TREBOL: Tree-Based Routing and Address Autoconfiguration for Vehicle-to-Internet Communications

Marco Gramaglia, Maria Calderon, Carlos J. Bernardos

***Abstract*—Efficient vehicle-to-Internet routing and address autoconfiguration are two of the missing pieces required to provide Internet connectivity from vehicles. Here, we propose TREBOL, a tree-based and configurable protocol which benefits from the inherent tree-shaped nature of vehicle to Internet traffic to reduce the signaling overhead while dealing efficiently with the vehicular dynamics. The paper describes the design and rationale of the solution, and presents the results of an experimental validation and performance evaluation, based on extensive simulations and real vehicular traces obtained in the region of Madrid.**

I. Introduction

Bringing IP connectivity to cars will enable classical and new Internet applications to be provided in vehicles. This will additionally help to speed up the adoption of vehicular communication systems by the users, since they will see an additional benefit in the installation of communication systems in their cars.

Current research efforts are focused on designing an architecture that, using an ad-hoc, short-range wireless and multi-hop paradigm, will be capable of connecting each vehicle inside the VANET to fixed roadside gateways placed along the roads, and from there to the Internet. Compared to other wireless communication approaches, using a multi hop solution brings benefits to the user (i.e., cost savings and high bandwidth), and to the network providers that can alleviate their already overloaded 3G infrastructure. In real deployments, these roadside gateways can be co-located with the Road Side Units (RSUs) deployed around the roads for safety purposes. To enable Vehicle-to-Internet communications, some functionalities are needed:

- *Address configuration:* Vehicles have to be able to autoconfigure a valid IP network address in an automatic way, without requiring manual intervention from the user.
- *Routing capability:* mechanisms for an efficient routing of IP datagrams, mainly unicast, from the vehicle to roadside gateways and vice versa.
- *Mobility management:* vehicular networks are characterized by high mobility. Thus, an effective mechanism for seamless handover between different networks and roadside gateways is required.

Standardization bodies working on vehicular networks (e.g., ETSI TC ITS, ISO TC204, IEEE 1609) are mainly focused on safety services and traffic efficiency, while Internet communications are considered to be of much lower priority. As an illustration of this, in the IEEE 802.11p, an amendment to the 802.11 standard especially designed for vehicular environments, Internet traffic gets a lower priority compared to safety and control messages. On the other hand, the coexistence of safety and Internet applications on the same communication box raises security issues (e.g., security attacks from malicious third party applications). Thus, we argue that in the future, cars will have two isolated communication boxes, a first one devoted to safety applications and conceived as one of the multiple safety devices inside the car (i.e., ABS or seat belts) and a second box that will use standard 802.11 wireless cards to provide Internet access to all the devices inside the vehicle (e.g., onboard embedded devices or user terminals such as laptops, smart phones or PDAs).

In this paper we focus on two of the previously mentioned functionalities, namely routing and address configuration. We propose (Section III) a new tree-based routing protocol (TREBOL) that can be used both in urban scenarios, i.e., where roadside gateways are deployed densely to provide good Internet access service in highly populated areas; and in highway scenarios, i.e., where roadside gateways are deployed sparsely due to cost reasons (Section IV). By slightly modifying existing IPv6 Stateless Address Autoconfiguration (SLAAC) [1] mechanisms, TREBOL may be used to also provide IP address autoconfiguration (Section III-A). The performance of TREBOL has been evaluated using real vehicular traces, including a comparison with other approaches (Section V).

II. Related work and motivation

Vehicular networks exhibit unique properties such as the high dynamics of the nodes (e.g., the link lifetime is subject to vehicles' movements). These particularities make the use of standard MANET routing protocols (either proactive or reactive) not suitable for this kind of environment. The knowledge of participant nodes' position information, typically supplied by a GPS receiver, can be exploited by VANET-specific routing solutions to increase their performance [2]. In particular, the need for tailored routing protocols to VANETs has led to two main families of position-based algorithms [3] according to the type of information they handle: Basic geographic protocols and Information-enriched geographic protocols.

