing in Ad Hoc Networks of Mobile Hosts. Proceedings of the Workshop on Mobile Computing Systems and Applications, pp. 158–163, IEEE Computer Society, Santa Cruz, CA, December 1994.
38. D. Johnson and D. A Maltz, “Dynamic Source Routing in A Hoc wireless networks”, Mobile Computing, pp.153-181, Kluwer Academic Publishers, 1996.
39. D. Johnson, Y. Hu, and D. Maltz, “The Dynamic Source Routing Protocol (DSR) for mobile ad hoc networks for IPv4,” Internet Requests for Comments, RFC Editor, RFC 4728, Feb. 2007.