Delay Tolerant Networks: An Analysis of Routing Protocols with ONE Simulator

Er. Richa Thakur
Department of Computer Science, Himachal Pradesh University, Shimla, India
E-mail: richathakur.uiit@gmail.com

Prof. K.L. Bansal
Department of Computer Science, Himachal Pradesh University, Shimla, India
E-mail: kishorilalbansal@yahoo.com

Abstract—Delay/disruption Tolerant Networks (DTNs) provide connectivity in those networks which lack continuous connectivity or considerable delays like that of terrestrial mobile networks, military ad-hoc networks, sensor or planned network in space. They lack in an end-to-end path between Source and Destination resulting in long variable propagation delays. The Internet Protocols do not operate properly in these networks, thus raising a variety of new challenging problems in this area. The DTN effectively improves the network communications where the connectivity in the network is intermittent or is prone to disruptions. Routing in DTNs is challenging because of long and frequent time durations of non-connectivity. There are several routing approaches that have been proposed with strategies ranging from flooding to forwarding approaches. In this paper these protocols are analyzed based on the quantitative data gathered by simulating each protocol in ONE simulator environment. The performance is discussed and compared for different routing protocols and results are discussed for different performance metrics.

Index Terms—Delay Tolerant Networks (DTNs), ONE simulator tool, Routing in DTN.

I. INTRODUCTION

Today wired and wireless networks have enabled a wide range of devices to be interconnected all over the world. Mobile devices like smart phones are gaining increasing importance both in private and professional sector. Around 4.61 billion mobile phone users exist today, in 2016, which are estimated to become 5.07 billion in 2019 [1]. 3.4 billion Internet users [2], which make around 40% of the world population, shows that today existing networks are very successful networks.

But in a strong contrast to omnipresent Internet access in the developed world, there are still 4.0 billion non-Internet users in the world at present [2]. Further, developing countries have very low internet rates where only 22% internet rates are there in India [3]. According to a new United Nations report on 21st September 2015-broadband Internet is failing to reach billions of people living in the developing world, including 90% of those living in the poorest nations [4].

One of the reasons of the non-availability of Internet access to the people is our over-burdened existing technology. The traffic volume generated by today's mobile-connected devices shows exponential growth with a 26-fold increase between 2010 and 2015, reaching over 6 Exabyte per month in 2015 [5]. Moreover, the current network technology still cannot reach everywhere, and also for some applications their infrastructure cost is very high. The two main reasons for these limitations are: the infrastructure-based design of the existing technologies and the fundamental assumptions on which these technologies rely. The first and most important of these assumptions is an end-to-end connectivity from the source to the destination, possibly via multiple intermediate nodes which can be easily violated due to power savings (as in case of sensor networks), mobility or unreliable networks connections in which the nodes are exposed to long delays or may be disconnections. So, here comes the Delay-tolerant networking (DTN) which is an attempt to provide the infrastructure-less networking, beyond the reach of existing networks. . Due to their infrastructure-less nature, they can be deployed where infrastructure access is not available or possible, or to offload the congested infrastructure networks.

The main feature of DTNs is disruption or delay which is mainly because of limited wireless radio range of widely scattered mobile nodes, limited energy resources, interference and attacks etc. Hence routing in these networks become a challenging task and is an active research area. Unlike the conventional routing strategies, DTNs lack permanent network connectivity which makes its routing more challenging. The lack of instantaneous path connections result in high latency of data delivery, overall low data rates, long queuing delays and limited longevity of individual nodes.

This paper aims to analyze the performance dynamics of various existing protocols on the basis of essential resources and performance metrics of a network i.e. buffer size, message creation-rate, delivery probability, average latency and overhead ratio. The organization of the paper is as follow: Section I gives introduction about the DTN. Section II briefly describes the related work by

various authors in the same field. Section III is focused on routing approaches and at last Section IV is the detailed description about the simulation results and analysis of the work done.

II. RELATED WORKS AND CONTRIBUTIONS

The very first work of collecting and classifying the existing routing protocols of delay tolerant networks was done by E.P.C. Jones et al. [6] which classified the protocols on two key properties: Replication and Knowledge. The authors then identified the need of a hybrid technique that exploits both of the properties to work on real scenarios.

