Distance-Aware Routing with Copy Control in Vehicle-Based DTNs

Wei-Zen Lo Jhih-Siao Gao Shou-Chih Lo
Department of Computer Science and Information Engineering
National Dong Hwa University
Hualien 974, Taiwan ROC
e-mail: sclo@mail.ndhu.edu.tw

***Abstract*—Delay-Tolerant Networks (DTNs) are special types of network environments that are subject to delays and disruptions. Traditional end-to-end routing fails in DTNs due to intermittent connections. We apply the concept of DTN to vehicular environments, and propose a new routing protocol that applies both diversity and accuracy in routing decisions. Our proposed routing has one excellent feature that supports its use for a distributed location discovery service. We evaluate this routing, using a real street map, in terms of different performance aspects. Experimental results show the superiority of this protocol over other existing ones.**

Keywords-VANET; DTN; Routing; Conditional Flooding

I. INTRODUCTION

Traditional Internet protocols malfunction in challenged networks such as interplanetary and deep-space networks in which communications are subject to delays and disruptions [1]. These challenged networks are generalized as Delay-Tolerant Networks (DTNs), so named by Kevin Fall in 2002 [2]. A DTN has characteristic properties: high and variable latency, low data rate, frequent disconnection, and high error rate. Node mobility contributes greatly to these characteristics. A *store-and-forward* mechanism is commonly used to transfer data in a DTN. A node can carry and store data in its own buffer, and forward these data to other connected nodes when they are available. The basic unit of data transmission in DTNs is called a *message* or *bundle*.

Many application domains using DTNs have been studied and extended to wireless networks such as Mobile Ad Hoc Networks (MANET), Vehicular Ad Hoc Networks (VANET), and Wireless Sensor Networks (WSN). In the VANET, communication between two vehicles is intermittent due to the changes in driving speed and direction. VANETs have all the properties of challenged networks, and can be viewed as vehicle-based DTNs.

The routing protocols designed for DTNs and VANETs have been individually studied in different contexts. For example, contact-based routing in DTNs makes use of human social interaction. Location-based routing is appropriate in VANETs for GPS-equipped vehicles. Here, we hope to design a new routing protocol suitable for a vehicle-based DTN by combining these different methodologies.

A stable end-to-end routing path is hard to maintain if connections are intermittent. Most DTN routing protocols rely on the concept of message replication and path diversity, which generates many message copies to be routed individually along different paths. This feature of routing diversity can greatly increase the message delivery ratio, but also will cause a negative effect if too many message copies are generated.

However, most VANET routing protocols do not consider the routing diversity based on message replication but make good use of vehicle locations to increase routing accuracy. The weakness of this kind of routing strategy is the extra overhead required to set up a location server from which the current location of a vehicle can be obtained.

In this paper, we propose a flexible and efficient routing protocol which achieves both routing diversity and routing accuracy. By developing a new location discovery method, we eliminate any need for location servers, but still realize location-based routing operations. Moreover, we use certain criteria to restrict the scope of message replication.

The rest of this paper is organized as follows. Section II gives a brief survey of DTN routing protocols. The proposed routing is presented in Section III. Performance comparisons are presented in Section VI. Finally, we make some concluding remarks in Section V.

II. RELATED WORK

Most research efforts in DTN routing focus on two things: buffer management and routing decisions. One major task involved in buffer management is the arrangement of message transmission order and message drop order. This task can be accomplished by sorting messages in a buffer according to certain information indexes such as message received time, message remaining life time, and message delivery hop.

The routing decision aims to determine how to replicate and forward message copies to suitable nodes that are evaluated with regard to certain metrics such as contact frequency and contact waiting time. DTN routing protocols can be roughly classified into three categories according to how many copies of the same message are created: *forwarding*, *quota-replication*, and *flooding*. In a forwarding scheme, a single-copy message is forwarded through successive intermediate nodes to the destination. A quota-replication scheme creates a limited number of message copies as specified by a quota. A flooding scheme creates an extremely large number of message copies and spreads them into the whole network.

This research was partially supported by National Science Council of the Republic of China under Contract No. NSC 100-2221-E-259-020.

