Agent Aided Routing for Faster and Efficient Spreading of Messages in Delay Tolerant Networks

Shahid Md. Asif Iqbal[1],
Alok Kumar Chowdhury [2], and **Amina Akhter**[3], Non-members

ABSTRACT

Routing schemes in Delay Tolerant Networks depend on the intermediate nodes to carry messages till the forwarding of the messages to other nodes or delivery of the messages to their destinations. Thus, efficient and faster spreading of messages is essential to reduce the delay of sending a message from source to destination. In this work, we aim to design a new agent aided routing scheme in Delay Tolerant Networks to avoid large message delivery delay. Here, the source of a message is permitted to create unlimited copies of a message and distribute these copies to agents. Agents acting as the representative of the sources receive messages from the sources based on their node visited list and delivery predictability. After that, the agents forward the copies to the other intermediate nodes or deliver the message directly to the destination. Thus, the agents help in faster and efficient dissemination of a message in the network which helps in reducing message delivery delay. Besides, we compare our scheme with that of other schemes in different network scenarios through simulations. Simulation results show that our agent based routing scheme performs better than that of others in terms of Delivery Ratio, Delay and Communication Overhead.

Keywords: Delay Tolerant Networks, Agent, Visited List, Delivery Predictability, Delay, Intermediate Node

1. INTRODUCTION

Delay Tolerant Networks (DTN) are mobile wireless networks where most of the time there does not exist a complete path from a source to a destination, or such a path is highly unstable and may change or break soon after it has been discovered, even while being discovered. When the network is fairly sparse this nature causes the network to be viewed as set of disconnected, time-varying clusters of nodes [1]. It is an important class of emerging networks that exhibits some distinctive characteristics such as intermittent connectivity, large delay and high loss rate. DTNs arise in a variety of environments such as disaster relief, military communication, rural internet access, environmental sensing and surveillance, deep space communication, underwater sensing and inter-vehicular communication [2].

The unique characteristics of DTN such as high node mobility, low node density, frequent disconnection, limited resources (e.g., battery, computational power, bandwidth), environmental interference and obstruction, short range radio and malicious attacks [3] cause the network to suffer highly unstable paths. So the conventional routing algorithms for Internet and MANET fail in DTN.

To overcome network disconnections which occur frequently in DTNs, a new routing paradigm called store-carry-and-forward has been developed. In this approach, nodes buffer data in storage in case of network disconnection and forward data to other nodes when they are re-connected. DTN is based on message switching instead of packet switching. DTN routing schemes can be divided into two groups: Forwarding Based and Replication Based routing. In forwarding based routing schemes, it is assumed that each node in the network has the knowledge of (past as well as future) node trajectories, node meeting times and durations. So, only a single copy of a message is routed over fixed paths deterministically. However, performance of these routing schemes falls radically when the environment is completely opportunistic. Replication based routing schemes infuse multiple copies of each message into the network to ensure high delivery probability. Though, the forwarding based schemes are resource friendly and eliminate the need for an acknowledgement sent to the sender, these schemes do not allow for sufficient message delivery rates in many occasions. Since, nowadays nodes are marketed with more resources and it is wise to utilize these resources, by employing replication based routing schemes [5] - [13], rather left them unused. Consequently, it is clearly seen that there is a trade-off between the message delivery ratio and

Manuscript received on July 14, 2013 ; revised on November 10, 2013.
Final manuscript received April 19, 2014.
[1,2] The authors are with Department of Computer Science & Engineering, Premier University Chittagong, Bangladesh, E-mail: asifcsep@yahoo.com and alokchy04@yahoo.com
[3] The author is with Department of Computer Science, Asian University for Women Chittagong, Bangladesh, E-mail: runeycse99@yahoo.com

the energy consumption in the network. DTN routing schemes having the complete future knowledge but with limited resources, or with no or little future knowledge and unlimited resources both make the routing schemes far from optimal. Therefore, the trade-off should not be optimally derived because most of the DTN routing protocols are non-optimal and NP-hard [9].

Hence, while designing a routing scheme for DTNs, the important considerations must be (i) the total number of copies that are distributed to the network for each message, and (ii) the selection of nodes to which the message is replicated or forwarded. Replication increases the message delivery ratio but at the same time increases the delivery costs, while forwarding maintains at most one copy of the message in the network but achieves low delivery radio due to the node mobility and uncertainty [4].

In this paper, we have proposed a new Agent Aided Routing scheme. The idea of our routing scheme is as follows: first we allow a source to generate unlimited copies of a message. Then we allow the source to forward one or more copies of the message to the agents based on the visited list and delivery predictability of the agent. An agent is a node that meets the source directly. We then propose each agent to spread the message by forwarding only a single copy of the message to each node seen. The nodes carry the message until the destination is met and once found it delivers the message directly to the destination. In this way our agents help in efficient and faster spreading of messages they receive from the sources. Simulation results show that the proposed scheme, the Agent Aided Routing, maintains good delivery and latency performance while maintaining low overhead in various scenarios.

The rest of the paper is as follows. In Chapter II we describe some current works on this topic. Chapter III depicts our Agent Aided Routing. Simulation setup and analysis of results come in Chapter IV and finally in Chapter V we concluded our paper with some future research guidelines.