Among various routing protocols Epidemic protocol [7] is a popular one using complete flooding approach for message transmission in the delay tolerant network. Ram Ramanathan et al. [8] have extended the work to get Prioritized Epidemic which uses expiry time information and topology awareness to decide which bundles to delete or hold back when faced with a resource crunch. Many papers have studied ways to make Epidemic Routing consume fewer resources [9, 10]. Flooding families create a lot of redundancy by generating message replicas and there is a need to remove these replicas from DTN [11]. On the other hand are forwarding protocols MaxProp [12] and Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) which make use of mobility patterns for routing decisions. The DTN design makes a different set of choices in the architectural design of the protocols [13]: messages versus packets, a form of hop-by-hop reliability and security versus end-to-end, name based routing versus address based routing, and a routing abstraction of partially-connected rather than fully-connected network graph making the TCP/IP protocols useless in this scenario.

Jian Shen et al. [14] have also surveyed the existing/proposed routing protocols. The protocols are mainly classified for Flooding and Forwarding families and both families are compared and analyzed in terms of various performance metrics. A further classification is given by Salman Ali et al. [15]. The authors classified the protocols on the basis of replication, knowledge and coding based and gave a comprehensive comparison of the DTNs routing strategies. Further a simulation based performance comparison of protocols is done in [16]. New protocols keep on emerging in DTNs. Lately Ahmad El Shoghri et al. [17] have proposed Augur that uses spatiotemporal information of the networking nodes and route the message in the network using this information. The authors then compared the performance of the existing protocols with the Augur on ONE simulator tool. Similarly CASPaR [18] is a congestion avoidance shortest path protocol for DTNs.

Applications of DTNs are significantly recognized in various scenarios today [19]. The Disaster Response system in [20], HimSwan [21] - a healthcare system, Military [22] and Mobile Adhoc NETworks (MANETs) are some of the application areas which need DTNs.

Finally Ari Keranen et al. [23] have given the detailed description about the ONE simulator which is a Java-based simulator for the evaluation of the DTNs. It offers a wide variety of tools to create complex mobility scenarios that come closer to reality than many other mobility models.

III. ROUTING IN DTNS

The important issue of routing in DTNs is a challenging task. Delay Tolerant Networks have to deal with disconnections, waiting time might range from seconds to days, buffer space of intermediate nodes must meet the demand of the network and finally energy consideration of individual nodes is an important task [24]. Delay Tolerant Networks forward messages opportunistically and cooperatively on occurrence of contacts between physical devices when mobile devices come into mutual communication range. They employ a *store-carry-forward* routing strategy where messages are *stored* for longer duration, *carried* through mobile devices and *forwarded* if the destination device or a better suited device is encountered.

A. Store-Carry-Forward:

Store-carry-forward is a message passing approach that a node follows after receiving a message. The "Store" phase adds the message to the node's buffer that allows the data to wait for a suitable time to forward the message. In the "Carry" stage the message is propagated to other regions of the network physically by the movement of the node carrying the data. Finally, "Forward" is the stage when the node decides to send the message to another node due to the availability of other better candidates or to the message's final destination [25].
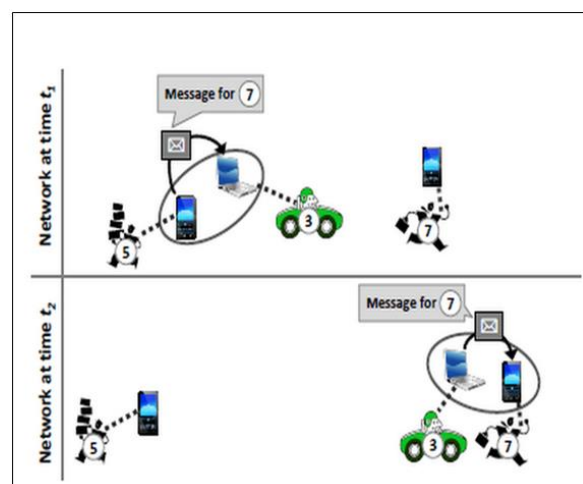

Fig.1. Message Delivery in a DTN

Fig.1 shows an exemplary DTN scenario where a message is carried through and then delivered to the destination device: At time t1 device 5 sends a message destined for device 7. It forwards the message to device 3 as it is more capable of carrying the message to device 7.

