Store-and-Forward Protocol Advantage in a M2ANET Network

Ahmed Alghamdi, Raid Alghamdi, John DeDourek, Przemyslaw Pochec
Faculty of Computer Science
University of New Brunswick
Fredericton, Canada
E-mail: a.alghamdi, r.alghamdi, dedourek, pochec@unb.ca

***Abstract*— In this work, the performance of a store-and-forward protocol in M2ANET is investigated. The current protocols in MANETs require an end-to-end connected path between source and destination to transmit data. In a store-and-forward protocol packets are allowed to be carried at any node for a period of time until the node gets connected to another node and is able to retransmit the packets. The store-and-forward protocol is shown to maintain communication even if the end-to-end connection never occurs. Among the proposed store-and-forward protocols the GPS (Global Positioning System) enabled produce the largest throughput in the network.**

Keywords-store-and-forward protocols; disturbance–tolerant network; MANET; M2ANET; network simulation.

I. INTRODUCTION

A network is a number of computers or devices that are connected to each other through physical or wireless links. An Ad Hoc network is a type of local area network where each individual device in this network can communicate directly with any other device in a peer-to-peer style. This arrangement eliminates the involvement of a central device that acts as a base station or a router. Ad Hoc networks operate with the absence of a fixed infrastructure. The nodes in Ad Hoc networks can be hosts as well as routers which allows a message to be transmitted form node to node through the network until it reaches its final destination [1,2].

A MANET (Mobile Ad Hoc Network) is a type of Ad Hoc network with mobile nodes moving around [1,2]. The mobile nodes in a MANET change locations and re-configure connections between the nodes as needed. Due to the spontaneous and dynamic nature of a MANET, routing is a challenge. There are two aspects of MANET networking that are affected by the node movement: (i) changes in the network topology, and (ii) intermittent connectivity [3].

A response to the changes in the network topology is typically built into the MANET routing protocols. A specialized routing protocol for MANETs, like DSR (Dynamic Source Routing) [8], discovers routes dynamically when the source node attempts to send the data, and when a route is broken the protocol initiates a route discovery again.

Intermittent connectivity calls for an altogether different approach to transmitting data in a network. When a permanent path between a source and the destination cannot be maintained, the routing protocol must rely on a store-and-forward approach [4]: the data can be transferred from node to node as a set (bundle) rather than one packet at a time [5]. Such an approach to data transmission is called Delay Tolerant Networking (DTN, also known as Disturbance Tolerant Networking) [5].

We propose to use for communication such a DTN mobile network with nodes moving randomly and sending data using a store-and-forward protocol. Such a network will appear as a formless communication medium consisting of a cloud of mobile nodes. We call this scenario M2ANET for Mobile Medium Ad Hoc Network [17].

We start by presenting a literature review in Section II. We review in details the protocols used in both MANETs and DTNs respectively. In Section III, we discuss our research objective. In Section IV, we describe our proposed solutions. In Section V, we describe the simulation environment. In Section VI, we present the results. Finally, we conclude the paper and give some ideas for future work.

II. STATE OF THE ART

A. *Protocols used in MANETs*

Many protocols have been proposed to work in MANETs. Each one of these protocols has specific properties and structure to deliver a solution to a problem facing MANETs. A wide range of these protocols have been categorized into three major categories. These categories are Proactive Routing Protocols, Reactive Routing Protocols and Hybrid Routing Protocols [7, 8].

In proactive routing protocols, each node in the MANET network stores a table holding information about the other nodes in the network. Depending on the implementation of the protocol, the table might hold information about every other node in the network or some selective nodes [7]. These tables are updated periodically or whenever the

 ISBN: 978-1-61208-211-0

topology of the networks changes [7, 8, 9]. The protocols in this category differ in the way they update the table/s and the information kept in them. Examples of protocols that fall under this category are DSDV (Destination Sequenced Distance Vector), WRP (Wireless Routing Protocol), GSR (Global State Routing), FSR (Fisheye State Routing), STAR (Source Tree Adaptive Routing) and DREAM (Distance Routing Effect Algorithm for Mobility).