2. RELATED WORKS

Despite of the challenges of routing in DTN many routing schemes have been reported in the literature. Some of the routing schemes inject multiple copies of a message into the network and achieve higher delivery ratio and lower delay, and they are known as replication based routing schemes. But due to redundant copies of messages, these schemes consume network resources like buffer space, bandwidth and energy [5]. Replication based schemes like Epidemic [6], PROPHET [7], MaxProp [8], RAPID [9], PREP [10] etc. are considered into the category of flooding based schemes, because these schemes replicate as many copies of a message as the network permits which is vulnerable to high network contention and could lead to huge overhead and latency. Spray and Wait [1], ORWAR [11], Spray and Focus [12], and EBR [13] are Quota based schemes which relay limited number of copies of a message in the network to save the network resources.

Spray and Wait (SW) [1] makes use of two phases; Spray phase and Wait phase. Initially in this scheme, L number of copies of a message is generated by the source node. The source node forwards the L copies of a message to L nodes which do not have a copy of the message during the spray phase. If the destination is not among these L nodes, then the scheme switches to the Wait phase. In the Wait phase the L nodes hold their copy of the message until they reach to the destination. By improving the spray phase of SW, Binary Spray and Wait (BSW) [1] routing scheme has been evolved. During the spray phase of BSW routing scheme, a node forwards half the copies of a message, till it has more than one copy of that particular message, to a node that does not have a copy of the message. The node switches to the Wait phase when it is left with only a single copy of the message. Node performs direct transmission when it meets the destination.

As SW and BSW both approaches forward message copies to nodes without considering their performance, these quota based routing schemes suffer from blind flooding problem. In addition, due to message forwarding nature of BSW, it suffers from high communication overhead. In SW routing, only the sources take the responsibility to spread a message to a node met during the spray phase. Therefore, the communication overhead is less in SW. But this strategy also reduces the delivery ratio and also increases the delay as the intermediate nodes between the source and the destination except the neighbours (nodes that meet the source directly) of the source do not have a copy of message. On the other hand, BSW routing allows source nodes to forward half the copies of a message to a node which (the node) in turn follows the same criteria to forward the message to newly seen nodes. So in this technique more nodes are involved to forward a message during spray phase. As a result it achieves higher delivery ratio, lower delay but also experiences high communication overhead which becomes severe with higher number of copies of a message.

In PROPHET [7], each node forwards a message to its destination node using delivery predictability. When two nodes meet, the node with the more delivery predictability of the desired destination wins the message destined for that destination. However, PROPHET has higher average delay when nodes have low buffer size. Also it experiences high overhead because the average number of forwarded messages is high here. Another replication based routing technique, Epidemic [6] routing, is flooding based in nature. In this routing scheme, nodes continuously replicate and transmit messages to newly discovered contacts that do not already possess a copy of the message. This routing is particularly resource hungry because it deliberately makes no attempt to eliminate replications. In Epidemic routing, numerous copies of a single message in the network results in higher average delay and lower delivery ratio when the nodes have low buffer size.

Our scheme introduces a new type of nodes call agents. The agents are nothing but nodes which directly meet with the sources. Agents actually are able to represent the sources by receiving appropriate number of copies of a message and then disseminating those copies to other nodes. The sources generate infinite number of copies of a message but deliver only a appropriate numbers of copies to each node seen i.e. the agent, based on some performance parameters such as visited list and delivery predictability. An Agent forwards a copy of the message to other nodes, which do not have a copy of the message, until it is left with a single copy. The nodes with the single copy of a message hold their copy until they see the destination. In our proposed scheme optimum number of nodes, sources and the agents, are involved in the distribution of messages and hence our strategy shows higher delivery, lower delay and lower communication overhead compared to that of most of the others.

3. PROPOSED SCHEME

We propose to employ a new routing scheme in the DTN to minimize the message delivery delay while maintaining the lower communication overhead and desired delivery rate. Instead of spreading unlimited copies of a message and source spraying only, we introduce the concept of agent in the routing scheme. Next, we design a new agent aided routing scheme based on the nodes visited list and delivery predictability.

Our proposed scheme has got two important parts. First, we define the agents of a source node and the number of copies each agent should receive from the source node. The source node utilizes the visited list and delivery predictability of an agent to decide the number of copies of a message the agent should receive from it. The agents act as the representative of the sources. Later, we allow an agent to replicate a copy of a message to each node that does not posses it till the agent has more than one copy remaining.

3.1 The source and the Agent

A source is a node that wants to convey some messages to their particular destinations. To spread a message in the network quickly and to create more agents we do not impose any limit on the number of copies a source can generate. A source can send messages to multiple agents and it can work as an agent for other sources as well.

An agent is an intermediate node which meets the source directly i.e. the first hop of a source. An agent receives one or more copies of a message from the source. Based on the number of nodes visited by the agent and the delivery predictability of the agent for the destination of that message, the source gives one or more copies to the agent. An agent with more than one copy of a message can replicate a copy to a node lacking the message. A node working as the agent for other sources acts as the source for messages generated by the node itself.

3.2 Visited List of a Node

We allow each node to keep a list of nodes it has visited throughout the lifetime of the network. We use the set $\mathbf{V}_i$ to represent the visited list of node $\mathbf{i}$. If we have n number of nodes in the network then at certain instance of time $\mathbf{V}_i$ can be represented using

$$V_i \subseteq N$$

Where $\mathbf{N}$ is the set of $(\mathbf{n}-1)$ nodes in the network.