At the network time t2 device 3 and destination device 7 comes in the communication range of each other and then the message is delivered to destination device by device 3. Note that there was no communication path between device 5 and device 7. The path only exists in parts over time.

B. Flooding-based Routing strategies:

Strategies in the flooding family transmit multiple copies of each message to a set of nodes called relays. These relays store the messages until they got connected with the destination, at which point the message is delivered. Usually, these strategies are studied in the context of mobile ad-hoc networks, where random mobility has a good chance of bringing the source into contact with the destination. Message replication is then used to increase the probability that the message gets delivered. The basic protocols in this family do not need any information about the network; however more advanced schemes use some knowledge to improve performance [6].

Direct-Contact/Direct-Delivery (DD): This is a simple routing approach in which the source delivers the message only to the final recipient. Clearly it is a one hop routing scheme. Because of its simple characteristics it does not consume many resources.

First Contact (FC): In the first contact routing only a single copy of the message is used. The source node forwards the message to the first available contact, which in turn forwards it to the next first available contact and so on.

Epidemic Routing (Epi): Epidemic routing is the extreme end of the flooding family approach. It works as follows: when a message is sent to a destination, it is first saved in a local buffer and tagged with a unique ID. When two nodes come in the contact range of each other they exchange the list of all the message IDs they have in their buffer and exchange those messages they don't have in their local buffer. Hence in the end of this exchange both the nodes will have same messages. This process continues until all the nodes have all messages in their buffer. In epidemic routing, it is ensured that messages can be delivered with a high probability. However, the network resources are consumed heavily [26].

Spray and Wait: Spray and Wait is a controlled flooding scheme. It consists of two phases: Spray phase and Wait phase. Given an initial number of allowed replicated copies per message N, the spray phase sprays the message to the first encountered devices, keeping N/2 replicas and forwarding rest of it. The process continues recursively until N=1. If the destination is not found in the spraying phase, each of the N nodes carrying a message copy performs direct transmission to the destination node [27].

C). History/ Prediction-based Routing Strategies:

The routing strategies in this family use network topology information to select the best possible path in the network and the message is then forwarded from one node to another along this path. The strategies in this family require some knowledge about the network. They typically send a single message along the best path; hence they do not use replication.

Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET): PRoPHET uses the statistics of previous encounters made by a node with other neighbors. In this routing strategy each node locally gathers meeting probabilities with other nodes when in mutual communication range. Replication of a message between two nodes in transmission range is performed if the node currently not storing the message has a higher meeting probability for the message's destination than the node currently storing the message. In PRoPHET to forward data from one node to another it uses a probabilistic metric called *delivery predictability*, $P_{(a, b)}$ E [0, 1], at every node *a* for each known destination *b*. This metric indicates the likelihood of a node to deliver a message to that destination [28]. Delivery predictability is stored in internal delivery vector and it gets updated whenever nodes meet each other. This delivery predictability metric is used by each node and is recalculated at each opportunistic encounter according to these rules:

(1) When a node *a* encounters another node *b*, the predictability for *b* is increased. This is shown in Equation (1).

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) \times P_{init} \qquad (1)$$

where P_{init} is an initialization constant.

(2) The delivery predictability of nodes must age because if two nodes do not encounter/meet each other in a while, then they are less likely to forward messages to each other. Equation (2) shows this ageing equation.

$$P(a,b) = P(a,b)old \times \gamma k \qquad (2)$$

where γk is an aging constant.

(3) The delivery predictability follows the transitive property i.e. if a node *a* frequently meets node *b* and node *b* frequently encounters node *d*, then node *d* probably is a good node to forward message intended for node *a*. The effect of transitivity on delivery predictability is shown in Equation (3).

$$P_{(a,d)} = P_{(a,d)old} + (1 - P_{(a,d)old}) \times P_{(a, b)} \times P_{(b,d)} \times \beta \quad (3)$$

where β is the scaling constant that decides how much large impact the transitivity should have on the delivery predictability.

IV. Performance Evaluation and Simulation

To analyze the performance of various routing protocols we have used the Opportunistic Network

Environment (ONE) simulator tool to create a real time scenario. We analyzed our simulation results mainly on the basis of three performance metrics viz. Delivery Probability, Average Latency and Overhead Ratio. Also an important resource in DTNs which is the Buffer Size is explored for the considered protocols. The main aim of the evaluation is to analyze the various aspects of the DTNs in order to gain the better insight and understanding of the limitations of the existing protocols in terms of various performance metrics.

