International Journal of Engineering & Technology
CENTRE OF PROFESSIONAL RESEARCH PUBLICATIONS
ISSN 2049-3444

International Journal of Engineering and Technology Volume 3 No. 3, March, 2013

Adaptive Priority Routing Protocol for DTN Networks

Mamoun Hussein Mamoun, Saud Barrak
Faculty of Computer Science & Information
Al Jouf University

ABSTRACT

In *DTN* networks the existence of a simultaneous path between a source and a destination is not assumed. This type of networks completely breaks the main assumptions on which *MANET* routing protocols are built. Routing in *DTN* networks is usually based on some form of controlled flooding. But often this results in very high resource consumption and network congestion. In this paper we propose a fuzzy-based routing protocol for *DTN* networks, called Adaptive Priority Routing Protocol (*APRP*). It uses three local parameters (forward transmission count, buffer size and remaining life time) from each message as inputs to fuzzy system to prioritize which message is to be transmitted next from the buffer. There is no need to know a priori information about network. Simulation results are used to draw conclusions regarding to the proposed routing algorithm and compared it with well known routing protocols: Epidemic, PRoPHET, Spray and Wait routing protocols. Conducted experiments showed that our proposed algorithm exhibits superior performance with respect to the well known routing protocols in terms overall average hop count, delivery of messages, average message delay and average buffer occupancy time.

Keywords: *DTN Routing; Fuzzy-based Routing; New Routing Protocol; Adaptive Priority Routing*

1. INTRODUCTION

DTN networks are one of the most interesting evolutions of classic Mobile Ad Hoc Networks (*MANET*). The main assumption of *MANET* environments is that a sender and a destination are connected to the network at the same time. If the destination is not connected when the sender wishes to transmit messages, they get dropped at some point of the network. However, in a pervasive networking environment, nodes will be seldom connectable at the same time through a multi-hop path. For example, devices that users carry with them might be only sporadically attached to the Internet, e.g. when the user moves close to an Access Point. In other words, it is foreseeable a scenario in which a large number of wireless devices and limited-size networks will be *just occasionally* connected to each other. *DTN* networks aim at make users able to exchange data even in such a disconnected environment, by opportunistically exploiting any nearby device to move messages closer to the final destination. To this end, legacy protocols designed for *MANET* should be drastically redesigned [1], [2], [3]. Currently, envisioning routing and forwarding protocols1 for *DTN* networks is one of the most exciting topics [4]. In *DTN* networks, the traditional routing paradigm of Internet and *MANET*, in which routes are computed based exclusively on topological information, is not adequate anymore. A first approach to routing in *DTN* networks is some variation of controlled flooding: Messages are flooded with limited Time-To-Live (*TTL*), and delivered to the destination as soon as it gets in touch with some node that received the message during the flood [5]. Many researchers have proposed new routing protocols such as Epidemic [5], Prophet [6], Spray-and-Wait[7], Spray-and-Focus [8], MaxProp [9] and *ORWAR* [10] to handle this specific problem for *DTN*.

In this paper, we propose a fuzzy-based routing mechanism, called Adaptive Priority Routing Protocol (*APRP)* in *DTN*. Our mechanism is simple with low overhead. It uses only three parameters, namely, *Forward Transmission Count* (*FTC*), *Message Size (MS) and remaining life time (RLT)*, to prioritize messages for transmission that are stored in its buffer. *FTC* information is updated whenever messages are transmitted during the contact time. The order of messages, to be transmitted next, is done using simple fuzzy rules. The remainder of this paper is organized as follows: Section 2 presents the related work. In section 3 our proposed mechanism is described. Simulation and results of our proposed routing protocol can be found in section 4. Section 5 discusses our conclusion.

2. RELATED WORK

In this Section, we present an overview of the significant concepts on routing protocols for DTN, namely Epidemic [5], PRoPHET [6], Spray and wait [7], along with their relative pros and cons.