3.3 Delivery Predictability of an Agent

Like PROPHET [7], we calculate the delivery predictability, $\mathrm{P}_{(\mathrm{a},\ \mathrm{b})} \in [0, 1]$, in each node A for each known destination B. The delivery predictability indicates the likelihood of node A delivering a message to a particular destination B. When two nodes encounter each other, we permit the nodes to exchange the P-values of their known destinations and use the information to update the delivery predictability information at both ends. Thus, when a source meets an agent we use the P-values of the agent to determine the number of copies of the message it should receive from the source node.

3.4 Agent Copy Determination

We permit a source node to generate unlimited copies of a message and disseminate these copies to the agents based on their visited lists and delivery predictabilities. When a source node S, carrying a message M with unlimited copies for a destination D, meets an agent A, we allow the source S to use (1) to calculate the number of copies to be forwarded or replicated to agent A

$$cs_{s,a} \;=\; vl_a \times P_{(a,d)} \times c \tag{1}$$

where

$cs_{(s,a)}$ is the number of copies to be forwarded to agent A by source S,

vl_a is number of distinct nodes in the visited list of agent A

$P_{(a,d)}$ is the delivery predictability of agent A for destination D,

c is the adjustment parameter used to control the overhead in the network.

If $cs_{(s,a)} = 0$, for an agent then the source forwards a single copy of a message to the agent for the fastest spreading.

An agent showing good values for vl and/or P, is the better candidate to receive more copies from the source. So, we multiply these equally weighted terms to determine the appropriate number of copies to be forwarded to an agent. The adjustment parameter c is used to control the copy number when the values of vl or P are too high or too low which may lead to higher communication overhead or lower delivery ratio respectively.

An agent visiting more nodes throughout the network lifetime and having good delivery predictability for the destination of a message is a better customer or steward to receive and carry more copies of the message.

3.5 Spreading of Messages by an Agent

Whenever an agent carrying more than one copy of a message meets a new node lacking the message, we permit the agent to forward a single copy of the message to the node. When the copy number of a message in an agent reaches one we compel the agent to stop forwarding or spreading the message. Having a single copy of a message an agent waits for the destination, if found, delivers the message to its destination. Thus, an agent, acting as the representative of a source, helps in faster and efficient spreading of a message which may reduce the message delivery delay.

An intermediate node carrying a single copy of a message can only deliver a message to its final destination.

In this work, we try to make a trade-off between the two members of replication based routing family i.e. the flooding based and quota based schemes. Using the technique like PROPHET [7] we permit source nodes only to create unlimited message copies (to increase delivery ratio) and distribute them to agents based on their performance. While, limiting the copies forwarded by an agent to an intermediate node to one (1) like Spray and Wait [1], we have reduced the communication overhead too.

3.6 The AAR Framework

A node in AAR framework, shown in Fig. 1, can play one or more of the four different types of roles for a message i.e. as the source, as an agent, as an intermediate node, and as the receiver. A message, generated in the source, continues its propagation until it reaches at the Receiver. Like all the previous schemes, in AAR a message having expired TTL will be dropped.

When a node (A) containing the message meets another node (B), one of the following three cases may occur.

Case 1: B itself is the Receiver

Case 2: B is a node who already has a copy of the message

Case 3: B is a node who does not have a copy of the message

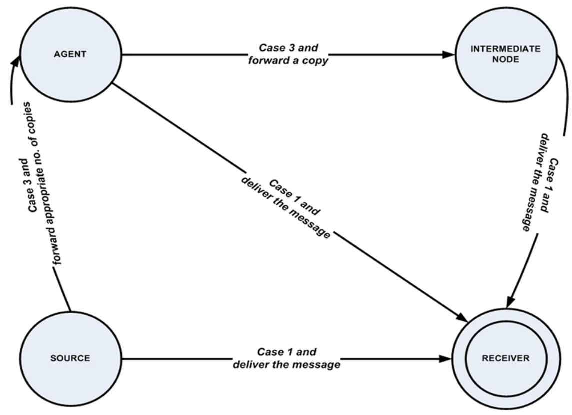

Fig.1: The AAR Framework.

At any stage, a node meeting the destination or receiver directly, Case 1, delivers the message but skips forwarding copies of messages for Case 2. But, for case 3, source forwards appropriate number of copies, determined by eq. (1), to the agents and an agent forwards a single copy to an intermediate node seen.

4. SIMULATION SCENARIO & RESULT ANALYSIS

ONE (Opportunistic Network Environment) [14] developed by Helsinki University of Technology is suitable to perform simulation for DTN. We have implemented and validated our new routing scheme in ONE. The performance of the proposed new routing scheme has been compared with that of replication based routing schemes (Epidemic and PROPHET) and quota based routing schemes (SW and BSW).

We used a typical scenario consists of a 4500 × 3400 m^2 area of Helsinki city. Fig. 2 shows a snapshot of our simulation scenarios.

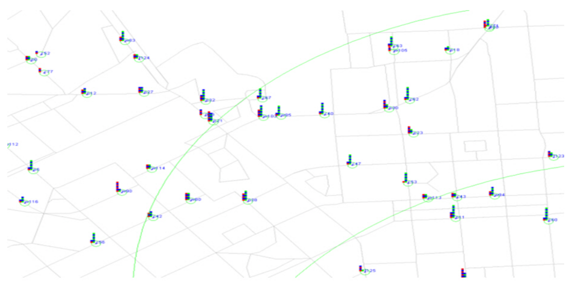

Fig.2: A Snapshot of Simulation Scenario.