A. Simulation Tool:

Simulation plays an important role in analyzing the behavior of DTN routing and application protocols. We used the ONE to extensively evaluate the different performance dynamics of the various routing strategies. ONE is a powerful simulator used to implement realistic DTN scenarios. It is a Java-based tool offering a broad set of DTN protocol simulation capabilities in a single framework [23]. The ONE simulator can be run on Linux, Windows or any other platform supporting Java. It is an agent-based discrete event simulation engine that is designed for evaluating the performance of DTN routing protocols. At each simulation stage, the engine updates a number of modules that implement the main simulation functions. Unlike other DTN simulators which focus only on routing simulation, the ONE combines mobility models, inter-node contacts, DTN routing protocols, message handling and visualization in one package and provides a rich set of reporting and analyzing modules. A detailed description of the simulator is available in [23] and the ONE simulator project page [29], where the source code of the simulator is also available. Source codes are written in Java programming language.
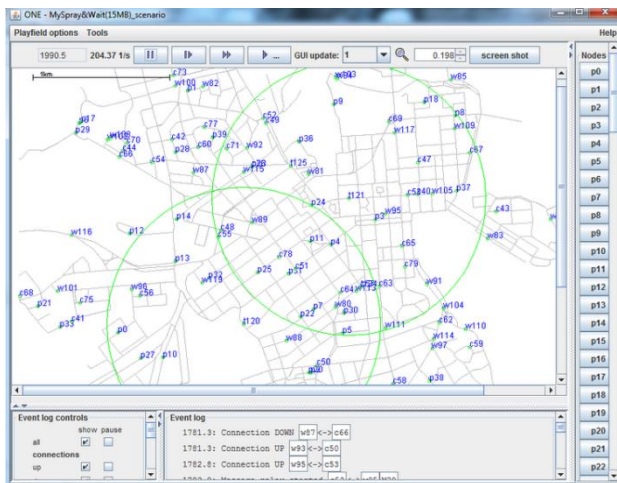

Fig.2. ONE Simulator at Work

Fig. 2 shows a snapshot of the ONE simulator at work. The graphical user interface appears once the simulator starts running. The main window in the snapshot shows the map of Helsinki city which is the by-default map provided in the simulator. Bottom left window shows the event log controls which can be controlled by various checkboxes provided in there. Beside this window is the event log window showing various connections between nodes. On the right hand side is the list of nodes taking part in the simulation with each node having a unique id (e.g. p0, p1, p2…). Messages created in a simulation run also have unique ids. Two green circles in the snapshot are the radio communication range of the two nodes. All settings related to the simulation and for the nodes are done in *default_settings* file provided with the simulator.

In order to analyze the performance metrics of different DTN routing protocols we visualized the simulation in real-time and analyzed the various reports generated by ONE simulator using the settings shown in Table 1.

Table 1. Settings used for ONE Simulator

Parameters	Values
Simulation time	7200s = 2 hours
Bus Movement Model	Map Based Movement
Number of Nodes	126
Message size	500KB – 1MB
Message ttl (time-to-live)	120min
Interface transmission speed	250kbps
Interface transmission range	100m

B. Performance Metrics used:

The DTN protocols are evaluated and analyzed using the following performance metrics:

Delivery Probability: It is the ratio of the number of delivered messages to the total number of messages created by the source node.

$$Delivery\ Probability = \frac{Number\ of\ messages\ delivered\ successfully}{Total\ number\ of\ messages} \tag{4}$$

Average Latency: It is the average time taken by the messages from their creation to their first delivery at the destination node.

$$Average\ Latency = \left(\frac{Delivery\ time - Creation\ time}{Total\ delivered\ messages}\right)_{Average} \tag{5}$$

Overhead Ratio: Overhead ratio is another important metric which shows how efficient a protocol is in terms of correct relay decisions.

$$Overhead\ Ratio = \frac{(Number\ of\ transmitted\ messages\ which\ are\ not\ delivered)}{(Total\ number\ of\ transmitted\ messages)} \tag{6}$$

C. Simulation Results:

Impact of Buffer Size:

Buffer storage is a valuable resource the participating

nodes have in a network. We start our study by exploring this resource for various participating protocols and then choosing an optimum buffer size on which all protocols can run optimally. We run our experiment for following buffer sizes: 2MB, 4MB, 8MB, 12MB and 15MB. Message creation rate is set to 1 every 30 seconds. Our aim is to see the effect of varying the buffer size on delivery probability of different routing protocols.