Although forwarding protocols [3][4] save buffer space in each node, they present a low delivery ratio unless contact events are frequent in the network. Flooding protocols such as Epidemic [5] and MaxProp [6] create as many message copies as possible, to increase the opportunity of a message reaching its destination. However, such a large volume of message copies exhausts the buffer space of each node and causes the dropping of some message copies. Quota-replication protocols such as Spray&Wait [7] and EBR [8] limit the maximum number of copies of the same message by setting an initial quota. The setting of the quota is a tradeoff between resource consumption and message deliverability, and hence is a challenge.

In a vehicle-based DTN, regular contact behavior between two vehicles rarely exists. Those routing protocols that heavily relay on information derived from contact history fail in such an environment. Fortunately, useful information such as location and motion vector can be acquired from a GPS (Global Positioning System) device in a vehicle. A typical VANET routing scheme, like GPSR [9], shows significant improvement in terms of reducing end-to-end delay if a message is always forwarded by a node which is closer to the destination. However, this kind of forwarding routing has poor performance in terms of delivery ratio, as evaluated in our previous paper [10].

Some routing protocols [11][12][13][14] that consider both location information and message replication have also been proposed. In DAER [11], one node always copies a message to all contact nodes that are closer to the message destination if this node is moving toward the destination. Otherwise, this node forwards the message to a contact node by removing the message from its buffer. Conceptually, DAER is a hybrid approach, combining forwarding and flooding. In VR [12], one node when driving on a horizontal road would copy messages to those contact nodes driving on vertical roads with high probability. However, VR performs worse than DAER. POR [13] considers the neighborhood of a node. In a wireless communication environment, a node may contact several nodes simultaneously within its communication range. These nodes become neighbors of this node. In POR, one node copies messages to one of its neighbors, but not to all contact nodes. ORWAR [14] behaves like Spray&Wait but additionally predicts the connection time between two contact nodes to avoid any incomplete message transmission.

All the above location-based routing protocols require the exchange of some motion-related data between two contact nodes. Furthermore, the locations of destinations of all messages need to be known in advance in GPSR, DAER, and POR, which makes these protocols impractical.

III. The Proposed Routing

Here, we introduce our proposed routing, which is called Distance-Aware Routing with Copy Control (DARCC). DARCC performs conditional flooding that copies messages to other nodes based on certain criteria, and additionally performs location-based routing if the location of a message destination can be known. Instead of building a centralized location server, we provide a distributed method to disseminate location information. Due to the nature of distributed information dissemination, one node may or may not know the current location of a message destination at any given time. Moreover, the location information acquired may or may not be up-to-date.

In our routing, we make the following assumptions. Each node (or vehicle) is equipped with a GPS device and has a globally unique identification (ID). Each message generated by any nodes is uniquely identified by a message ID which is the combination form of the source node ID, the destination node ID, and a sequence number given by the source node. Each node will also periodically broadcast a beacon message for announcing its appearance to its neighbors.

A. Routing Principles

To increase message spreading, we adopt flooding routing, but conditionally control message replication. When one node would like to copy a message to another node, we use the neighborhood concept to properly select a certain number of neighbors as message relay nodes. There are different selection criteria, depending on whether or not the node knows the location of the message destination, and whether or not the node is close to a road junction.

If the location of the message destination is clear, we prefer selecting relay nodes along the direct path to the destination, and hence the following Principle 1 is used. Otherwise, we prefer spreading the message in different directions by using routing diversity. Moreover, more relay nodes are selected in a junction area, and hence the following Principle 2 is used in such cases.

Principle 1: *Select a single relay node that is closer to the destination from the neighbors of the current message holding node.*

Principle 2: *Select a single relay node that has the most difference in driving direction with the current message holding node from the neighbors, and select two such nodes if the current message holding node is close to a road junction.*

A simple junction detection algorithm is applied here by assuming that a node is close to a junction if the moving direction of one of its neighbors within 50 m is different from that of this node, with an angle larger than 45°.

Assume that a node has a quota k (one or two in our routing) to copy the same message to k relay nodes. If all these k relay nodes are still located in the communication range of this node, this node should not copy the same message to any other newly encountered neighbors. This naturally avoids redundant message replication when several nodes holding the same message copy are located nearby.