We consider six (6) groups of nodes. Group 1 and

Group 3 are pedestrian groups and Group 2 is a car group. Group 4, Group 5 and Group 6 are tram groups and these three groups consist of 2 nodes each in all simulation scenarios. Pedestrians move with speeds of 0.5 - 1.5 m/s with a waiting time of 0 - 120 seconds. Cars move with speeds of 10 - 50 km/h and waiting time of 0 - 120 seconds. Trams waiting for 10 - 30 seconds at each station move with speeds of 7 - 10 km/h. We use two types of devices namely Bluetooth and 802.11 WLAN devices in the experiments; Bluetooth devices have a transmission range of 10 m and a transmission speed of 2 Mbps, while the 802.11b WLAN devices work with a transmission range of 30 m and a transmission speed of 4.5 Mbps. For the movement of nodes in Group 1 through Group 3 we use the Shortest Path Map-Based movement model. Nodes in Group 4 through Group 6 move using the Route Map-Based movement model.

We permit nodes in each group to generate a message in each 30 - 40 seconds interval and vary the message size from 500 KB to 1 MB. We run the simulation for 12 hours. In the beginning of the simulations, we allow a warm up period of 1000 seconds.

We consider three simulation scenarios. The first scenario consists of 126 nodes with 40 nodes each from Group 1 to Group 3 and 2 nodes each from Group 4 to Group 6. In this scenario, we use a TTL value of 300 for each message generated by the nodes in the groups. The second scenario is same as the first scenario except that the TTL is changed to 360 in place of 300. We use 50 nodes each from Group 1 to Group 3 and 2 nodes each from Group 4 to Group 6 for a total of 156 nodes in the third scenario. The TTL is set to 300 again for the last scenario.

In order to assess the impact of nodes buffer capacity on the network, we start with 10 MB buffer size in each node and increase the size by 10 MB at each iteration until it reaches 70 MB. In all these scenarios we set the initial number of copies of a message to 40 for both SW and BSW.

To confirm the better performance of our scheme over various message generation intervals we also used two message generation intervals namely 25-35 seconds and 35-45 seconds in all of the scenarios.

We have used three performance parameters to compare the performances of the schemes.

Delivery Ratio is defined by the ratio of the total number of messages delivered to the total number of messages generated.

Delay is the time required for a message to reach its destination. It is calculated considering only those messages that reached their destination.

Overhead is the ratio of the total number of messages relayed to the total number of messages delivered.

Fig. 3, 4 and 5 report the Delivery Ratio, Delay and Overhead by varying the buffer size of nodes but setting the TTL of a message to 300, number of nodes to 126 and message generation interval to 25-35 s. In these figures we labeled the proposed scheme as "AAR".

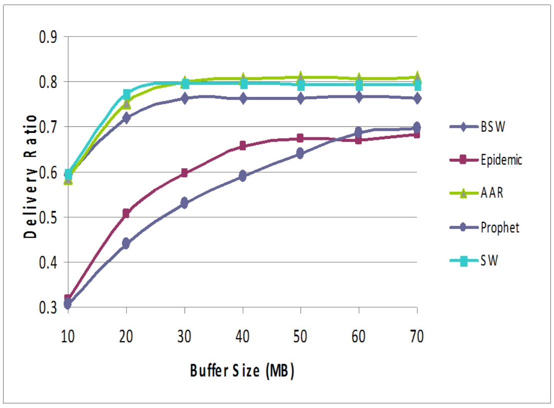

Fig.3: *Delivery Ratio: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 25-35 s).*

When the size of the buffer in each node is small, the total traffic or messages that can be carried by the nodes is low. As a result, more messages are dropped from the queue which causes nodes to fail to deliver or spread messages. With higher or larger buffer sizes, nodes can carry more messages which contributes to higher chance of delivering messages to their destinations. Thus, Delivery Ratio starts to increase with higher sizes of buffer. With higher buffer sizes, our scheme seems to maintain high Delivery Ratio compared to that of other schemes. As, we allow sources to distribute copies to agents considering their visited list and delivery predictability our scheme utilizes the resources efficiently. Agents acting as the representative of sources carry and distribute messages to other nodes which help to spread the messages faster and efficiently. Thus, agents or nodes carry only those messages for which they are good for. Since, necessary agents are being carried in the network by a source and these agents distribute or spread the message in the network our scheme experiences better Delivery Ratio (Fig. 3) than that of others.

When the buffer size of nodes is small, nodes carry less number of messages which reduces the delivery of messages to their destinations. Especially the high hop messages are dropped. Since, Delay considers the messages that are delivered to destination only all schemes have less Delay for lower buffer sizes. When the buffer size increases, nodes carry more messages which helps to deliver more messages especially the high hop messages to their destinations. As a result, the accumulated Delay increases for all schemes. Thus, Delay of all the schemes increases with increasing size of the buffer (Fig. 4).

As far as Delay is concerned, our scheme achieves lower Delay than that of the other schemes except Prophet. Prophet which delivers fewer messages shows lower Delay than the others. Since, we create agents which deliver more messages to their destinations efficiently and faster, our scheme forwards more messages than others. Thus, our scheme suffers less Delay than that of others.