Fig. 3 shows the simulation results for various protocols on different buffer sizes. The performance metric delivery probability is analyzed in this simulation. The simulation results show that in Direct Delivery (DD) protocol buffer sizes greater than 2MB do not results in high delivery probability. The reason for this behavior is that increasing the buffer size will surely increase the number of messages stored in the local buffer but it does not guarantee the availability of destination node for receiving the messages. Same results are obtained for First Contact (FC) routing protocol. However Epidemic (Epi), Spray and Wait (S&W) and PRoPHET deliver more messages with increase in buffer size. This is because Epidemic and Spray and Wait protocols use flooding approach for message transmission and more buffer size means less number of messages dropped. Note that there is no increase in delivery probability of all protocols except Epidemic when increasing the buffer size from 12 MB to 15 MB. Epidemic routing performs well compared to other routing protocols but at the cost of extensive resource usage.

at a scale from 0 to 1. An interesting thing to observe is that all these protocols are far from a sure delivery probability (i.e. a probability of 1 in current scenario) which shows all DTN protocols are still in their infancy and a lot improvement in routing approaches is needed to enhance the probability of delivering messages.

A further exploration of buffer size is shown in Fig. 5. The figure shows the number of messages dropped using different buffer sizes (message statistics are given in Table 3). On using a small size buffer of 2MB the number of messages dropped are high as there is less space to store the created messages. Epidemic routing has the highest message dropping rate which is because of extensive flooding used by it. The approach here for dropping message is drop-oldest-message. When increasing the buffer size, the number of messages dropped is decreased and this number goes down to 0 for 15MB buffer size except Epidemic routing. Using the result of this simulation we chose our optimal buffer size for all protocols as 12MB to further analyze the different performance metrics of a DTN protocol.

Fig.5. Number of Messages Dropped for Different Buffer Sizes

Table 2 shows the message statistics for different protocols at different buffer sizes which are used for analyzing different protocols.
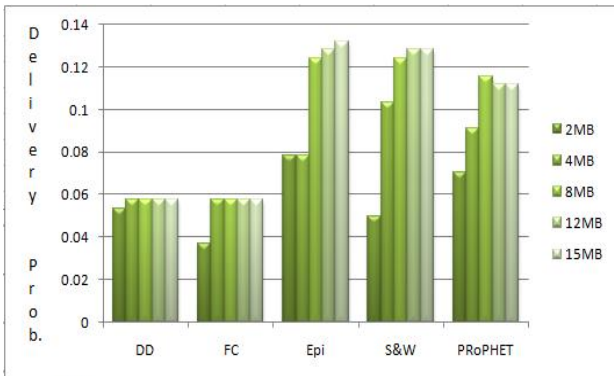

Fig.3. Delivery Probability vs. Buffer Size
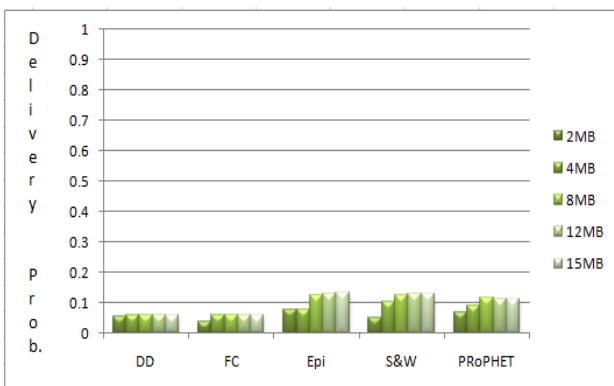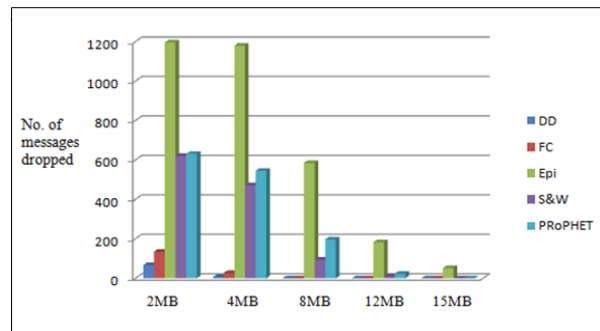