Marco Gramaglia is with Institute IMDEA Networks, Avda. del Mar Mediterraneo 22, 28918 Madrid – SPAIN and Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés – SPAIN (email: marco.gramaglia@imdea.org)

Maria Calderon and Carlos J. Bernardos are with Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés – SPAIN (email: {maria, cjbc}@it.uc3m.es)

In basic geographic protocols [4] [5], an intermediate node forwards a packet to the direct neighbor which is the *closest* to the geographic position of the destination, operation known as greedy forwarding. So, each node has to be aware of *i)* the position of its direct neighbors, and *ii)* the position of the final destination. To this end, nodes send periodic beacon messages informing neighboring nodes about their identifier, position and other relevant information. However, the selection of a proper beaconing interval becomes really important to find a good trade-off between control overhead and up-to-date neighborhood information. The lower interval, the more up-to-date information is acquired, but at the cost of extra control overhead, interferences and more frequent wireless collisions. As for the position of the final destination, this information is provided by a *location service*. This functionality may be centralized (i.e., nodes update their new locations on a location server) or distributed (e.g., the source node floods a message asking for the position of the destination node), but in any case, the location service is another source of control overhead.

Information-enriched geographic protocols [6], [7], [8] base their operation on the existence of additional (i.e., besides position) information specific to VANET scenarios such as maps, statistics about traffic density on different roads, number of lanes per road, speed limits or information about trajectory estimations. These protocols, instead of following a greedy approach (e.g., choosing as next hop the *closest* neighbor to the destination), can take wiser forwarding decisions (e.g., choosing as next hop the *best* neighbor). At first glance one would expect that the more information is available the better the routing protocol performance is. However, the performance of these protocols depends very much on how accurate this additional information is, since the forwarding decisions that are taken might be erroneous or really far from optimal. Taking into account that the information they are dealing with is, in many cases highly dynamic (e.g., speed or density of cars), there is a non negligible probability that this information is stale or outdated when it is considered for forwarding. On the other hand, keeping this information updated may be costly in terms of control overhead.

Position-based protocols can support information exchange with the Internet, considering that roadside gateways are just other nodes that participate in the routing protocol, but at the cost of a significant control overhead. However, we argue that vehicle-to-Internet unicast communications exhibit a common set of characteristics that may be exploited by the VANET routing protocol. In particular, not all network nodes behave in the same way: roadside gateways (RSG) play a critical role, since they operate as relays to the Internet. The required network connectivity graph is anchored at the RSG (i.e., all data traffic traverses the RSG), as opposed to other vehicle scenarios, in which a mesh graph is desired. In this paper we propose TREBOL, a tree-based routing protocol flexible enough to quickly react to topology changes, which aims at enabling unicast vehicle-to-Internet communications. Besides, forwarding in TREBOL is not based on positions, so neither beacon messages nor location service information is needed, allowing great savings in terms of control overhead.

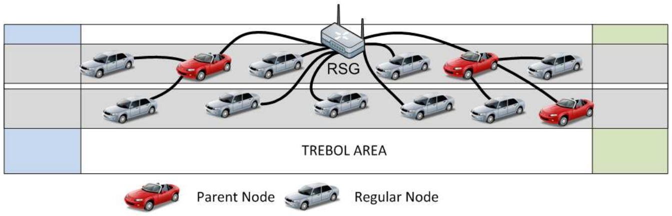

Figure 1. TREBOL area

III. TREBOL

In TREBOL, data forwarding decisions are based on IPv6 addresses (i.e., it is a topological routing protocol). Data paths follow a tree built by the TREBOL protocol, which is formed using position information (e.g., vehicles are assumed to have a GPS receiver) to minimize the control overhead load. We describe next how this is achieved. We assume for the time being that nodes are already provided with IPv6 addresses that can be used by the routing protocol (we describe how TREBOL can function also as address autoconfiguration protocol in Section III-A).

The main issue is how to build and update the tree in order to tackle the frequent topology changes in VANETs. The upstream tree (i.e., the tree used in the forwarding of data packets from the vehicle to the Internet) is built and updated when each node learns about its parent upon receiving periodical configuration messages (CM) sent by the roadside gateway (RSG). It is assumed that each RSG plays the role of relay (i.e., forwarding traffic from/to the Internet) for the vehicles within a limited geographical area, known as TREBOL area (see Figure 1). Thus, configuration messages sent by a RSG are spread within its TREBOL area. On the other hand, the creation of the downstream tree (i.e., the tree used in the forwarding of data packets from the Internet to the vehicle) follows a reactive approach: each node learns who are its children on a per data packet basis, as part of the forwarding of data packets.

