Small-Scale and Large-Scale Routing in Vehicular Ad Hoc Networks

Wenjing Wang, *Member, IEEE*, Fei Xie, *Member, IEEE*, and Mainak Chatterjee

***Abstract*—A vehicular ad hoc network (VANET) is a highly mobile wireless ad hoc network that is targeted to support vehicular safety, traffic monitoring, and other applications. Mobility models used in traditional mobile ad hoc networks cannot directly be applied to VANETs since real-world factors such as road layouts and traffic regulations are not considered. In this paper, we propose a vehicular mobility model that reflects real-world vehicle movement and study the performance of packet-routing protocols. First, we study the routing in small-scale VANETs and propose two routing schemes: 1) connection-based restricted forwarding (CBRF) and 2) connectionless geographic forwarding (CLGF). With the insights obtained, we consider routing in large-scale VANETs. Since road complexity and traffic variety may cause many potential problems that existing routing protocols cannot address, we introduce a two-phase routing protocol (TOPO) that incorporates road map information. The proposed protocol defines an *overlay* graph with roads of high vehicular density and *access* graphs that are connected to the *overlay*. While in the *overlay*, packets are forwarded along a precalculated path. As far as *access* routing is concerned, we employ the aforementioned CBRF and CLGF schemes and send packets to the *overlay* or handle packets delivered from the *overlay*. We argue that the TOPO can serve as a framework that integrates existing VANET routing protocols. We also consider data diversity in VANETs and design the TOPO as an *intelligent transportation system (ITS)-friendly* protocol. To validate our design philosophy and the routing protocol, we use different areas in the city of Orlando, FL, and generate vehicular mobility traces, following our mobility models. We feed the traces to network simulators and study the routing behavior. Simulation results demonstrate the performance and effectiveness of the proposed routing protocols for large-scale VANET scenarios.**

***Index Terms*—Mobility, networks, routing, vehicular ad hoc network (VANET).**

I. Introduction

THE approval of the 75-MHz spectrum at 5.9 GHz for dedicated short-range communications (DSRC) [9] by the Federal Communications Commission and the successful deployments of WLAN technologies are making vehicular ad hoc networks (VANETs) a reality [5], [27], [36]. In recent years, the VANET has emerged as a research area that has received increased attention from the research community. However, due to the cost and difficulty associated with the implementation of VANETs in the real world, computer simulations remain as one of the primary techniques in investigating networking characteristics of VANETs. In this regard, it is very important to adopt realistic vehicular mobility models and design network protocols that are capable of delivering good end-to-end performance in such highly mobile environments.

It is widely accepted that the underlying mobility models greatly affect ad hoc network performance [3], [6], [41]. Many studies on mobile ad hoc networks (MANETs) have used random waypoint (RWP) or the Manhattan model to simulate node movements in an open field. Although such mobility models work well in certain scenarios, they are not suitable for VANETs, simply because the movement of vehicles is constrained by the layouts of the roads. Moreover, traffic regulations (e.g., speed limits and traffic lights) and driver behaviors (e.g., overtaking or following) make realistic vehicular mobility far different from the commonly used ones in MANETs.

The traffic simulator framework introduced by Saha and Johnson [31] makes it convenient to use the real map of the U.S. in the form of a graph model where the edges represent the roads and the vertices represent the intersections. The (map) input to the simulator is the data provided by the U.S. Census Bureaus' Topologically Integrated Geographic Encoding and Referencing system (TIGER) Project [34]. In their framework, each vehicle randomly chooses a source and a destination on the map and moves along the route, which is calculated using Dijkstra's single-source shortest path algorithm. The output of the framework is the trace of the vehicles on the map, the data format of which is compatible with the network simulator ns-2 [45]. However, the framework only uses simple mobility models (similar to RWP [21]) and does not include realistic vehicle mobility models.

Using the aforementioned simulator framework, it is possible to design a traffic simulator with a sound mobility model, which generates realistic vehicle mobility traces that can be used for network simulation.[1] Moreover, we are motivated to investigate the performance of existing routing protocols in VANETs and design new protocols. We study the routing problem in both small (e.g., less than 1 mi^2) and large areas (e.g., tens of the mile scale). In small-area routing, we propose and compare two routing schemes: 1) connection-based restricted forwarding (CBRF) and 2) connectionless geographic forwarding (CLGF). Considering different road topologies, we conduct extensive simulations to evaluate the performance of the proposed protocols, as well as other existing counterparts,

Manuscript received September 22, 2008; revised February 23, 2009 and April 23, 2009. First published June 19, 2009; current version published November 11, 2009. The review of this paper was coordinated by Prof. H. Hassanein.