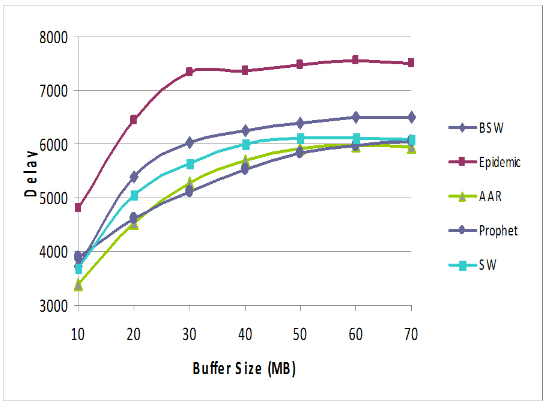

Fig.4: *Delay: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 25-35 s).*

In Fig. 5, we can see that Overhead of all the schemes decreases with the increasing size of the buffer. It is obvious from the figure that the quota based routing schemes are more resource friendly than their flooding based counterparts as they require lower overheads. The figure shows that SW is the most resource friendly protocol since only the sources distribute messages to other nodes. Despite of spreading more messages than SW, overhead of our scheme is slightly higher than the SW because we spread required number of copies of messages only.

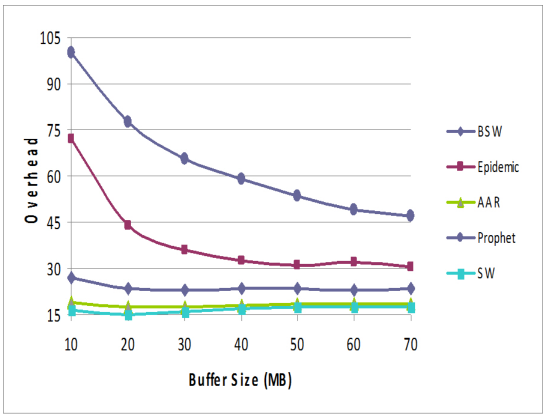

Fig.5: *Overhead: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 25-35 s).*

Fig. 6, 7 and 8 report the Delivery Ratio, Delay and Overhead by varying the buffer size of nodes but setting the TTL of a message to 360, number of nodes to 126 and message generation interval to 25-35 s.

In these scenarios, our scheme performs better than the other schemes due to the reasons mentioned earlier.

Delivery Ratio, Delay and Overhead by setting the TTL of a message to 300, number of nodes to 156 and message generation interval to 25-35 s while varying the buffer size of nodes are reported in Fig. 9, 10, and 11.

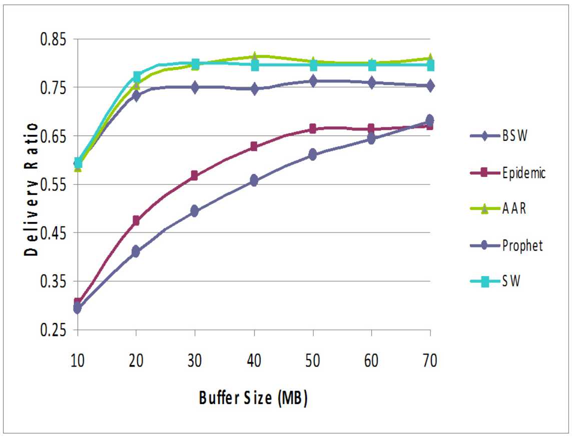

Fig.6: *Delivery Ratio: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 25-35 s).*

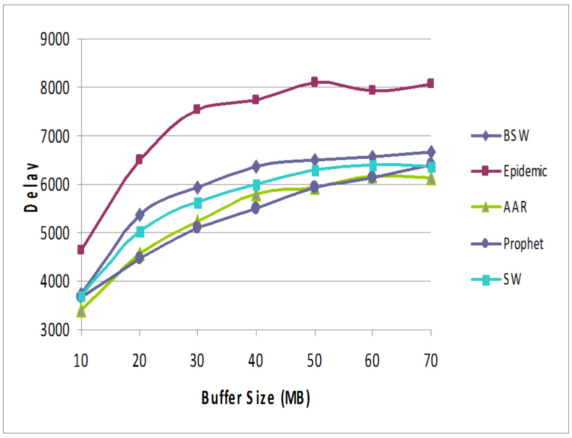

Fig.7: *Delay: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 25-35 s).*

Here we claim our proposed scheme outperforms the other routing schemes due to the factors mentioned for earlier cases. Apart from that, the Delay of our scheme becomes better than the Prophet due to more nodes helping in faster and efficient spreading and delivery of messages.

We also tested the Delivery ratio (Fig. 12, 15 and 18), Delay (Fig. 13, 16 and 19) and Overhead (Fig. 14, 17 and 20) of our scheme against the other schemes for the previously mentioned settings while setting the message generation interval parameter to 35-35 seconds. The results show that our scheme performed outstandingly with high delivery ratio, low delay and minimum overhead due to the same reasons for which it performed better in earlier cases.

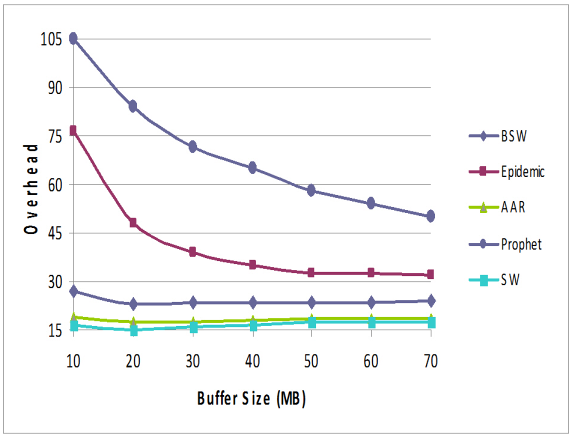

Fig.8: *Overhead: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 25-35 s).*