As already mentioned, TREBOL builds and refreshes the upstream tree by using periodical configuration messages (identified by a unique and incremental *sequence number*) which are initially sent by the RSG and then regenerated and sent by a subset of the VANET nodes. Once a node receives a CM with a newer sequence number, the sender of that CM becomes the parent of the receiving node, and the forwarding state is updated accordingly (i.e., the parent is used as next hop for upstream data traffic towards the Internet). Then, the node regenerates the CM (i.e., updating some fields but keeping the original sequence number) and sets a backoff timer. Only if this backoff timer expires, the node broadcasts this regenerated CM to its neighbors. In the meantime, if the node receives another CM with this same sequence number (i.e., sent by another node with a shorter backoff time), it cancels the sending of the regenerated CM. It is worth mentioning that only if a node sends a regenerated CM, it has the chance to become a *parent* node. *Parent* nodes take the responsibility of forwarding data traffic from/to the Internet from/to its descendants, so a critical issue in TREBOL is to select as parents those nodes that according to their characteristics (e.g., speed, position, etc.) lead to more stable trees. The CMs sent by the RSG include the following information:

- $areaBoundary$: geographic information describing the TREBOL area. Nodes outside this area receiving a CM discard the message.
- $sendPos$: geographic position of the sender of the CM. It is set initially to the location of the RSG and then overwritten with the position of the last node that regenerated and sent the CM.
- $prefR$: value that represents the preferred distance between consecutive *parents* (i.e., nodes with children). Lower values imply more dense, populated trees, while higher ones imply sparse trees.
- R: value fixing the maximum allowed distance between the receiver and the sender (i.e. the RSG or a potential parent node) of the CM. If the sender is farther away from the receiver node than R (i.e., $sendPos$ field), then the CM is discarded. In this way, R serves as a virtual wireless coverage radius.
- $prefS$: value that represents the preferred speed of nodes sending regenerated CMs (i.e., potential *parent* nodes). It is set by the RSG. This value is used to preserve the stability of the tree selecting as *parent* nodes those that travel at similar speeds (closer to $prefS$).
- $maxSpeedDiff$: nodes whose speed differs more than this value from $prefS$ will be prevented from sending regenerated CMs (i.e., becoming *parent* nodes).
- D_{pos} and D_{speed}: these two values set the maximum value for the backoff timer. The higher these values are, the more time is required to build the tree. On the other hand, too short values might cause many wireless collisions.

Selecting the potential *parent* nodes is a completely distributed process based on a backoff timer:

$$T_{backoff} = \frac{\|((\|pos - sendPos\|) - prefR)\|}{R} \times D_{pos} + \frac{\|speed - prefS\|}{maxSpeedDiff} \times D_{speed}$$

where pos is the node position and $speed$ is the node speed. A node that is located at a distance $prefR$ from the sender of the CM, and that travels at a speed of $PrefSpeed$ would immediately send the regenerated CM ($T_{backoff} = 0\ s$). After waiting $T_{backoff}$ seconds, the node sends the regenerated CM (updates the $sendPos$ field) only if it has not received another CM with the same sequence number from one of its neighbors before. In this way, the shorter the $T_{backoff}$ of a node is, the more likely the node sends a regenerated CM becoming a potential *parent* (i.e., assuming the responsibility of having children and forwarding their data traffic).

On the other hand, the TREBOL downstream tree (i.e., the tree followed to deliver data traffic from the Internet to the vehicle) is built and refreshed on a per data packet basis as part of the data packets forwarding process. A node will be aware of the identity (i.e., the IPv6 address) of its descendants (i.e., downstream nodes in the tree) when it receives data traffic addressed to the Internet from one of its children (i.e., the child has selected the node as next hop for traffic towards the Internet). Thus, upon receiving a data packet to the Internet, the node learns the identity of the descendant (i.e., the source address of the data packet) and updates the corresponding forwarding state information (i.e., the child which forwarded this data packet becomes the next hop for downstream data traffic towards the descendant).