Fig.4. Delivery Probability in a Range from 0 to 1

Fig. 4 shows the delivery probability of the protocols

Table 2. Message Statistics for different Buffer Sizes

Message creation Rate = 1 message/30seconds						
Messages	**Prot-ocols**	**2MB**	**4MB**	**8MB**	**12MB**	**15MB**
Created	All	242	242	242	242	242
Started	DD	29	33	33	33	33
	FC	1739	2111	2143	2143	2143
	Epi	2937	3491	3572	3542	3523
	S&W	1797	2224	2272	2277	2277
	PRoP HET	1806	2141	2177	2135	2130
Dropped	DD	67	6	0	0	0
	FC	135	27	0	0	0
	Epi	1196	1180	585	183	52
	S&W	621	472	95	10	0
	PRoP HET	631	545	197	23	0
Delivered	DD	13	14	14	14	14
	FC	9	14	14	14	14
	Epi	19	19	30	31	32
	S&W	12	25	30	31	32
	PRoP HET	17	22	28	27	27

Overhead Ratio

Overhead ratio is another important metric which shows how efficient a protocol is in terms of correct relay decisions. We run our simulation for the chosen buffer size of 12MB and analyzed the overhead created by each protocol.
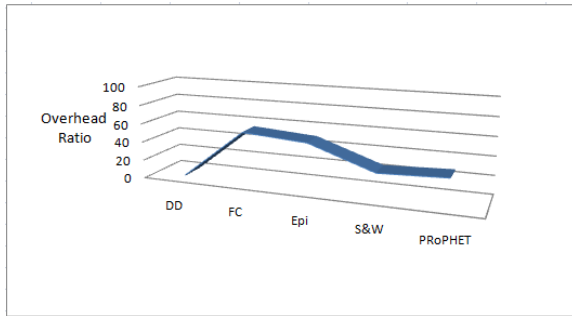

Fig.6. Overhead created by considered protocols at 12MB Buffer size

Fig. 6 shows the overhead created by considered protocols at 12MB buffer size. Obviously Direct Delivery routing has zero overhead as no messages are transmitted by it to the intermediate nodes. From rest of the considered protocols PRoPHET which is a forwarding family protocol performs the best on optimal buffer size. This is mainly because of correct relay decision taken by it at the time of transmitting a message to the intermediate node using the history data of the nodes in the network. Epidemic routing has a high overhead ratio while Spray and Wait protocol has overhead almost equal to PROPHET. This shows that a controlled flooding as done by S&W significantly reduces the overhead in a high traffic scenario.

Average Latency:

Finally the Average Latency at different loads is analyzed. Simulation is carried out on buffer size of 12MB and message creation rate is varied. Message creation rates are varied from 1 message created per 30 seconds to 1 message created per 240 seconds.
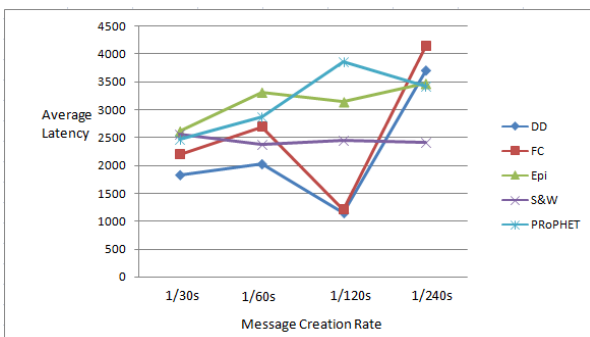

Fig.7. Average Latency vs. Message Creation Rate

Fig. 7 shows the average latency of the protocols with detailed statistics given in Table 3. Epidemic routing has the highest average latency rate while Direct Delivery protocol has the lowest on high traffic (1 message per 30s). This shows that at higher traffic load one must stop taking any routing decisions and use the simple direct delivery option since it would not load nodes further. While on low data rates (1 message per 240s) Epidemic routing performs well.