2.1 Epidemic

The Epidemic routing [5] protocol is a flooding-based scheme. Each node has two buffers, the first one for storing the messages generated by itself, and the second one for those messages received from other nodes. Each message is tagged with a unique ID. Each node also maintains a list of message IDs that it is currently holding in its buffer called the Summary Vector. When two nodes meet, they exchange their Summary Vectors with each other. By comparing these Summary Vectors, the nodes exchange those messages which they do not have with them. When this operation of message exchange is completed, all nodes have the same messages in their buffers. This creates a large amount of redundancy in the network, which incurs significant demand on both bandwidth and buffer capacity, but at the same time, makes it extremely robust to node and network failure. The simulation results obtained in this work show that for this protocol, the message delivery ratio is very high and the message is delivered in minimum amount of time if sufficient resources are available.

2.2 PRoPHET

In PRoPHET (Probabilistic Routing Protocol using History of Encounters and Transitivity) [6], before sending a message, each node estimates a probabilistic metric called Delivery Predictability for each known destination. It indicates the probability of successful delivery of a message to the destination from the source node. The calculation of the Delivery Predictability is based on the history of encounters between nodes or history of visits to certain locations. When two nodes meet, they exchange their Summary Vectors containing the Delivery Predictability. If two nodes are often encountered, they have high Delivery Predictability to each other. On the other hand, if a pair of nodes does not find each other for a period of time, they are not good forwarders of the message to each other. Hence, the Delivery Predictability value must decrease with time. Thus, a message is forwarded to a node from a set of available nodes which has the higher value of Delivery Predictability among them to the destination node. The simulation results obtained in this work show that this protocol has less message exchanges, less communication overhead, less delay, and higher delivery success rate as compared to the Epidemic Routing.

2.3 Spray and Wait

This protocol [7] provides an interesting technique to control the level of flooding. The message is mainly delivered in two phases: the Spray phase and the Wait phase. For every message originating at the source node, L copies of the message are spread over the network by the source node and other nodes receive a copy of the message from the source node to L distinct relays. In the Wait phase if the destination was not found during the spray phase, each relay node having a copy of the message performs the direct transmission. The simulation results show that this protocol has less number of transmissions and less delivery delay as compared to the Epidemic Routing.

3. PROPOSED ROUTING ALGORITHM

Previous works mentioned in section 2 are optimized under specific node density or node mobility pattern. Moreover, a few of them need special equipment and/or complicated algorithm to predict encounters. We propose an algorithm whose parameters are easy to calculate, and they adapt themselves to the network environment. The protocol is also suitable for arbitrary movement of nodes.

In this section we describe our proposed algorithm, (*APRP*) in more detail. *APRP* uses fuzzy technique to prioritize messages that are stored in the buffer, and then transfers as many messages from the buffer to its peer as possible, during its contact time. *APRP* uses simple store, forward and replication [11, 12] mechanism. It uses only three parameters, Forward Transmission Count (*FTC*), Message Size (*MS*) and remaining life time (*RLT*) , which are updated using simple rules. Associated with every message there is the *FTC* of the message transmitted together, *FTC* is just a count and overhead is negligible. By simulation, we have shown that *APRP* algorithm works well in all scenarios, with various node densities and distribution.

3.1 Input Variables for Fuzzy System

The opportunity of the source node to come in contact with the destination node directly and transfer the message is very low. Our approach uses forwarding and replication strategies [13, 14, 15] to increase the message delivery probability and reduce the message average delay.

Forwarding messages with longer remaining life time (*RLT*) are scheduled to be sent first because they have a higher probability to reach its destination. then the sequence of delivering messages is very important. Because bandwidth is limited, node cannot transmit all messages it has in its buffer to its peer node during contact time, as the contact time is limited and usually short, replication strategy is used in which the Probability of successful delivery of message increases as the message is made available at many nodes. Count of Duplicated Messages (*CDM*)or copies in the network, is used as our first metric for message transmission decision. The message that already has high *CDM* should be assigned lower priority for next transmission. Although hop count is a good metric to measure distance (how far the message has been relayed), it cannot precisely estimate the value of *CDM* information to how many nodes the message have been spread. A new metric to estimate *CDM* called *Forward Transmission Count* (*FTC)* is better than hop count as a measure for *CDM* [16] and the calculation of it is very simple.