A. Address Autoconfiguration Support

CM messages are received by all the nodes within a TREBOL area. So far we have assumed that VANET nodes are already provided with an IP address that can then be used by the TREBOL routing mechanism as identifier in the forwarding process. The same CM messages could also be used to convey prefix information, allowing nodes to autoconfigure IP addresses in a way similar to the standard IPv6 SLAAC [1]. In fact, the most straightforward approach is to slightly modify the IPv6 SLAAC mechanism so it is integrated with TREBOL as follows. All nodes within the same TREBOL area share the same IPv6 prefix (or set of prefixes), effectively forming a multi-link subnet. The RSG sends standard Router Advertisements (RAs) messages, containing the prefix(es) allocated to the TREBOL area, and have the on-link flag (L) unset [9]. The two minor modifications that TREBOL introduces consist of: *i)* RAs are regenerated by each *parent* node, keeping the same prefix, and *ii)* RAs are used by all VANET nodes (including *parent* nodes, which are also routers) to autoconfigure an address from the prefix. These RAs are extended with additional options to carry the fields defined in the CMs (needed by TREBOL routing). In order to avoid unnecessary control overhead, Duplicate Address Detection (DAD) is disabled, since we can safely assume that in a vehicular environment there exist unique identifiers that can be used to generate IPv6 addresses.

The main advantage of using this autoconfiguration mechanisms is that it reduces the overall control overhead required by combining routing and address autoconfiguration functions using a single set of signaling messages. Note that this address autoconfiguration mechanism feature could be disabled if required, since TREBOL can also work with different IP address autoconfiguration solutions.

IV. Deployment considerations

From a deployment perspective, a goal is to configure TREBOL so it provides routing trees as much stable as possible, without imposing too high performance penalties. Out of the parameters that can be configured, R is determined by the chosen wireless technology, and $prefR$ can be expressed as a fraction of it. When selecting $prefR$, there is a tradeoff that needs to be considered: higher $prefR$ values lead to shorter, but less reliable/stable routes, as more nodes located at the border of the coverage would be selected as *parent* nodes. On the other hand, shorter $prefR$ values lead to more stable, but longer routes. $prefS$ can be fixed by taking the speed limit in the zone or by the RSG taking the average speed (by sampling the vehicles' speed in real time). $maxSpeedDiff$ can be expressed as a fraction of $prefS$.

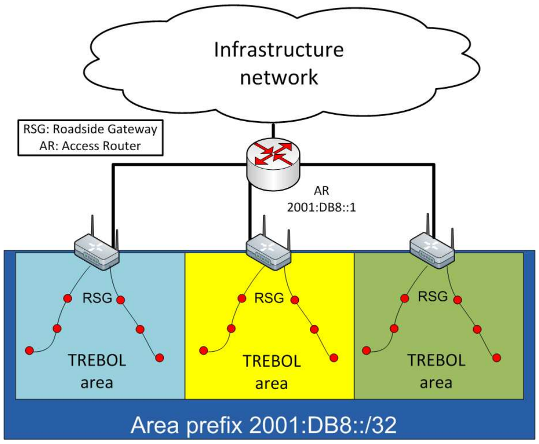

Figure 2. Example of TREBOL deployment hierarchy

There is a wide range of deployment scenarios where TREBOL might operate. These scenarios are mostly defined by the size of the TREBOL area, which is conditioned by different aspects, such as performance, vehicular density, cost considerations, etc. In large TREBOL areas (i.e. one single RSG provides service to a geographical area reasonably large), associating an IPv6 prefix to a TREBOL area does not introduce any issues. However, in small TREBOL areas, it is more convenient to associate the same IPv6 prefix to several adjacent TREBOL areas, avoiding the cost imposed by frequent IP address changes.

TREBOL easily supports a flexible association of IP prefixes to multiple TREBOL areas by introducing a simple hierarchy, with the possibility of having several RSG connected to a single Access Router (AR) on the infrastructure (see Figure IV). The AR plays the role of the *parent* of the RSGs (this is statically configured, without making use of the backoff timer).

V. Performance Evaluation

In order to evaluate the performance of the TREBOL routing protocol, we conduct simulations based on real vehicular traces. We compare the performance obtained with TREBOL with a pure geographic based routing protocol: the Greedy Perimeter Stateless Routing for Wireless Networks (GPSR) [4]. Additionally, we also compare some merit figures with the ones that would be obtained with an "optimal" (ideal) routing protocol, in which each node knows the best route to/from the Internet at any time. In this experimental evaluation we focus on the following performance metrics: the packet delivery ratio, the number of hops to reach the RSG and the control overhead.

A. Simulation environment

We run a set of trace-driven simulations with input data coming from real traffic measurements taken from one of the most important arterial road around Madrid (recorded on May, 10th 2010 from 8.30am to 9.30am), namely the orbital highway M-40. In this road, vehicles can span over three lanes (with an average speed of 90 km/h and a density of around 50 veh/km). Simulation settings are summarized in Table I.