B. Routing Information

In our routing operation, each node locally maintains some data, and exchanges part of the data with a newly encountered node. A node i will keep the following data:

- POS_i: current position.
- MOV_i: current motion vector (speed and direction).
- TS_i: timestamp when getting POS_i and MOV_i.
- $QUEUE_i$: set of messages locally buffered.
- M-$LIST_i$: list of IDs of those messages in $QUEUE_i$.
- $LIST_i$: list of IDs and timestamps of those messages that have reached their destinations.
- D-$LIST_i$: list of motion-related records of message destination nodes.

- $NTABLE_i$: neighbor information table.

M-LIST is used to avoid transmitting redundant messages to a node. I-LIST is used to clean other message copies from the buffer when one message copy has reached the destination. When a destination node successfully receives a message, this node adds a new record with a timestamp for this message to its I-LIST. Two contact nodes will exchange and merge their I-LISTs. The timestamp is used to remove an old and useless record in I-LIST by setting an expired time.

D-LIST is the base of our distributed location service and contains records in this form $\{i, POS_i, MOV_i, TS_i\}$. A node will declare itself as a destination node if this node receives a message that is destined for it. Moreover, the role of a destination node is removed when this node does not receive any destined messages after a certain time period. A destination node has the responsibility to announce a motion record including node ID, current POS and MOV values, and the current timestamp to any encountered nodes. All non-destination nodes will collect these motion records into a D-LIST. Moreover, two contact nodes will exchange and merge their D-LISTs. In the merging, an old motion record is replaced by a new one by referring to the timestamp, and is even deleted by means of setting an expired time. After successive contact events between nodes, the location information of a destination node is gradually spread into the whole network. Based on this motion record, a node can easily estimate the current position of a destination node i by computing this value: POS_i + (current time – TS_i)×MOV_i.

NTABLE records data about all the current neighbors of one node. For a neighboring node i, we keep a record in this form $\{i, POS_i, MOV_i, TS_i, \text{M-LIST}_i\}$ and insert it into NTABLE. A joining or leaving neighbor is detected by listening to the beacon message.

C. Routing Algorithm

DARCC contains two basic procedures: contact and routing. The contact procedure is performed when two nodes encounter each other for information exchange. The operations in the procedure are illustrated from the viewpoint of one side, but they are actually performed by both sides. The routing procedure is executed for routing messages in the buffer. The following notations are used in our algorithm:

- DST(m): destination node of message m.
- MC(m): number of neighboring nodes that hold message m.
- DIS(a, b): Euclidean distance between nodes a and b.

In the routing procedure, we propose a new sorting policy to manage the buffer. Suppose that each message in the buffer is annotated with a hop count, which is the number of hops taken from the source for the message. Then we put those messages whose destination locations can be found in the D-LIST into the first part of the buffer, and put the remaining messages into the second part. The first part is placed before the second one in the buffer. Messages in the first part are sorted in ascending order of distances to their destinations, and messages in the second part are sorted in ascending order of hop counts.

The transmission order is from the head to the tail of the buffer. That is, we give low transmission priority to a message that is far away from its destination or has travelled the network via many hops. Moreover, when the buffer cannot accommodate a newly received message, the message at the tail of the buffer is dropped first.

Contact Procedure
When node i contacts node j,
    Refresh M-LIST_i from $QUEUE_i$.
    If i has declared itself as a destination node,
        Update a new motion record of i into D-LIST_i.
    End if
    Transmit POS_i, MOV_i, I-LIST_i, M-LIST_i, and D-LIST_i to j.
    If i and j have successfully exchanged the above data,
        Merge two I-LISTs into a new I-LIST_i.
        Merge two D-LISTs into a new D-LIST_i.
        Record the information of j into $NTABLE_i$
        Remove records of those messages that are indicated in I-LIST_i from $QUEUE_i$.
    End if

Routing Procedure
While (True),
    If ($QUEUE_i$ and $NTABLE_i$ are not empty),
        For each message m in $QUEUE_i$,
            If DST(m) is in $NTABLE_i$,
                Copy m to DST(m) directly.
                Remove m from $QUEUE_i$.
            End if
        End for
        Sort messages in $QUEUE_i$,
        For each message m in $QUEUE_i$,
            Compute MC(m) from $NTABLE_i$.
            If (DST(m) can be found in D-LIST_i and MC(m)<1),
                Let k be the closest node to DST(m) among the nodes without holding m in $NTABLE_i$.
                If DIS(DST(m), k) < DIS(DST(m), i),
                    Copy m to node k.
                    Add m into M-LIST_k in $NTABLE_i$.
                End if
            Else if (i is close to a junction and MC(m)<2) or (MC(m)<1)
                Let k be the node with the largest variation in moving direction with i among the nodes without holding m in $NTABLE_i$.
                Copy m to node k.
                Add m into M-LIST_k in $NTABLE_i$
            End if
        End for
    Else,
        Wait for a timer.
    End if
End while