Instead of updating the tables of all the nodes in the network, in Reactive Routing Protocols, updates are only preformed on the nodes that need to send data at a specific time. This is called On-demand routing [7, 8]. This means the route from the source to the destination is determined upon sending. Usually the source floods packets into the network to determine the best route to the destination. The packets flooded are small packets known as route request packets (RREQ) [8]. Based on the acknowledgment/response/replay resulting from sending (RREQ), the best route is chosen to deliver the data. Reactive/On-demand routing is further categorized into two categories known as hop-by-hop routing and source routing [7, 8]. The difference between these two categories occurs in the header of the sent packets. In source routing, the full information of the address is stored in the packet header. In hop-by-hop routing, only the addresses carried by a packet are the final destination address and the next hop address. The source routing is reported to be inefficient due to the overhead resulting from carrying too much information in the packets headers. Reactive/On-demand protocols are reviewed in [8] and include AODV (Ad Hoc On-demand distance vector), DSR (Dynamic Source Routing), ROAM (Routing On-demand acyclic multi-path), LMR (Light-weight Mobile Routing), TORA (Temporally Ordered Routing Algorithm) and ABR (Associativity-Based Routing).

Hybrid Routing protocols adopt a mix of the first and second categories' properties. Hybrid routing protocols are reviewed in [7] and include ZRP (Zero Routing Protocol), ZHLS (Zero Based Hierarchical Link State), DST (Distributed Spanning Trees based routing protocol) and DDR (Distributed Dynamic Routing).

B. *Protocols used in DTNs*

Due to the repeated end-to-end connection loss, routing in DTN (Disturbance Tolerant Network) is challenging. A store-and-forward approach is used often in such networks [4]. In Store-and-forward, a message is stored in an intermediate node until the node sees an opportunity to retransmit the message. This gives the DTNs the advantage of delivering the message without the need of an end-to-end connection. The routing protocols in a DTN are designed to overcome the problem of repeated disconnections. They are two categories of DTN protocols: Deterministic Routing Protocols and Dynamic or Stochastic routing protocols [10]. In Deterministic DTNs, the future topology of the network is known or could be predicted simplifying finding a route.

In Dynamic DTNs, the topology is not known. Dynamic Routing Protocols differ in the way they make decision to which node a message is forwarded. Two routing strategies are used in DTNs; one approach is called *flooding* (replication-based) routing and the other one is called *forwarding* [12]. In flooding [11, 12, 13], multiple copies of the same message are sent to other nodes in the network. These nodes stores these multiple copies until one of them come in contact with the destination node then; the message is retransmitted and delivered to the destination. The advantage of such a strategy is to increase the probability of a message getting delivered. Different types of flooding have been investigated in the literature. These types include; Two-Hop Relay, Tree Based Flooding, Spray and Wait [14], MaxProp strategy and Epidemic Routing [12]. In Epidemic routing, each node sends the message to be delivered to each other node in range. A node accepts a message only if it does not already have another copy of the same message in its buffer. Flooding strategy suffers from bandwidth consumption due to the multiple copies of the same message circulating in the network. The multiple copies along with the repeated transmissions, lots of storage space are wasted. This raises the need of having a recovery scheme to deal with the copies of the data left in the network after a message is delivered. One solution is to introduce a life time parameter where a message is discarded if it has been carried for a period exceeding its life time. This life time scheme is optimal since the message would not reach the destination if the life time is too short. If it is too long, the storage capacity would be wasted. Another recovery scheme introduces acknowledgments that are flooded into the network once a message is received at the final destination. Each node in the network receives such acknowledgments. Then, it deletes the corresponding message stored in its buffer. These acknowledgments could be used as a way to guarantee successful delivery [10].

Forwarding strategy uses local or global knowledge to find the best path (i.e., the existing shortest path) to deliver a message without having to create multiple copies. One forwarding algorithm is the Single Hop Transmission (Direct Delivery) where the source sends a message to the destination only if the destination is in range [10, 12, 13]. This means that the message does not propagate through the network and needs not to be stored and forwarded by any intermediate node. This type of forwarding suffers from long delays [12, 13]. First Contact Routing is another type of forwarding [10, 12, 13]. In First Contact Routing, a message is forwarded to an in-range randomly chosen neighbor. The decision made to choose a random node is not efficient since this randomly-chosen node might not be moving towards the destination. The drawback in First Contact algorithm is that a message might be exchanged between only two or three nodes all the time which causes transmission delay or even loss of data.