The authors are with the School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816 USA (e-mail: wenjing@eecs.ucf.edu; xiefei@eecs.ucf.edu; mainak@eecs.ucf.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2009.2025652

[1] We make the traces compatible with both ns-2 and GTNetS [10].

in terms of the average data packet delay, data packet delivery ratio, and routing overhead.

However, when applied to a large-scale VANET, neither of the small-scale routing protocols maintains a fair performance. To this end, we use additional information such as traffic flows and road layouts to assist the routing. Thus, we propose a two-phase routing protocol (TOPO) for large-scale VANETs. The two phases are *access* and *overlay*. While *overlay* is a graph of high vehicular density roads, (e.g., state roads and highways), *access* is the rest of the areas/roads connecting to the *overlay*. The TOPO utilizes the road and traffic information on the *overlay* and delivers the message along the *overlay* to the *access* area of the destination, where the routing can again be handled in a small-scale area.

The advantages of the proposed routing protocol are twofold: First, it serves as an effective routing protocol in large-scale VANETs, as our simulation validation is conducted with a 30-km^2 real-world map. Second, the TOPO is a framework that can be integrated with existing VANET routing algorithms. Various routing protocols can reside in the two defined stages in the TOPO and independently perform with the TOPO as an interface.

The rest of this paper is structured as follows: In Section II, we discuss the related work on VANET mobility models and routing protocols. We propose the vehicular mobility model in Section III. In Section IV, small-scale VANET routing protocols are presented with simulation results and a discussion about performance. The routing for large-scale VANETs is presented in Section V. In Section VI, we further evaluate the performance of the TOPO and suggest enhancements to make the TOPO more suitable for intelligent transportation system (ITS) applications. Section VII concludes this paper.

II. Related Work

Mobility models in MANETs have extensively been studied, among which, the RWP is the most popular. The RWP is a tractable model for evaluating the worst-case performance; however, it does not provide much insight on realistic vehicular movements. Other mobility models in MANETs are not suitable for VANETs, because they consider neither macro (e.g., road information and traffic information) nor micro (e.g., driver behavior) features, which are crucial in VANETs.

As far as modeling mobility in VANETs is concerned, there are generally two kinds of approaches: 1) analysis and 2) simulation. Analytical modeling of mobility characterizes the traffic flow from a macroscopic perspective. Such methods are shown to accurately model how vehicles move on highways [39], [40]. However, in a more generic scenario, e.g., a residential area, analytically modeling mobility is much more challenging [25]. Although sophisticated mathematical tools can be applied, factors such as driver behaviors are difficult to capture. In terms of the simulation study, traffic engineers have designed different ways to characterize and simulate vehicular traffic, with a variety of traffic simulators being proposed. Wu *et al.* [38] used CORSIM to generate traffic traces. However, since CORSIM is not open source, it is not easy to pursue research on that. Another disadvantage of the proprietary traffic simulator is that the output data format may not directly be compatible with existing network simulation tools. Recorded real-world vehicular traces [13], [26] are also a way to model mobility, but the cost in gathering those data is considerably high. Saha and Johnson [31] developed an open-source traffic simulator framework with real-world U.S. maps. The work in [21] is based on the same framework, but, instead of using real maps, it deploys regular blocks. Other realistic traffic simulation efforts for VANETs are presented in [11], [22], and [29], whereas [17] shows thorough analysis how the mobility would affect the network connectivity.

As far as routing protocols for ad hoc networks are concerned, there are many. See [1], [2], [7], and [20] for a good survey of recent advances in this topic. Implementing these protocols on VANETs is a feasible approach in studying and evaluating routing issues. Again, the performance of routing protocols is highly dependent on the vehicular mobility model and environmental factors (e.g., traffic density and road layouts), and there is still very little understanding on the end-to-end routing performance when all physical, network, and radio characteristics are incorporated [26].