Table 3. Average Latency in Seconds for Different Message Creation Rate

Message Rate (in sec)	DD	FC	Epi	S&W	PRoP HET
1/30	1827.2	2197.2	2610.8	2555.9	2464.5
1/60	2022.0	2697.6	3314.3	2381.7	2869.9
1/120	1145.6	1211.5	3140.7	2454.3	3854.7
1/240	3700	4136.6	3473.6	2417.2	3411.8

Simulation Results:

On the basis of the detailed analysis and quantitative data gathered by us the performance of the different protocols can be summarized in the Table 4. On a scale from 1 to 5 where 1 is *least*, 2 is *low*, 3 is *moderate*, 4 is *high* and 5 is *highest* the different protocols are given values.

Table 4. Simulation Results

Buffer Size = 12MB					
Metric		DD	FC	*Epi*	S&W	*PRoPHET*
Overhead Ratio		0	High	High-est	Low	Least
Average Latency	At high traffic	Least	Low	High-est	High	Moderate
	At low traffic	High	Highest	Low	Least	Moderate
Delivery Probability		Low	Low	High	Highest	Moderate

The overhead ratio of PRoPHET is the lowest while of Epidemic protocol is the highest. Average Latency at high traffic (i.e. 1 message/30s) for Epidemic is the highest while for Direct Delivery protocol it is the lowest. At low traffic (1 message/240s) Spray and Wait performs considerably good while First Contact routing performs the worst. Delivery probability of Spray and Wait is the highest.

With these results it can clearly be seen that no one protocol is best in every metric. There is always a trade-off between one or more performance metrics.

V. CONCLUSION AND FUTURE WORK

Routing in Disruption Tolerant Networks is a challenging task. Different routing protocols have tried to minimize the delays, which are unavoidable, in DTNs but there is still a trade-off between high resource consumption and high delivery ratio of a protocol. In this paper we have analyzed the five DTN protocols on quantitative data gathered from simulating the protocols environment on ONE simulator. The results show that simple flooding protocol like Epidemic has high delivery probability but the message latency is high. On the other

hand forwarding protocols performs good but at the cost of resource consumption. Also all these protocols are far away from the guaranteed delivery probability of 1.

Future work in the DTN could be to consider the trade-off between the different performance metrics of various protocols and exploit a hybrid technique which makes use of flooding as well as forwarding to gain the best performance for a specific application. For this purpose the deep analysis of the different aspects of these protocols from different angles are required.

REFERENCES

[1] Mobile phone users worldwide 2013-2019, http://www.statista.com, (retrieved July 20, 2016).

[2] Internet users in the world, http://www.internetlivestats.com, (retrieved July 20, 2016).

[3] World Internet Users-Internet World Stats, http://www.internetworldstats.com, (retrieved July 21, 2016).

[4] Billions of people in developing world still without Internet access, UN News Centre, http://www.un.org, (retrieved July 21, 2016).

[5] Cisco Systems Inc. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2010-2015. http://newsroom.cisco.com/dlls/ekits/Cisco_VNI_Global_ Mobile_Data_Traffic_Forecast_ 2010_2015.pdf February 2011.

[6] E. P. C. Jones and P. A. S. Ward, *"Routing Strategies for Delay-Tolerant Networks"*, Submitted to Computer Communication, 2008.

[7] A. Vahdat and D. Becker, *"Epidemic Routing for Partially Connected Ad Hoc Networks"* Tech. rep. CS-200006, Duke University, Apr 2000.

[8] Ram Ramanathan et al*., "Prioritized Epidemic Routing for Opportunistic Networks"*, Proceedings of the 1st international MobiSys workshop, p 62-66, NY USA 2007.

[9] Teerapong Choksatid, Worrawat Narongkhachavana and Sumet Prabhavat, *"An Efficient Spreading Epidemic Routing for Delay-Tolerant Network"*, 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2016.

[10] Bhed Bahadur Bista and Danda B. Rawat, "*A Robust Energy- Efficient Epidemic Routing Protocol for Delay Tolerant Networks"*, IEEE International Conference on Data Science and Data Intensive Systems, 2015.

[11] H. Bindra and A. L. Sangal, *"Need of Removing Delivered Message Replica from Delay Tolerant Network- A Problem Definition"*, MECS publisher, International Journal of Computer Network and Information Security, IJCNIS Vol. 4, No.12, November 2012.