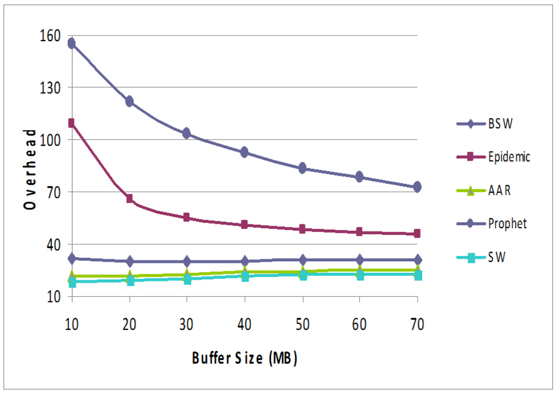

Fig.11: *Overhead: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 25-35 s).*

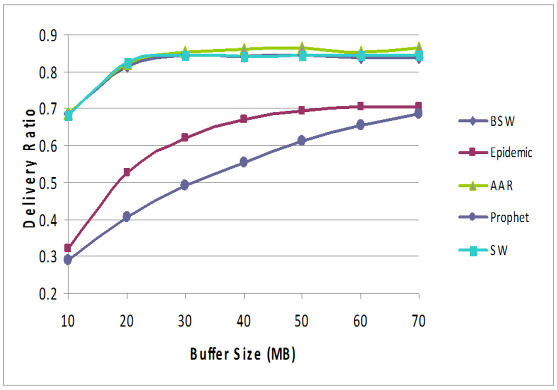

Fig.9: *Delivery Ratio: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 25-35 s).*

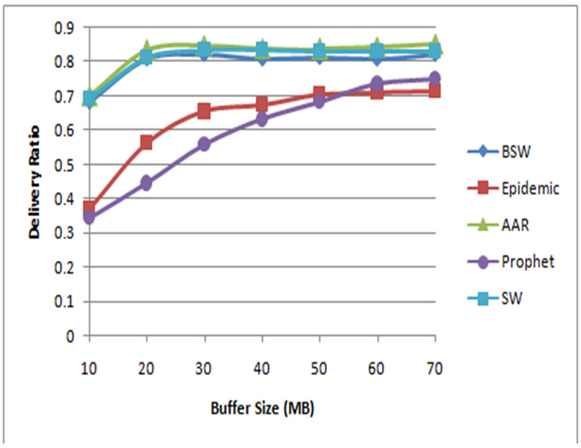

Fig.12: *Delivery Ratio: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 35-45 s).*

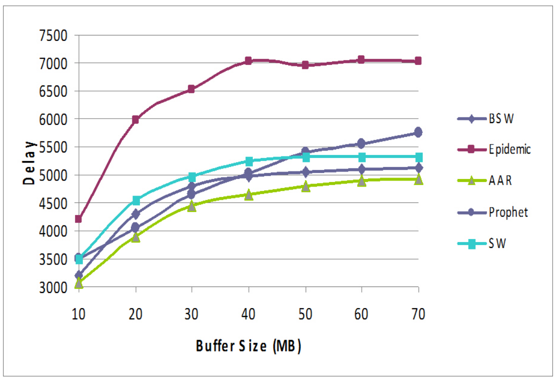

Fig.10: *Delay: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 25-35 s).*

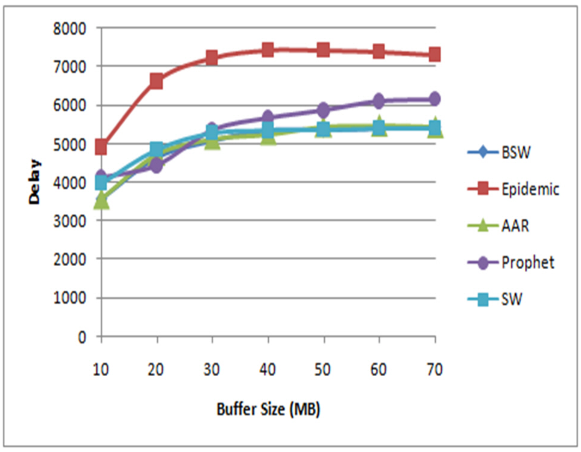

Fig.13: *Delay: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 35-45 s).*

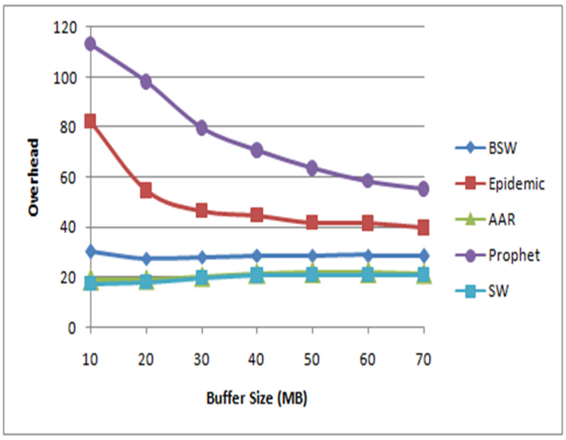

Fig.14: *Overhead: varying buffer size (Nodes = 126, TTL 300 and Message generation interval 35-45 s).*