Another type of forwarding algorithm is location-based routing algorithm [12, 15]. This algorithm makes use of the physical location of the nodes. The physical location could be provided the Global Positioning System (GPS). The best path (i.e., the shortest) is determined upon the location of the source, the destination and the intermediate nodes. One location-based strategy is to forward the message to a node that is closer to the destination than the current node [16]. Another location-based algorithm is called Motion Vector (MoVe) which uses knowledge about location, velocity, and direction of a node to determine the closest and best path to the destination node [16].

C. *The Mobile Medium Network Model*

While a conventional view of a MANET is a (fully) connected network, we proposed in [17] to use a MANET as simply a medium for establishing a connection between two selected terminal stations. This medium is formed by the MANET nodes with forwarding capability allowing data to propagate through the medium in a way analogous to interaction between gas molecules allowing for the propagation of sound waves in a medium like air. This view differentiates explicitly between two MANET node types: (i) the terminal nodes (two communicating stations), and (ii) the mobile routing nodes (all other nodes in the MANET). We call such a network a **M**obile **M**edium MANET, or M^2ANET (pronounced "*square(d) MANET*" or "*MANET Two*") [17]. The main task of the M^2ANET is to establish communication between the terminal nodes (and not necessarily to link all the MANET nodes into a connected network). We propose to separate the network nodes into two categories: the terminal nodes and the communication nodes. A "cloud" of communication medium nodes forms a medium through which the communication channel is formed. The objective of this proposed research is to demonstrate the principles of the new M^2ANET and to establish the conditions under which a channel is established between two (fixed, or mobile) stations by means of forwarding in a mobile medium network. As opposed to a typical approach of studying MANETs, e.g. the study described in [18], in this scenario some mobile nodes may become isolated without a detrimental effect on the channel formation.

To further the idea of working with a mobile medium rather than having to deal with the individual nodes, we investigated a hypothesis that the performance of a mobile medium network depends on the characteristics of the mobile medium, rather than the performance (continuity of operation) of any one network node [19]. In a simulation experiment we switched the network nodes on and off periodically (i.e., put them to sleep) and shown that, given a sufficient number of nodes, such a network would still be able to transmit data reliably. In other words: in the experiment the performance of the network depended on the characteristics of the medium (*density*) rather than the performance of individual nodes (*on/off states*).

III. OBJECTIVES

In order to exchange messages or packets between any two nodes, the existing MANET networks require an end-to-end direct path. Without a closed path no data gets through to the destination. The same applies to the new M2ANET; the mobile medium only "conducts" the message if it is "continuous" between the source and the destination. In order to overcome this problem, a message may be carried by the intermediate nodes and may have to stay there for some time until the nodes get connected to other nodes, which essentially follows the principle of store-and-forward approach. Then, the message is retransmitted again. In other words, thanks to store-and-forward capability the mobile medium (and M2ANET) acquires some inherent storage capacity, and can breach the temporary gaps in its continuity if they occur. Our objective is to test such a scenario by simulating different versions of store–and-forward protocols for establishing communication between two selected nodes and compare them to a standard scenario when a closed path between terminal nodes is required.

IV. PROPOSED SOLUTIONS

Different versions of store-and-forward protocols are proposed in this paper. The difference between these versions is in the algorithms that are used to forward the packets. The common property which all versions share is the ability to carry/store a message for a while until a connection occurs.

These versions are; (1) First hop in the list routing (FLR), (2) closest hop routing (CHP) and (3) farthest hop routing (FHR). By introducing a GPS location (Global Positioning System), so that the distance to each node in the topology is known, (4) the closest to the destination routing (CGPS) and (5) forwarding to the hop that has the best next location to the destination (NGPS) are proposed. One last version of the store-and-forward protocol is simple flooding. In order to understand the underlying implementation of each version, a brief discussion is mandatory.