[12] John Burgess et al., *"MaxProp: Routing for Vehicle-based Disruption Tolerant Networks"*, Proceedings of IEEE Infocom, 2006.

[13] Kevin Fall, *"A delay-tolerant network architecture for challenged internets,"* 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp.27-34, 2003.

[14] Jian Shen, Sangman Moh, and Ilyong Chung, *"Routing Protocols in Delay Tolerant Networks: A Comparative Survey"*, The 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2008).

[15] Salman Ali, Junaid Qadir, Adeel Baig, *"Routing Protocols in Delay Tolerant Networks – A Survey"*, 6th International Conference on Emerging Technologies (ICET), 2010.

[16] Tamber Ambdelkader, Kshirasagar Naik, Amiya Nayak, Nishith Goel and Vineet Srivastava, *"A Performance Comparison of Delay-Tolerant Network Routing Protocols"*, IEEE Network, March/April 2016.

[17] Ahmad El Shoghri, Branislav Kusy, Raja Jurdak, Neil Bergmann, *"Augur: A Delay Aware Forwarding Protocol for Delay-Tolerant Networks"*, IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2015.

[18] Michael F. Stewart, Rajgopal Kannan, Amit Dvir and Bhaskar Krishnamachari, *"CASPaR: Congestion Avoidance Shortest Path Routing for Delay Tolerant Networks"*, International Conference on Computing, Networking and Communications, Wireless Ad Hoc and Sensor Networks, 2016.

[19] Wei Sun et al., *"On Delay-Tolerant Networking and its Applications"*, International Conference on Computer Science and Information Technology (ICCSIT 2011).

[20] H. Chenji et al., *"A Wireless Sensor, Ad-hoc and delay tolerant network system for disaster response"*, Technical Report LENSS-09-02, Sept 2011.

[21] Vikram Jeet Singh et al., *"Research on Application of Perceived QoS Guarantee through Infrastructure Specific Traffic Parameter Optimization"*, I.J. Computer Network and Information Security, Feb 2014.

[22] Ziyi Lu and Jianhua Fan, *"Delay/Disruption Tolerant Network and its Application in Military Communications"*, International Conference on Computer Design And Appliations ICCDA 2010.

[23] Ari Kerane et al., *"The ONE Simulator for DTN protocol evaluation"*, 2nd International Conference on Simulation Tools and Techniques, 2009.

[24] E. P. C. Jones, L. Li, P. A. S. Ward, *"Practical Routing in Delay-tolerant Networks"*, Proc. of ACM SIGCOMM workshop on Delay-tolerant networking, pp. 237-243, Sep. 2006.

[25] S. Yamamura, A. Nagata, and M. Tsuru, *"Store-carry-forward based networking infrastructure: vision and potential,"* 2011 Third International Conference on Intelligent Networking and Collaborative Systems (INCoS), pp.594-599, 2011.

[26] Jian Shen, Sangman Moh , and Ilyong Chung, "*Routing Protocols in Delay Tolerant Networks: A Comparative Survey*"**,** The 23rd International Technical Conference on Circuits/Systems, Computers and Communications ,ITC-CSCC 2008.

[27] Thrasyvoulos Spyropoulos, Konstantinos Psounis, Cauligi S. Raghavendra*," Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks"*, Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN'05), August 2005.

[28] Anders Lindgren et al. "Probabilistic Routing in Intermittently Connected Networks", ACM SIGMOBILE Mobile Computing and Communications Review, Volume 7 Issue 3, July 2003.

[29] Project page of the ONE simulator, https://www.netlab.tkk.fi/tutkimus/dtn/theone.

Authors' Profiles

Er. Richa Thakur has completed her Bachelor of Technology (B.Tech.) in Information Technology from University Institute of Information Technology, Himachal Pradesh University, India, in 2013 and is currently pursuing her M.Tech in Computer Science from Department of Computer Science, Himachal Pradesh University, Shimla, India.

Dr. Kishori Lal Bansal is a Professor in the Department of Computer Science, Himachal Pradesh University with a teaching experience of more than 20 years. He has completed his Ph.D. in Computer Science and Applications in 2005. He is an active research scholar with more than 50 research papers published in reputed journals.