Simulation framework	OMNeT++ and MIXIM
Wireless Device	802.11b @ 6Mb/s
Channel Model	Pathloss with channel fading
Coverage radius	225m
Distance between RSGs [m]	1000, 1500, 2000, 3000
Data traffic	ICMP Echo Request / Reply (packet size: 1KB)

Table I
Simulation settings

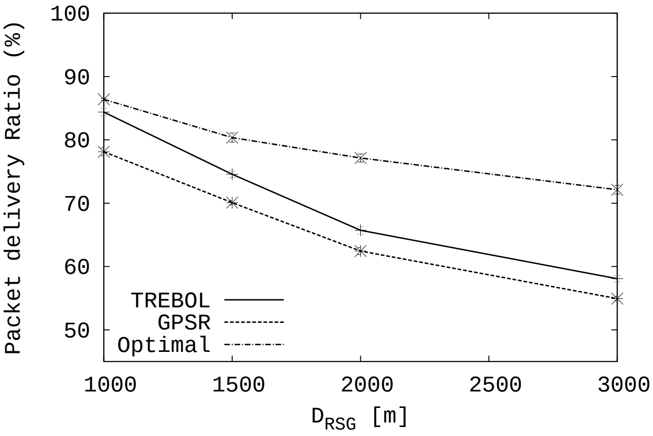

Figure 3. The packet delivery ratio with 95% confidence intervals

TREBOL and GPSR are configured to allow a fair comparison between them. For GPSR, the average time between beacons messages is set to 1 sec (uniformly distributed): $f_{GPSR} = 1msg/sec$. For TREBOL, the time between CMs is set to 1 sec as well (uniformly distributed): $f_{TREBOL} = 1msg/sec$. $prefR$ parameter is set to 180m and the $prefS$ is configured to be equal to the scenario's average speed.

B. Results analysis

Our goal is to compare TREBOL and GPSR in relation to the following three parameters: packet delivery ratio, average number of hops and signaling load.

Figure 3 shows the packet delivery ratio obtained in the simulations for TREBOL, GPSR and also the "optimal" value that could be achieved by an *ideal* routing protocol, under different deployment scenarios – characterized by the distance between RSGs (D_{RSG}). The "optimal" protocol always finds the best path if it exists, so when D_{RSG} increases the probability of having a gap in the multihop path is higher. As it can be observed, TREBOL provides a higher delivery ratio than GPSR.

The second metric we are interested in analyzing is the average length (i.e. number of hops) of the routes computed by TREBOL. Figure 4 shows the obtained results, including also GPSR and the "optimal" routing protocol. As expected, TREBOL uses slightly longer routes, as it tries to come up with routes composed of *parent* nodes that are separated by $prefR$ meters. The average route length achieved by GPSR is very close to the ideal one as GPSR tries to use the shorter possible route, by making use of the farthest forwarding node

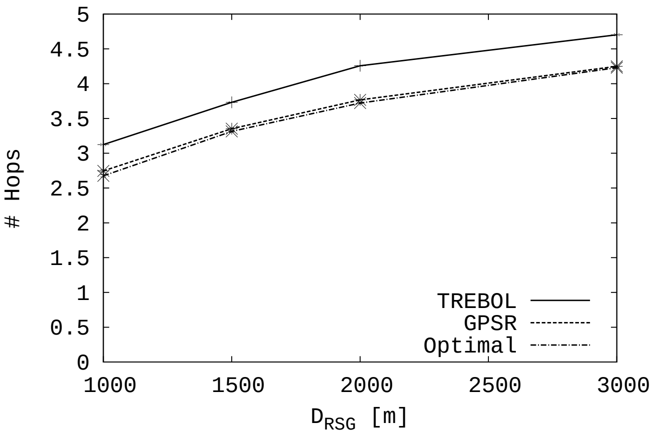

Figure 4. The average number of hops with 95% confidence intervals

D_{RSG} (m)	RRL
1000	6.68
1500	6.88
2000	7.03
3000	7.16

Table II
RELATIVE ROUTING LOAD OF GPSR COMPARED TO TREBOL

in the direction towards the destination. This however has an impact on the resulting packet delivery ratio, as the next-hop selected as best by GPSR might become unreachable due to nodes' movement, and this is not detected until the next beaconing period (i.e., until GPSR finds another best next-hop, data packets are lost).