IV. Performance Evaluation

We evaluate routing performance using the Opportunistic Network Environment (ONE) simulator [15]. A city street map of size 3000 m × 3000 m is imported from the TIGER database [16]. Each road has two forward lanes and two reverse lanes. One hundred vehicles are generated using VanetMobiSim [17]. The average moving speed of each vehicle is 60 km/h.

The parameter settings of our experiments are listed in Table 1. Fixed-size messages are constantly generated per interval time during a generation period. Message source nodes are randomly selected, and 10 message destination nodes are specified.

Table 1. Parameter Settings.

Parameter	Value
Simulation time	3600 s
Number of nodes	100
Buffer size	3 MB
Transmission rate	250 kBps
Communication range	200 m
Message size	200 kB
Message interval	25 s, 20 s, 15 s, 10 s, 5 s
Message generation period	600 s to 3000 s

Three cost metrics are used in our evaluation, as below:

- *Delivery ratio*: ratio of the total number of messages received at destinations to the total number of messages sent from sources.
- *Average relay cost*: ratio of the total message relay count to the total number of messages received at destinations. The relay count is increased by one as any message is forwarded or copied from one node to another node in the network.
- *Effective latency*: ratio of the average end-to-end delay for a message from the source to the destination to the delivery ratio.

The effective latency above is different from the conventional end-to-end delay, and can clearly justify a protocol with short end-to-end-delay but with low delivery ratio.

In the first set of experiments, 10 message destination nodes are randomly selected from all vehicles. We compare DARCC with Epidemic, MaxProp, ORWAR (initial quota is 20), and Spray&Wait (initial quota is 6). The experimental results are shown in Figures 1~3. DARCC presents the highest delivery ratio. Flooding routing (Epidemic and MaxProp) suffers from significant performance degradation when the message interval is short, because heavy traffic load causes message dropping in limited buffers.

The number of message copies significantly affects average relay cost. As can be seen, flooding routing has high relay cost, and quota-replication routing has low relay cost. DARCC conditionally generates message copies, so the relay cost is medium compared to other schemes.

DARCC presents low effective latency, because we greedily forward messages to nodes which are closer to destinations. The other protocols blindly forward messages, introducing the possibility of forwarding them along long routing paths.

In the second set of experiments, 10 message destination nodes are fixedly assigned by installing static nodes beside roads. This setting is for simulating data transmission to RSUs (Road-Side Units) in VANETs. Moreover, this setting helps for the easy testing of those protocols relying on a centralized location server. All testing protocols including DARCC, DAER, GPSR*, and POR are performed by assuming destination locations are known in advance. GPSR* is a modified version that performs the store-and-forward mechanism instead of the right hand rule for solving connection holes, and is more suitable for networks with sparse nodes.

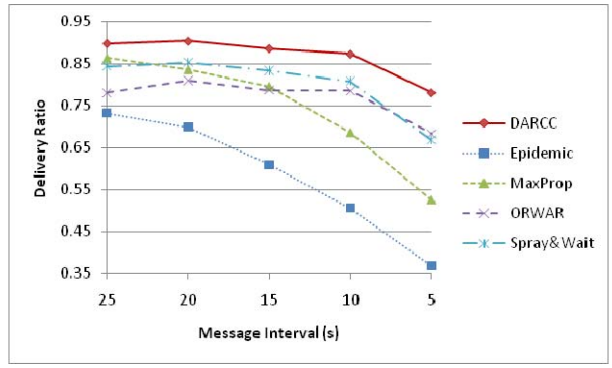

Figure 1. Delivery ratio with mobile destinations.