The connectivity between the nodes is stored and maintained as a matrix of 0s and 1s, which means not connected and connected respectively. In "First in the list routing", the first node spotted, by the node currently carrying a packet (the carrying hop), is the one the packet is forwarded to. In "closest hop routing", the distance between each connect node is calculated and the packet is forwarded to the one that is closest to the carrying node (in other words, the one having the strongest transmission signal). In "farthest hop routing", the packet is forwarded to the farthest node from the carrying node (in other words, the one having the weakest transmission signal).

In GPS-enabled routing, the current position of the destination is known to all the nodes in the network. In closest to destination routing, the distance between every connected node and the destination is calculated and sent to the carrying hop. The packet then is forwarded to the closest node to the destination. Since there is movement involved in the network, it cannot be guaranteed that the closest node to the destination is not moving away from the destination. To overcome this issue, forwarding to the closest *next location* to the destination is proposed. Rather than sending the packets to the closest current location of the node to the destination, they are forwarded to the node that has the closest next location to the destination.

V. SIMULATION ENVIRONMENT

A customized JAVA simulator was used to simulate the network and to develop the store-and-forward protocol. The JAVA simulator provides a controlled environment for testing the store-and-forward protocol and (unlike a standard off the shelf simulator, e.g., NS2 [20, 22]) provides a full control over the parameters and the algorithms used by the store-and-forward protocol. To simplify the simulation, some assumptions have been made. To be able to accurately monitor the data flow, it was decided that there was only one source sending and one destination giving only one data flow. To better visualize the network as well as to better understand how the intermediate nodes move, the source and the destination were assumed to be stationary with all other nodes mobile. Another assumption is related to the forwarding mechanism. It is assumed that packet transmission and delivery takes exactly 200 ms for every node (forwarding cycle). In 500 byte packets, this corresponds to the link transmission rate of 5kbps (assuming zero propagation time). In each forwarding cycle, only one packet is sent form each node if the node has a packet to transmit. Another assumption is regarding the type of the data flow and the data generator. The data generator is assumed to generate CBR traffic (Constant Bit Rate), with no acknowledgements required, at the source with fixed interval (one packet every 800 ms) between each packet generated. The packet is carried in the node for at most 10 seconds and then, if not retransmitted, is dropped. Buffers were introduced in each node buffering no more than 50 packets at a time. If the buffer is full, it is assumed that the node is not going to receive any more packets until the buffered packets get resent or dropped. In case of the buffers in the sending and receiving end nodes, a drop tail queue was introduce at the source which allow dropping the new generated packets if the buffer is full. At the receiving end, the buffer size is assumed to be infinity due to the fact that the receiving end is the final destination of the packets.

The movement of the nodes is random in our simulator. The way the nodes move is by generating a random X and Y coordinate (treated as the next location and bounded by the network area) and then moving at a constant speed towards this next location. After the next location is reached, it is set to be the current location and a new next location is generated. The movement pattern used in the simulation is the same pattern used by the SetDest utility supplied with NS2.

We investigated the randomness of the movement generated for our custom generator by calculating Variance to Mean Ration (VMR) and comparing it to the standard setdest movement generated in NS2 [22]. Throughout the simulation we divided the area into 64 quadrats, recorded node positions every 10 seconds and counted the nodes falling into 36 interior quadrats of the 8*8 grid. At each instance (101 samples used), the count in each of the 36 quadrats was used to calculate the mean and the variance. The VMR ratios for NS2 setdest and for our generator are plotted in Figure 1. The VMR values for our generator fall in the same range on the VMR value for setdest.

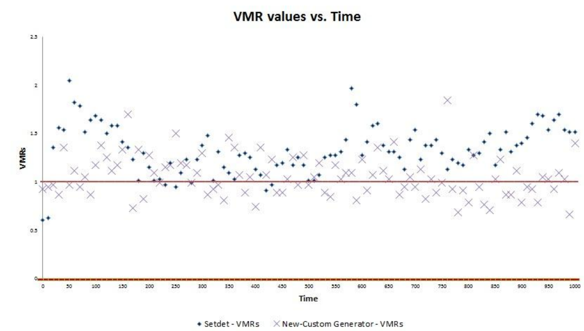

Figure 1. Comparison between setdest and our custom generator.