The last important metric we analyze is the signaling load. We define the Relative Routing Load (RRL) as the ratio of the total number of control messages generated by routing protocol X within the routing domain in a given amount of time, compared with the number of messages generated by routing protocol Y:

$$RRL_Y^X = \frac{\text{\# signaling messages sent by X}}{\text{\# signaling messages sent by Y}}.$$

Since the signaling overhead is constant and independent of the data traffic generation rate for both GPSR and TREBOL, we can evaluate the advantage provided by TREBOL in terms of control messages savings by looking at RRL_{TREBOL}^{GPSR}:

$$\begin{aligned} RRL_{TREBOL}^{GPSR} &= \leq \frac{\beta D_{RSG} f_{GPSR}}{(D_{RSG}/prefR + 2) f_{TREBOL}} \qquad (1)\\ &\leq \beta prefR = \frac{prefR}{\gamma}, \end{aligned}$$

where β is the vehicular density and γ is the average inter-vehicular distance in the area (i.e. the distance between two consecutive vehicles). With TREBOL on average only one node every $prefR$ meters has to regenerate and send a CM, while with GPSR every node has to perform beaconing.

Using the average inter vehicular distance obtained from the traces ($\gamma = 18.55m$) the calculated RRL_{TREBOL}^{GPSR} is 9.70. We have also performed simulations, measuring RRL_{TREBOL}^{GPSR} (see Table II). The results are coherent with the analytical formulation in Eq. (1) as the calculated value is a limit superior. As observed, TREBOL provides an important signaling overhead saving due to the fact that in GPSR every node has to periodically send beacons – in order to keep its position updated into the other nodes' neighbor tables – while in TREBOL only *parent* nodes send signaling messages (and on average there is only one *parent* node every $prefR$ meters).

To sum up, the performance evaluation results show that TREBOL provides a better performance than GPSR – being this performance similar to the one achieved by the "optimal" one in terms of packet delivery ratio and average route length – while outperforming GPSR in terms of control overhead.

VI. CONCLUSION

TREBOL is a tree-based routing protocol that benefits from the inherent tree-shaped nature of vehicle-to-Internet traffic to reduce the signaling overhead while dealing efficiently with vehicular dynamics. Furthermore, the protocol could also be used to allow nodes to autoconfigure IPv6 addresses, reducing even more the overall control overhead required by routing and address autoconfiguration functions. Another remarkable feature of the proposed protocol is the wide range of deployment scenarios, mostly defined by the size of the TREBOL area, where it may operate, making it suitable for both urban and highways scenarios.

Results from simulations using real vehicular traces got in Madrid show that our proposal outperforms GPSR protocol, providing better traffic delivery ratio and allowing at the same time a significant saving of control overhead, aspect considered critical in wireless VANETs networks.

Future work includes analytical modeling of TREBOL performance and further performance optimizations.

REFERENCES

[1] S. Thomson, T. Narten, and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," RFC 4862 (Draft Standard), September 2007.

[2] T. Camp, J. Boleng, B. Williams, L. Wilcox, and W. Navidi, "Performance comparison of two location based routing protocols for ad hoc networks," 2002, pp. 1678–1687.

[3] F. J. Ros, V. Cabrera, J. A. Sanchez, J. A. Martinez, and P. M. Ruiz, *Vehicular Networks: Techniques, Standards and Applications Book.* CRC Press, 2009, ch. Routing in Vehicular Networks.

[4] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, 2000, pp. 243–254.

[5] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker, "Lazy cross-link removal for geographic routing," in *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems.* New York, NY, USA: ACM Press, 2006, pp. 112–124.

[6] I. Leontiadis and C. Mascolo, "GeOpps: Opportunistic geographical routing for vehicular networks," in *In Proceedings of the IEEE Workshop on Autonomic and Opportunistic Communications*, 2007.

[7] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," in *IEEE INFOCOM*, vol. 3, 1997, pp. 1405–1413.

[8] C. Lochert, H. Hartenstein, J. Tian, H. Fuessler, D. Hermann, and M. Mauve, "A routing strategy for vehicular ad hoc networks in city environments," in *In Proceedings of the IEEE Intelligent Vehicles Symposium*, 2003, pp. 156–161.

[9] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)," RFC 4861 (Draft Standard), September 2007.