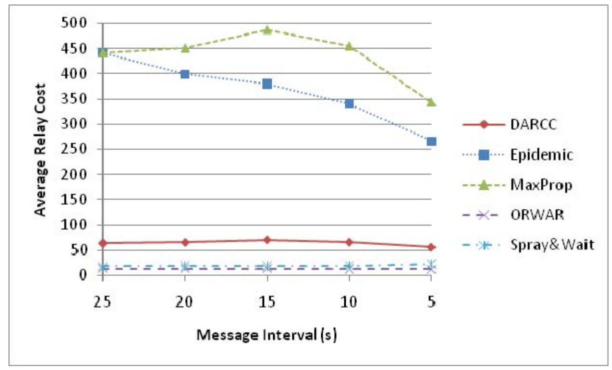

Figure 2. Average relay cost with mobile destinations.

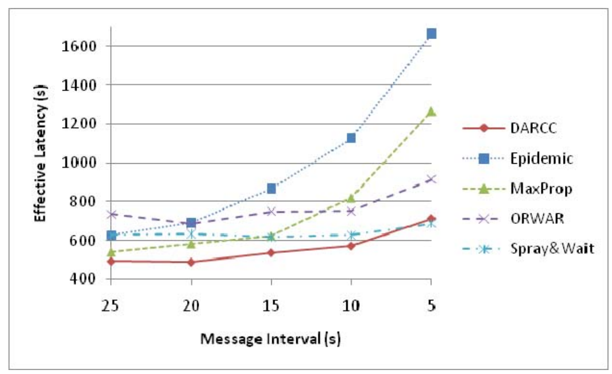

Figure 3. Effective latency with mobile destinations.

The experimental results are shown in Figures 4~6. DARCC again presents the highest delivery ratio among all the routing protocols. GPSR* performs worse due to the nature of forwarding routing, but still performs better than DAER and POR under the heavy traffic condition. DAER and POR suffer from serious message dropping and perform worse if the traffic load is high.

DARCC still incurs little relay cost due to effective control of message replication. GPSR* creates only one message copy for each generated message, so the relay cost is the lowest among all routing protocols.

Though all the testing protocols use the similar greedy approach, their performances in terms of delivery ratio are quite different. DARCC has the lowest effective latency, which is partially contributed to by its high delivery ratio. GPSR* relies less on the feature of routing diversity, so the discovered routing path may be long.

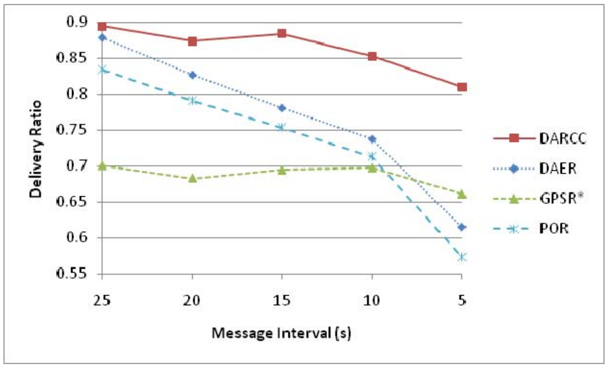

Figure 4. Delivery ratio with static destinations.

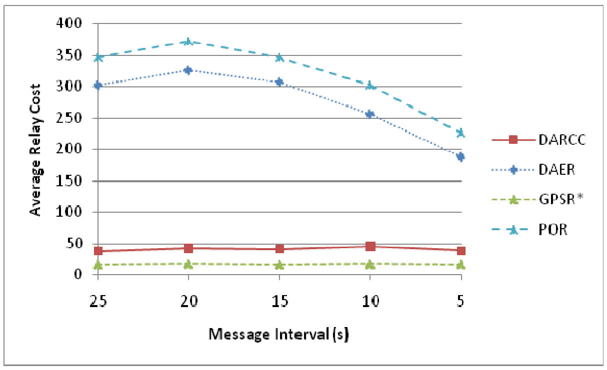

Figure 5. Average relay cost with static destinations.