VI. RESULTS

We run the simulation of M2ANET using five different versions of DTN protocols proposed in Section IV. The results are compared based on the delivery ratio. We compare the number of bits received for different DTN protocols among themselves and against an idealized scenario where data would be transferred between the source and destination over a closed end-to-end path, when it exists. Two simulation scenarios were used in testing. One with N=7 mobile nodes and the other with 50.

Figure 2 shows the end-to-end connectivity between the source and destination during the simulation captured from the first simulation scenario with low density (N=7).

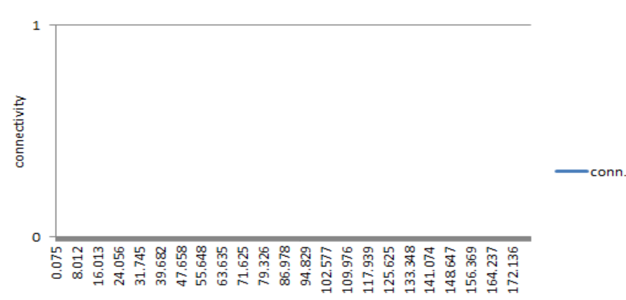

Figure 2. End-to-end connectivity for a scenario with N=7 nodes (connectivity is ZERO all the time).

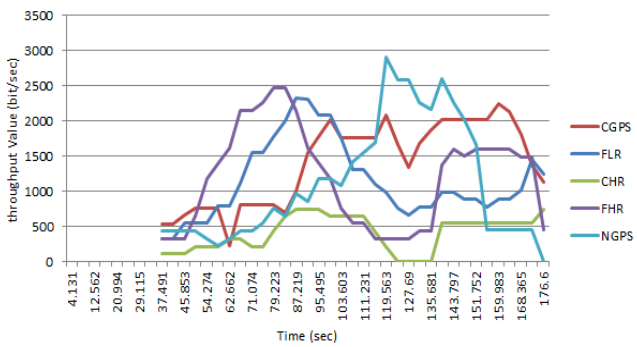

Figure 3. Instantaneous throughput for different protocols for a scenario with N=7 nodes.

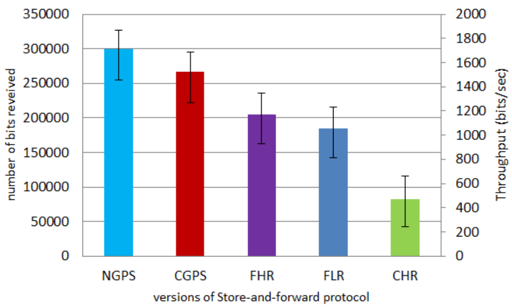

Figure 4. Total number of bits received (N=7).

The graph shows that, in this experiment, the source and the destination are never connected by a closed path. Lack of end-to-end path in conventional MANETs prevents the data from being delivered.

Figure 3 shows the throughput resulting from implementing different versions of the store-and-forward. Although, there was no end-to-end path between the source and the destination, a store-and-forward protocol allows the data to go through and be delivered to the destination. GPS-enabled versions, CGPS and NGPS, allow for the largest amount of data to be delivered.

Figure 4 shows the total number of bits received at the destination resulting from using different versions of store-and-forward protocol. Again, the graph shows that the GPS-enabled versions have the highest delivery.

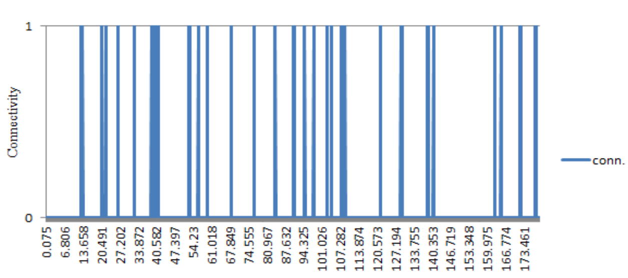

Figure 5. End-to-end connectivity for a scenario with N=50 nodes.