In our approach we consider the message size which affects the number of partial message transmission. For example, in very short contact time node cannot deliver large message successfully. The probability of successful transmission for smaller messages is higher. In *DTN* network nodes lack the knowledge of contact schedule. Delivery decision has to be based on local information. In order to decrease the amount of partial message transmission prioritizing message delivery by their sizes is reasonable.

3.2 Membership Functions of Input Variables

APRP uses *RLT*, *FTC* and *MS* as inputs to the fuzzy rule system to prioritize which message is to be transmitted next from the buffer. From our discussion in previous subsections, forwarding messages with longer *RLT* should have more priority to sent, messages with low *FTC* should have more priority than messages with high *FTC*. Similarly, smaller messages should have more priority than larger messages, to reduce the numbers of partial message transmission.

The membership function of *FTC* is divided into 3 sections, low, medium and high, with linear symmetric shape. Figure 1 illustrates the degree of membership function of *FTC* for 200 nodes *DTN*. The most important problem of fuzzy logic is to define the appropriate ranges for each input. We run simulation in different environments in order to investigate appropriate ranges, in terms of percentage of network nodes, for any arbitrary size of *DTN* network. We found that the maximum number of *FTC* is about 12% of number of nodes, for various network sizes from 40 to 200 nodes.

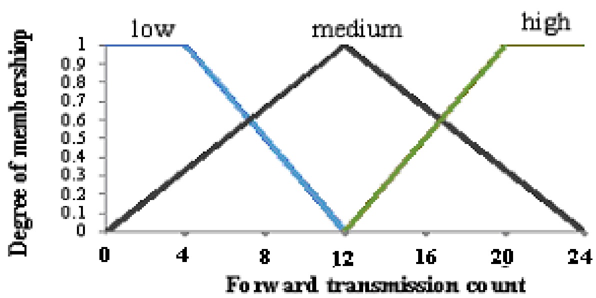

Fig.1. Memberships function of transmission count

Figure 2 illustrates the degree of membership function of message size. We again divide message size into 3 categories, small, medium and large. We assume the range of message size is known at the time the network is deployed. If the message size is known to be between 0 and 100,000 byte, we can simply set the membership function (symmetric shape) as its shown in figure 2. Thus, the fuzzy membership functions are adapted depending on a priori available information like network node numbers and message size range.

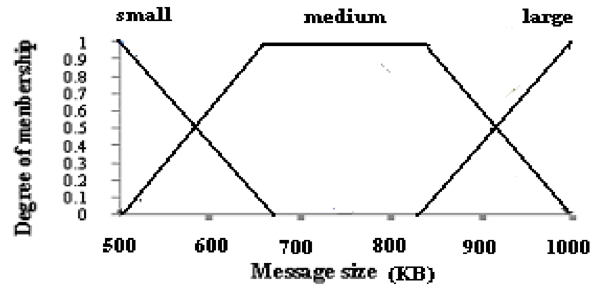

Fig.2. Membership function of message size

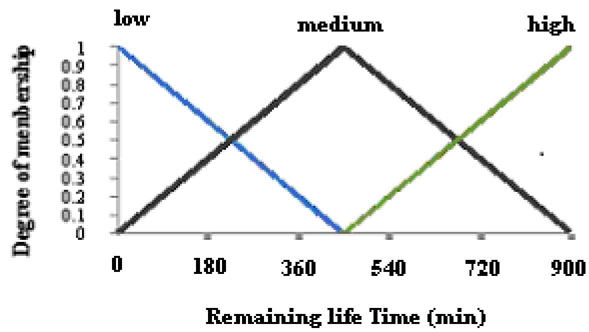

Fig.3. Membership function of remaining life time

Figure 3 illustrates the degree of membership function of *RLT*. We again divide it into 3 categories, small, medium and large. We assume that the range of *RLT* is known. If it is known to be between 0 and 300 minutes, we can simply set the membership as its shown in figure 3.