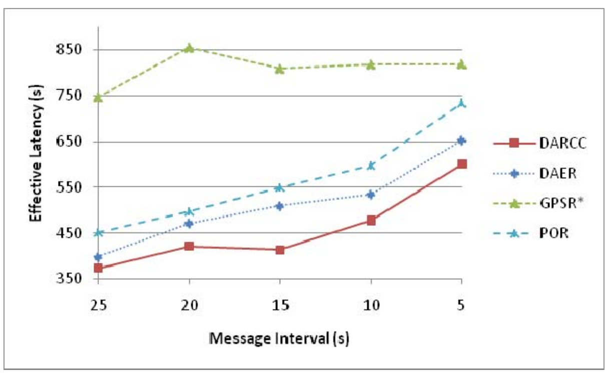

Figure 6. Effective latency with static destinations.

V. CONCLUSIONS

There are many emerging applications for VANETs based on considerations of safety and comfort. Efficient data dissemination is the basic requirement for the success of these applications. Traditional VANET routing considers motion-related information about a vehicle to increase routing accuracy, but still has a low delivery ratio. In this paper, we combine the DTN routing mechanisms of message replication and store-and-forward into VANET routing.

Our proposed routing is based on both routing diversity and routing accuracy, and achieves excellent performance in both delivery ratio and effective latency. We use the contact behavior among vehicles to gradually disseminate destination locations into the whole network. This method alleviates the need to use a centralized location server, which makes our routing more practical than other routing schemes.

In the future, we would consider a method to aggregate or compress the exchanged data in the network to reduce transmission overhead. An extension of our work by assuming that only parts of vehicles are equipped with GPS devices will be studied.

REFERENCES

[1] A. McMahon and S. Farrell, "Delay- and Disruption-Tolerant Networking," *IEEE Internet Computing*, vol. 13, no. 6, pp. 82-87, Nov. 2009.

[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," *in Proc. ACM SIGCOMM Conf.*, pp. 27-34, Aug. 2003.

[3] E. P. C. Jones, L. Li, and P. A. S. Ward, "Practical Routing in Delay-Tolerant Networks," *IEEE Trans. Mobile Computing*, vol. 6, no. 8, pp. 943-959, Aug. 2007.

[4] E. M. Daly and M. Haahr, "Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs," *in Proc. ACM MobiHoc Conf.*, pp. 32-40, Sept. 2007.

[5] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Duke University, Technical Report CS-200006, Apr. 2000.

[6] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," *in Proc. IEEE INFOCOM Conf.*, pp. 1-11, Apr. 2006.

[7] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," *in Proc. ACM SIGCOMM Workshop*, pp. 252-259, Aug. 2005.

[8] S. C. Nelson, M. Bakht, and R. Kravets, "Encounter-Based Routing in DTNs," *in Proc. IEEE INFOCOM Conf.*, pp. 846-854, Apr. 2009.

[9] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," *in Proc. MobiCom*, pp. 243–254, 2000.

[10] S. C. Lo, M. H. Chiang, J. H. Liou, and J. S. Gao, "Routing and Buffering Strategies in Delay-Tolerant Networks: Survey and Evaluation," *in Proc*. IEEE ICPP Workshop, Sept. 2011.

[11] H. Y. Huang, P. E. Luo, M. Li, D. Li, X. Li, W. Shu, and M. Y. Wu, "Performance Evaluation of SUVnet with Real-Time Traffic Data," *IEEE Trans. Vehicular Technology*, vol. 56, no. 6, pp. 3381-3396, Nov. 2007.

[12] H. Kang and D. Kim, "Vector Routing for Delay Tolerant Networks," *in Proc. IEEE VTC Conf.*, pp. 1-5, Sept. 2008.

[13] X. Li; W. Shu, M. Li, H. Y. Huang, and M. Y. Wu, "DTN Routing in Vehicular Sensor Networks," *in Proc. IEEE Global Telecommunications Conf.*, pp.1-5, Nov. 2008.

[14] G. Sandulescu and S. Nadjm-Tehrani, "Adding Redundancy to Replication in Window-Aware Delay-Tolerant Routing," *Journal of Communications*, vol. 5, no. 2, pp. 117-129, Feb. 2010.

[15] A. Keränen, J. Ott and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," *in Proc. SIMUTools*, Mar. 2009.

[16] Topologically Integrated Geographic Encoding and Referencing system, http://www.census.gov/geo/www/tiger/

[17] J. Härri, M. Fiore, F. Fethi and C. Bonnet, "VanetMobiSim: generating realistic mobility patterns for VANETs," *in Proc. ACM Intl. Workshop on Vehicular Ad Hoc Networks*, Sep. 2006.