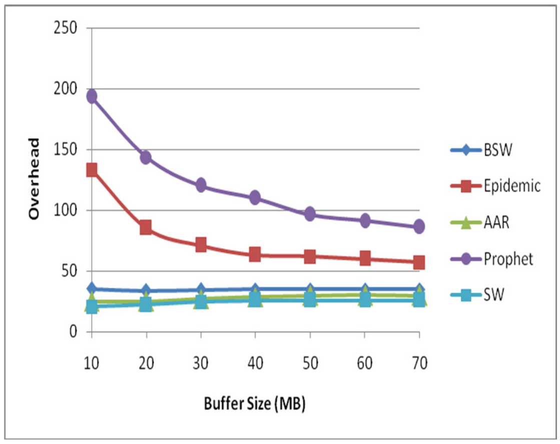

Fig.17: *Overhead: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 35-45 s).*

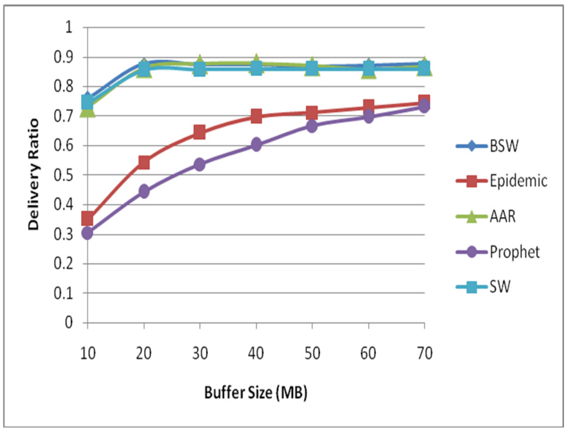

Fig.15: *Delivery Ratio: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 35-45 s).*

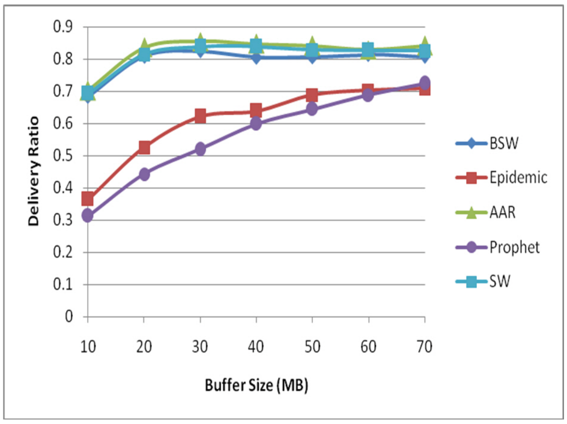

Fig.18: *Delivery Ratio: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 35-45 s).*

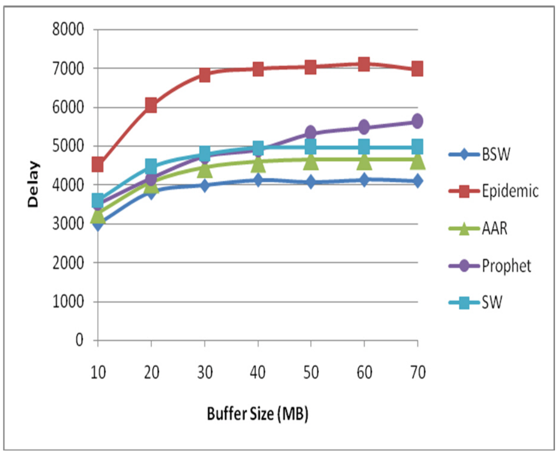

Fig.16: *Delay: varying buffer size (Nodes = 156, TTL 300 and Message generation interval 35-45 s).*

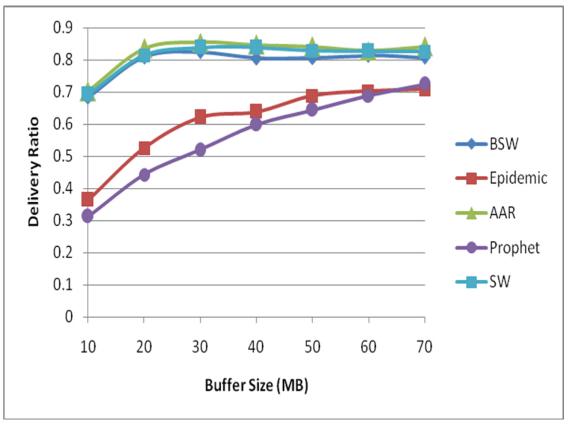

Fig.19: *Delay: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 35-45 s).*

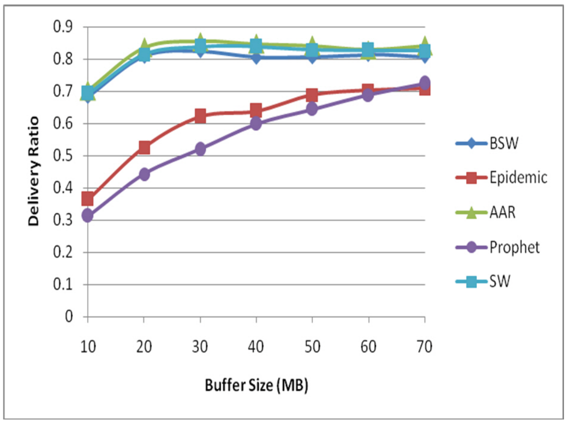

Fig.20: *Overhead: varying buffer size (Nodes = 126, TTL 360 and Message generation interval 35-45 s).*

Simulation results show that our scheme performs better than that of others in terms of Deliver ratio, Delay, and Communication overhead for varying number of nodes and different TTL values. Also, we have varied the message generation interval from 25-35s to 35-45s to determine the performance of our scheme under different traffic load. Since, all of these variable factors have significant impact on the traffic load, our scheme performs well because we have designed eq. (1) in such a way to deal with the traffic load.