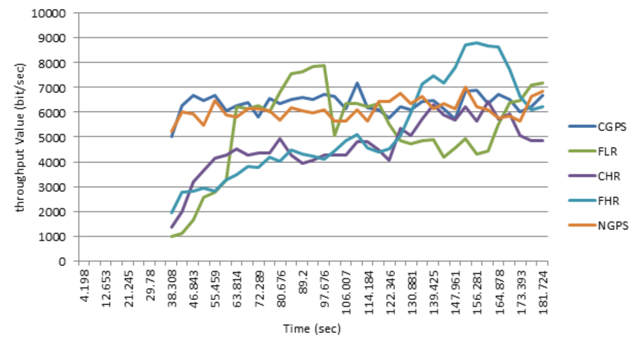

Figure 6. Instantaneous throughput for different protocols for a scenario with N=50 nodes.

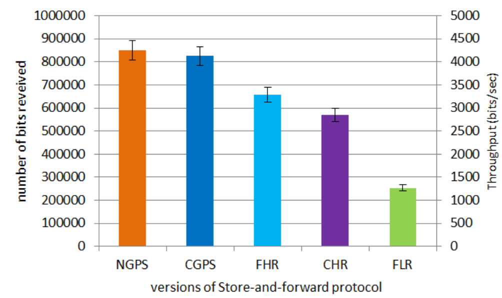

Figure 7. Total number of bits received (N=50).

Figure 5 shows the end-to-end connectivity between the source and the destination in the second experiment with high node density when the number of nodes is 50 (N=50). With more nodes used for the experiment the end-to-end connectivity occurs periodically, i.e., when the mobile nodes are positioned close one to another and forming a path from the source to the destination.

The total time of end-to-end connectivity recorded in this experiment was 6300 ms. If we use 5kbps link rate we can estimate the maximum possible number of bits delivered. The estimated number of bits delivered over the 5kbps connection is 18,000 bits. This would be the maximum throughput achievable using a conventional MANET.

The throughput for the store-and-forward protocol for N=50 is shown in Figure 6. The total number of bits delivered is in Figure 7. For NGPS protocol, we recorded 1,200,000 bits received in the course of the experiment.

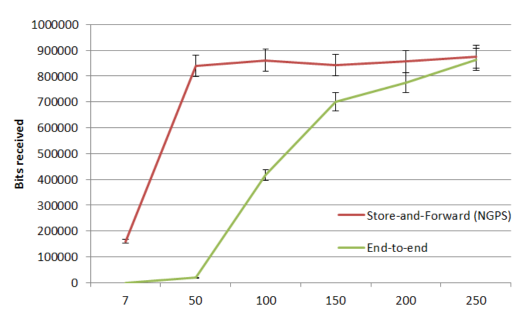

Figure 8. Store-and-forward vs. end-to-end protocols performance.

 ISBN: 978-1-61208-211-0

The clear advantage of the store-and-forward protocols over any protocol based on end-to-end connectivity only is clearly illustrated in Figure 8. The graph show the number of bits delivered from source to destination in a mobile ad hoc network for the best of the investigated store-and-forward protocols (NGPS) vs. the theoretical maximum for any end-to-end connectivity based MANET protocol all simulated at different node density (for the number of nodes ranging from 7 to 250). We observe that a store-and-forward protocol gives the top performance at N=50 nodes, while end-to-end connectivity based protocol requires at least N=250 node to achieve the same number of bits delivered.

VII. CONCLUSION AND FUTURE WORK

We used simulation to compare the number of bits delivered by a M2ANET store-and-forward protocol vs. an end-to-end connectivity based protocol. The simulation shows that, under the simulated conditions, a store-and-forward protocol offers the same throughput (i.e. bits delivered) as connectivity based protocol but with only 20% of the mobile nodes required (N=50 vs. N=250) in M2ANET.

In the future, we plan to investigate the role of mobile node characteristics in M2ANET for example the buffer size and the data retention time at the node on the throughput of M2ANET. We also plan to investigate other performance characteristics of store-and-forward protocols in M2ANET like the propagation delay.

VIII. ACKNOWLEDGMENTS

A part of this work is sponsored and funded by the Ministry of Higher Education of Saudi Arabia through the Saudi Arabian Cultural Bureau in Canada.

REFERENCES

[1] A. Seneviratn and B. Sarikaya, (1998), Cellular networks and mobile internet, *Computer Communications,* 21(14), pp. 1244-1255.

[2] S. Basagn, M. Conti, S. Gioradano and I. Stojeneoric, (2004), Mobile Ad Hoc Networking. Willy-IEEE Press.