3.3 Output Variable and Fuzzy Rules

Since there are three inputs (each with 3 fuzzy categories), we simply divide the message in to 27 *buffer sections* (*BS*), and set up the rules as follows:

1. if *FTC* is low and *MS* is small and *RTL* is low then BS is BS_0
2. if *FTC* is low and *MS* is small and *RTL* is medium then *BS* is BS_1
3. if *FTC* is low and *MS* is small and *RTL* is high then BS is BS_2
4. if *FTC* is low and *MS* is medium and *RTL* is low then *BS* is BS_3
5. if *FTC* is low and *MS* is medium and *RTL* is medium then *BS* is BS_4
6. if *FTC* is low and *MS* is medium and *RTL* is high then *BS* is BS_5
7. if *FTC* is low and *MS* is large and *RTL* is low then *BS* is BS_6
8. if *FTC* is low and *MS* is large and *RTL* is medium then *BS* is BS_7
9. if *FTC* is low and *MS* is large and *RTL* is high then *BS* is BS_8
10. if *FTC* is medium and *MS* is small and *RTL* is low then *BS* is BS_9

11. if *FTC* is medium and *MS* is small and *RTL* is medium then *BS* is BS_{10}
12. if *FTC* is medium and *MS* is small and *RTL* is high then *BS* is BS_{11}
13. if *FTC* is medium and *MS* is medium and *RTL* is low then *BS* is BS_{12}
14. if *FTC* is medium and *MS* is medium and *RTL* is medium then *BS* is BS_{13}
15. if *FTC* is medium and *MS* is medium and *RTL* is high then *BS* is BS_{14}
16. if *FTC* is medium and *MS* is large and *RTL* is low then *BS* is BS_{15}
17. if *FTC* is medium and *MS* is large and *RTL* is medium then *BS* is BS_{16}
18. if *FTC* is medium and *MS* is large and *RTL* is high then *BS* is BS_{17}
19. if *FTC* is high and *MS* is small and *RTL* is low then *BS* is BS_{18}
20. if *FTC* is high and *MS* is small and *RTL* is medium then *BS* is BS_{19}
21. if *FTC* is high and *MS* is small and *RTL* is high then *BS* is BS_{20}
22. if *FTC* is high and MS is small and *RTL* is low then *BS* is BS_{21}
23. if *FTC* is high and *MS* is medium and *RTL* is medium then *BS* is BS_{22}
24. if *FTC* is high and *MS* is medium and *RTL* is high then *BS* is BS_{23}
25. if *FTC* is high and *MS* is medium and *RTL* is low then *BS* is BS_{24}
26. if *FTC* is high and *MS* is small and *RTL* is medium then *BS* is BS_{25}
27. if *FTC* is high and *MS* is small and *RTL* is high then *BS* is BS_{26}

BSi is the buffer position *i* in buffer space, where *BS0* is at the front or head of the buffer queue and has highest priority for transmission and BS_{26} is at the tail of the queue. Figure 4 shows the degree of membership function for each *BS*. For defuzzifier process, Center Of Area (*COA*) is applied to create a crisp value. The priority of a message, *P*, is *1 – COA*, and the messages in buffer are all sorted by values of their *P*, from highest to lowest, where highest *P* value is at the head of the queue, and lowest *P* value is at the tail.

The pseudo code of *APRP is* similar to what is proposed in [6] with small modification given in Figure 5. This mechanism is required so that destination node after receiving the intended message sends an acknowledgement. Node that received this acknowledgement will remove the corresponding message from its buffer queue. Every node will hold the list of acknowledgements and this list will always be exchanged at the beginning of each contact time.

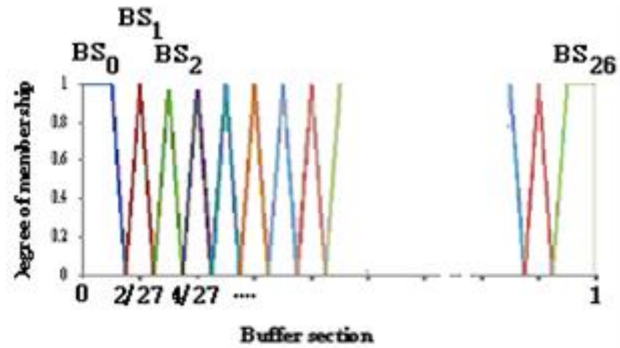

Fig. 4. Membership function of buffer section

```
Function compute_fuzzy(msgi)
    Variables:
        listCombinedArea  /*set of areas*/

    /*define degree of membership of low, medium,
    high for FTC, small, medium, large for size and
    low, medium, high for RLT*/
    setMembershipDegree(msgk)
    listCombinedArea:= computeFuzzyRule(msgk)
    return calCOA(listCombinedArea)
end function compute_fuzzy
```