5. CONCLUSION

In this work, we have designed a new agent aided routing scheme for Delay Tolerant Networks. We first allow the sources to generate unlimited copies of a message and replicate one or more copies to the agents based on their visited list and delivery predictability. An agent, a node which meets the source directly, spreads the message by forwarding a copy to a node in short of the message till it has more than one copy of the message. An agent or a node carrying a single copy of a message waits for the destination, delivers the message to its destination if found. Our main contribution is that we proposed to reduce the message delivery delay by spreading the messages faster and efficiently. Thus, we allow sources to forward only necessary number of copies of a message to the agents to keep the overhead as less as possible and increase the delivery and delay performance. Simulation results also show that our proposed Agent Aided Routing scheme achieves a better performance than the quota based routing schemes (SW and BSW) and flooding based routing schemes (Epidemic and Prophet) in different network scenarios.

In future, we will force the agents to distribute or forward a message only to the nodes which have greater probability of reaching the destination. It will help in reducing the overhead to an extent.

References

[1] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks,” *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, USA, pp. 252-259, August 2005.

[2] W. Zhao, *Routing and network Design in Delay Tolerant Networks*, Doctor of Philosophy thesis, College of Computing, Georgia Institute of Technology, 2006.

[3] M. A. T. Prodhan, R. Das, M H. Kabir, G. C. Shoja, “TTL Based Routing in Opportunistic Networks,” *Journal of Network and Computer Applications*, vol. 34, issue 5, pp. 1660-1670, September 2011.

[4] E Bulut, *Opportunistic Routing Algorithms in Delay Tolerant Networks*, Doctor of Philosophy thesis, Rensselaer Polytechnic Institute Troy, New York, February, 2011.

[5] J. Zhang, G. Luo, “Adaptive Spraying for Routing in Delay Tolerant Networks,” *Wireless Pervasive Communication*, Springer, May 2011.

[6] A. Vahdat and D. Becker, “Epidemic routing for partially connected ad hoc networks”, *Technical Report CS-20O006*, Duke University, April 2000

[7] K A. Lindgren, A. Doria, and O. Schelen, “Pobabilistic Routing in Intermittently Connected Networks,” *SIGMOBILE Mobile Computing Communications Review*, vol. 7, no. 3, pp. 19-20, July 2003.

[8] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “Maxprop: Routing for Vehicle-Based Disruption-Tolerant Networks,” *INFOCOM*, pp. 1-11. IEEE, April 2006.

[9] A. Balasubramanian, B. Levine, and A. Venkataramani, “Dtn Routing as a Resource Allocation Problem,” *SIGCOMM Computing Communications Review.*, vol. 37, no. 44, pp. 373-384, August 2007.

[10] R. Ramanathan, R. Hansen, P. Basu, R. Rosales-Hain, and R. Krishnan, “Prioritized Epidemic Routing for Opportunistic Networks,” *Proceedings of the 1st international MobiSys Workshop on Mobile Opportunistic Networking*, USA, pp. 62-66, June 2007.

[11] G. Sandulescu and S. Nadjm-Tehrani, “Opportunistic DTN Routing with Window-Aware Adaptive Replication,” *Proceedings of the 4th Asian Conference on Internet Engineering*, USA, pp. 103-112, November 2008.

[12] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility,” *Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications Workshops*, USA, pp. 79-85, March 2007.

[13] S. C. Nelson, M. Bakht, and R. Kravets, "Encounter-Based Routing in DTNs," *INFOCOM*, pp. 846-854, April 2009.

[14] http://www.netlab.tkk.fi/tutkimus/dtn/theone/., One Simulator, available at HTTP: November 2010.

Shahid Md. Asif Iqbal is Assistant Professor in Computer Science & Engineering (CSE) Department of Premier University, Chittagong, Bangladesh. He received his Bachelor degree in CSE from Chittagong University of Engineering and Technology (CUET), Bangladesh, in the year 2005. He received his Masters degree in CSE from Bangladesh University of Engineering and Technology (BUET), Bangladesh, in 2011. His major research interests lie in Wireless Networks, Network Security, VLSI, Cloud Computing, and Cognitive Radio. He started his teaching and research career at Premier University in 2006.

Alok Kumar Chowdhury received his B.Sc. degree in Computer Science & Engineering from Chittagong University of Engineering & Technology (CUET) in 2009. Currently, he is pursuing M.Sc in Computer Science & Engineering in the same University. Since 2009, he has been serving as a faculty member in the Department of Computer Science & Engineering at Premier University, Chittagong, Bangladesh. His research interest includes Artificial Intelligent, Computer Vision, Digital Image Processing, Digital Watermarking, Audio Signal Processing, Cloud Computing, etc.

Amina Akhter has been working in the department of Computer Science in Asian University for Women since July 2010. She did MEngg. in Information and Communication Technology (ICT) from Asian Institute of Technology (AIT), Thailand. Since her graduation, she had been working as a researcher in internet education and research laboratory (intERLab), AIT till joining AUW. She has been leading the students majoring in Computer Science and has taught a variety of courses since joining AUW. Her research interest focuses on but not limited to computer networks, wireless communications, mobile ad hoc networks and the ways in which ICT can be employed to develop the socio-economic infrastructure of the rising countries in Asia