[3] Z. Zhang, (2006), Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges, *Communications Surveys & Tutorials, IEEE* , vol.8, no.1, pp. 24-37.

[4] F. Dai, S. Yang and J. Wu, (2007), Logarithmic Store-Carry-Forward Routing in Mobile Ad Hoc Networks, Parallel and Distributed Systems, IEEE Transactions on , vol.18, no.6, pp. 735-748.

[5] S. Burleigh, et al., (2003), Delay-Tolerant Networking: An Approach to Interplanetary Internet, *IEEE Communications Magazine*, pp. 128-136.

[6] K. Fall, (2003), A Delay-Tolerant Network Architecture for Challenged Internets. Karlsruhe, Germany, *SIGCOMM'03*, pp. 27-34.

[7] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, (2003), A Review of Routing Protocols for Mobile Ad Hoc Networks. Elsevier B.V.

[8] N.S. Kulkarni, I. Gupta and B. Raman, (2009), On Demand Routing Protocols for Mobile Ad Hoc Networks: A Review, *Advance Computing Conference, 2009. IACC 2009. IEEE International* , pp. 586-591.

[9] L.A. Latiff and N. Fisal, (2003), Routing Protocols in Wireless Mobile Ad Hoc Network - a review, *Communications, 2003. APCC2003. The 9th Asia-Pacific Conference on*, pp. 600- 604.

[10] A. Socievole, F. De Rango, and C. Coscarella, (2011), Routing approaches and performance evaluation in delay tolerant networks, *Wireless Telecommunications Symposium (WTS)*, pp. 1-6.

[11] M. Balakrishnan, K. Birman, S. Pleisch, and R. Renesse, (2006), MISTRAL: Efficient Flooding in Mobile Ad-hoc Networks. [Electronic Version]. MobHoc, Florence, Italy.

[12] M. Atique and R.S. Mangrulkar, (2010), Routing Protocol for Delay Tolerant Network: A Survey and Comparison. *Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference,* pp. 210-215.

[13] A. Islam and M. Waldvogel, (2008), Reality-Check for DTN Routing Algorithms. *Distributed Computing Systems Workshops, 2008. ICDCS '08. 28th International Conference,* pp. 204-209.

[14] W. Yu, C. Wu and X. Hu, (2010), Spray and Routing for Message Delivery in Challenged Networks, Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on, pp. 472-475.

[15] Y-B. Ko and N.H. Vaidya, (2000), Location-adid Routing (LAR) in mobile adhoc networks, *Wireless Networks* vol.6, pp. 307-321.

[16] J. LeBrun, C. Chuah, D. Ghosal and M. Zhang, (2005), Knowledge-Based Opportunistic Forwarding in Vehicular Wireless Ad Hoc Networks*Vehicular Technology Conference, 2005, VTC 2005-Spring. 2005 IEEE 61st*, pp. 2289-2293.

[17] J. DeDourek and P. Pochec, (2011), M^2ANET: a Mobile Medium Ad Hoc Network, Wireless Sensor Networks: Theory and Practice, WSN 2011, Paris, France, pp. 1-4.

[18] C. Bettstetter and J. Zangl, (2002) How to achieve a connected ad hoc network with homogeneous range assignment: an analytical study with consideration of border effects", 4th International Workshop on Mobile and Wireless Communications Network, pp. 125-129.

[19] K. Patel, J. DeDourek and P. Pochec, (2011), M^2ANET Performance Under Variable Node Sleep Times, The Third International Conference on Advances in Future Internet, AFIN2011, Nice, France, pp. 31-34.

[20] J. Zhang, W. Li, D. Cui, X. Zhao and Z. Yin, (2009), The NS2-Based Simulation and Research on Wireless Sensor Network Route Protocol, *Wireless Communications, Networking and Mobile Computing, 2009, WiCom '09, 5th International Conference on*, pp. 24-26.

[21] H. Ekram and T. Issariyakul, (2009), Introduction to Network Simulator NS2. Springer.

[22] J. Taylor, (1983), Quantitative Methods in Geography: An Introduction to Spatial Analysis, illustrated version, Waveland Pres.