Fig.5. Pseudo-code for the function which is used in defuzzifier process

4. SIMULATION AND RESULTS

4.1 Simulation Setup

To validate behavior and evaluate the performance of *APRP*, we use ONE (Opportunistic Network Environment) simulator [12] to implement it and other DTN routing protocols, including Epidemic, Prophet and Spray and Wait routing.

In this work, we assume that all the nodes are mobile in nature, e.g. modern mobile phones or similar devices. The nodes communicate with each other using Bluetooth at 2Mbit/sec data rate with 10m of radio range. For our simulations, we have used a part of the Helsinki downtown area (4500x3400m) as prescribed in [12]. We divided the total number of nodes into four different equal sized groups. Group 1 and Group 3 nodes are pedestrians which move at random speeds of 0.5-1.5 m/s with pause times of 0-120 sec. Group 2 nodes are cars which move at the speeds of 2.7-13.9 m/s with pause times of 0-120 sec. Group 4 nodes are trams which move at the speeds of 7-10 m/s with pause time of 10-30 sec.

In addition to normal roads, we have added some trams routes to the map for trams. Groups 1, 2, 3 nodes have up to 5MB of free buffer space while Group 4 nodes have 50MB of free space for storing and forwarding the messages. Nodes generate one new message on average after every 25 to 35 seconds. The message size varies between 500KB to 1MB and the message lifetime is se t to 300 minutes. Each simulation ran for 43200 seconds for all six different protocols.

We chose only the shortest path map based movement model as mobility model for our simulations. In the shortest path map based movement model, the nodes move on a path defined in the form of maps. They choose the shortest path from the source to destination from that map path. We obtained the results for average hop count, message delivery and average message delay for all protocols with the different values of number of nodes, node speeds, and the message time to live (*TTL*) field. These are used to compare the performance of the above mentioned four protocols. The following setting and configurations are used while varying the aforementioned fields:

- *Varying the number of nodes:* We increased the number of nodes from 40 to 240 with an increment of 10 nodes in each group each time. The time to live field is set to 300 minutes and all nodes moves according to their respective group speeds.

- *Varying the Speed of nodes:* We varied the speed of all group nodes from 2.5 to 15 m/s. The total number of nodes is kept fixed at 200 and the time to live field is set to 300 minutes.

- *Varying the Message TTL:* We varied the *TTL* from 50 to 300 minutes. The total number of nodes is kept fixed at 200 and the speed of all group nodes is kept fixed at 2.5-5.0 m/s. We use the following performance metrics in our comparison:

- *Average hop count:* It is the average number of intermediate nodes between source and its destination in which a message is traveled.

- *Delivered messages:* It is the number of messages that are successfully delivered during simulation.

- *Average message delay*: It is the average of the difference between the message delivery time and the message creation time of all the delivered messages to the destination.

- *Average buffer occupancy time*: It is the average of the time for which messages were stored in the nodes buffer

4.2 Simulation Results

The results obtained form our simulations are depicted in Figures 6 to 9. In Figure 6, it is observed that the value of average hop count increases with an increase in the number of nodes. Its value is maximal when using the Epidemic protocol and minimal when using our proposed routing protocol (*APRP*). Any other protocols performance is in between that of the Epidemic routing protocol and *APRP*. Thereby a message sent from a source needs fewer hop count than before to arrive to the destination in our approach.

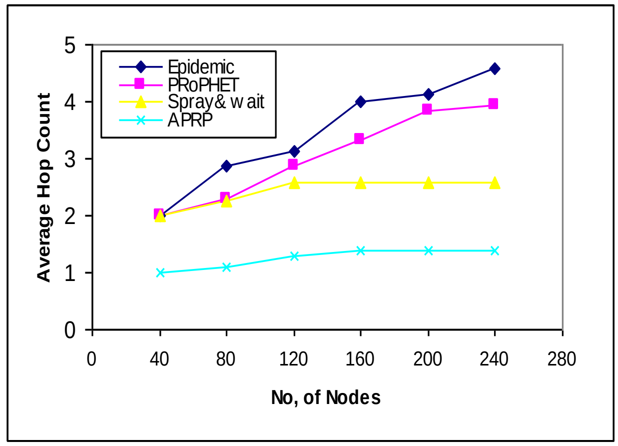

Figure 6. Average hop counts versus no. of nodes

Figure 7 shows that the number of messages delivered peaks at node's speed of 5 m/sec and then decreases with an increase in speed for all protocols. Its value is maximal for *APRP* and minimal for Epidemic protocol. Again we prove that our approach is the best.

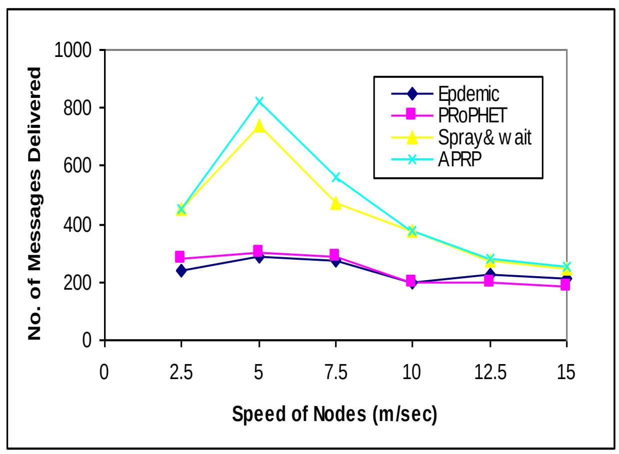

Figure 7. No. of messages delivered versus speed of nodes

In Figure 8, it can be observed that the average message delay time increases when the time to live (TTL) increases, for all protocols. Its value is maximal when using the PRoPHET protocol, and minimal when using the *APRP routing* protocol. Again we prove that our approach is the best.

Figure 9, it is observed that the average buffer occupancy time is maximal when using *APRP* and minimal when using PRoPHET routing. Thereby a message will be stored for a longer time in buffer with our approach than with Epidemic, PRoPHET AND Spray& wait.. Again we prove that our approach is the best.

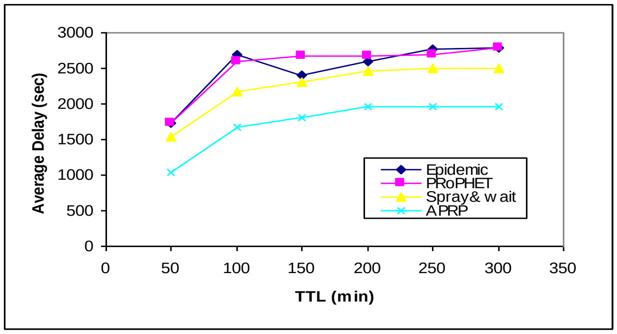

Figure 8. Average message delay versus TTL

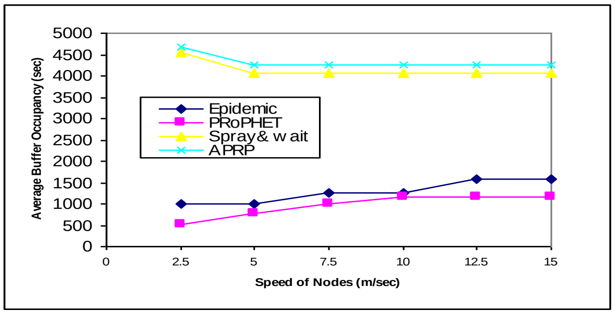

Figure 9. Average buffer occupancy time versus speed of nodes

5. CONCLUSION

In this paper we propose a new fuzzy decision system, namely Adaptive Priority Routing Protocol (*APRP)*, to prioritize messages in the buffer for transmission during next "contact" of the node. Three parameters were used which are simply available locally at the nodes. During transmission it is needed only to pass an extra number (*FTC*) along with the actual message to the peer. The fuzzy membership functions can be adaptively constructed based on known network parameters like number of nodes range of message-lengths and *TTL*. The fuzzy decision mechanism is very simple compared to complex prediction mechanisms used in many other *DTN* protocols. In spite of that, we have shown that *APRP* is the best performing protocol, in terms of average hop count, message delivery (no. of messages delivered successfully) average message delay...and average buffer occupancy time. Furthermore, *APRP* is also less sensitive to the chosen parameters (choice of fuzzy membership function), than other algorithms (i.e. choice of *L for Spray and wait routing protocol*).

REFERENCES

[1] K. Fall, "A delay-tolerant network architecture for challenged internets", Proc. of ACM SIGCOMM, pp 27-34, Aug., 2003.

[2] S. Jain, K. Fall, and R. Patra, "Routing in a delay-tolerant network", Proc. of ACM SIGCOMM, pp 145-158, 2004.

[3] J. Scott, P. Hui, J. Crowcroft, C. Diot, "Haggle: A Networking Architecture Designed Around Mobile Users" Proc. of IFIP WONS, 2006.

[4] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic Networking: data forwarding in disconnected mobile ad hoc networks", IEEE Communications Magazine, vol. 44, no. 11, Nov. 2006.

[5] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks", Tech. Rep. CS-2000-06, CS Dept., Duke University, April 2000.

[6] A. Lindgren et al, "Probabilistic Routing in Intermittently Connected Networks", Mobile Comp. and Comm. Rev, vol. 7, no. 3, July 2003.

[7] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: Efficient routing in intermittently connected mobile networks", In Proceedings of ACMSIGCOMM workshop on Delay Tolerant Networking (WDTN'5), pp 252-259, 2005.

[8] T. Spyropoulos, K. Psounis, and Cauligi S. Raghavendra, "Spray and Focus Efficient Mobility-Assisted Routing for Heterogeneous and correlated Mobility", in Proc. PerCom. Workshops apos, pp 79-85, March 2007.

[9] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption- Tolerant Networks", In Proc. IEEE Infocom, pp. 1-11, IEEE, April 2006.

[10] Gabriel Sandulescu and Simin Nadjm- Tehrani, "Opportunistic DTN Routing with Window-aware Adaptive Replication", AINTEC'08, November 18-20, Bangkok, Thailand.

[11] Jian Shen, Sangman Moh, Ilyong Chung, "Routing Protocols in Delay Tolerant Networks: A Comparative Survey", The 23rd International Technical Conference on Circuits/Systems, Computers and Communications, pp. 1577 - 1580, 8 July, 2008.

[12] Ari Keranen and Jorg Ott, "Increasing Reality for DTN Protocol Simulations", Technical report, Helsinki University of Technology, Networking Laboratory, July 2007

[13] Z. Zhang, "Routing in Intermittently Connected Mobile Ad Hoc Networks Overview and Challenges", Communications Surveys & Tutorials, IEEE, 2006.

[14] Kalantari, M. La, R.J. Dept. of Electr. & Comput. Eng., Maryland University, College Park, MD, "A DTN packet forwarding scheme inspired by thermodynamics", Information Sciences and Systems, CISS, pp. 1216-1221 , 19-21 March, 2008.

[15] Yin-Ki Ip, Wing-Cheong Lau On-Ching Yue, "Forwarding and Replication Strategies for DTN with Resource Constraints, Vehicular Technology"., Conference, pp. 1260-1264, Dublin, 22-25 April, 2007.

[16] A. Mathurapoj, C. Pornavalai, "Fuzzy-Spray: Efficient Routing in Delay Tolerant Ad-hoc Network Based on Fuzzy Decision Mechanism", Fuzzy Systems, 2009. FUZZ-IEEE Aug. 2009 pp: 104 – 109.