It is generally agreed that the routing protocols in VANETs should incorporate geographic information (i.e., road topology and GPS) [12]. Greedy perimeter coordinator routing (GPCR) [19] is the extension of greedy perimeter stateless routing (GPSR) in the VANET. It delivers messages along the road; however, it ignores the vehicle mobility and route maintenance. GVGrid [32] is a quality-of-service-based VANET outing protocol that exploits geographic information. It divides a geographical area into grids and forwards packets along the roads crossing different grids. However, it assumes that both the source and the destination are static, which does not always hold true in VANETs. Vehicle-assisted data delivery (VADD) [42] reflects a consent in VANET research that carry-and-forward schemes can be useful, because disconnection frequently happens; however, routing is not its focus. Zhao *et al.* [43] showed that the use of a data buffer at intersections can significantly improve the data delivery ratio. Other protocols based on link or traffic metrics are also proposed for VANETs, e.g., MUlti-hop Routing protocol for Urban VANET (MURU) [24] and improved greedy traffic aware routing protocol (GyTAR) [14]. Moreover, it is very much desirable that all the routing protocols be compatible with ITS services [33].

This research is motivated by the aforementioned work. Similar to Choffnes and Bustamante [8], we propose a vehicular mobility model. With the road data from the TIGER project [34], we developed a Java-based traffic simulator to generate a vehicular mobility trace. Different from the approach taken in [8] and [26], where the performance of well-known routing protocols, such as dynamic source routing (DSR) [15], ad hoc on-demand distance vector routing (AODV) [30], and GPSR [16], have been evaluated, we focus on how to modify the existing protocols to better fit a VANET. Furthermore, we propose a novel two-stage routing scheme for a large-scale VANET. Our scheme, i.e., the TOPO, which handles both static and mobile scenarios, is complementary to existing protocols, such as [32] and [42]. ITS-compatible features are also added to the TOPO so that it can better support ITS services. The

performance of the TOPO integrated with the roadside unit is also studied, with the results serving as a supportive extension to [43].

III. Vehicular Mobility Models

The most distinctive and basic characteristic for any vehicular mobility model is that the vehicles move on roads. In the TIGER data files, a *road* is defined by two *intersections* it has with other *roads*. By this definition, a geographic road in real life is divided into several segments of *roads*. We define and implement our mobility models based on *roads* and *intersections*.

A. Road Mobility Model

On any road, the mobility of vehicles is constrained by the posted speed limit. Generally, in the U.S., the speed limits are among 15 mi/h (24 km/h), 35 mi/h (56 km/h), 55 mi/h (88 km/h), 60 mi/h (96 km/h), and 75 mi/h (120 km/h). We model the speed of vehicles as per three rules.

1) If there is no other vehicle on the same road in the same direction for d m ahead, a vehicle can choose an arbitrary speed within ± 5 mi/h (8 km/h) of the speed limit.
2) If there is another vehicle on the same road in the same direction within d m ahead, a vehicle can decide whether to overtake or not. The overtake probability is p. To overtake, the vehicle will speed up 5 mi/h faster than the vehicle in front. It will maintain this speed until it is 50 m ahead of the overtaken vehicle.
3) If the vehicle decides not to overtake, it will go to the *following mode*. In the *following mode*, a vehicle keeps a safe distance (usually 2-s distance) at its speed away from the followed vehicle. The vehicle then keeps the same speed as the one ahead. We understand that there are many advanced car-following models [4]; however, through simulations, we find that car-following models do not have a major influence on the wireless network performance.

B. Intersection Mobility Model

When a vehicle approaches an intersection, it follows the intersection mobility rules given here.

1) If an intersection is within 50 m ahead of a vehicle, the vehicle will reduce its speed at a deceleration of a so that it can come to a complete stop at the intersection.
2) At an intersection, based on the type of the intersection, a vehicle may need to stop and wait for a certain period of time. The maximum values for waiting times are listed in Table I. If the intersection type is a stop sign, the vehicle is required to stop for 3 s before proceeding to another road. If the intersection type is a traffic light, the vehicle can pass it without waiting if the vehicle sees green; otherwise, it will stop and wait. The traffic light durations are also listed in Table I. Each row represents the speed limit of the road the vehicle is on; each column represents the maximum speed limit of the intersection roads. SS and TL stand for the "stop sign" and "traffic light," respectively. It is noted that this table is not symmetric, because there is more traffic on a road with a higher speed limit and thus should enjoy less waiting time.
3) When passing the intersection, the vehicle increases its speed at an acceleration of b. The mobility is then regulated by the road mobility model.

TABLE I
Maximum Stop Time at Intersections

	15mi/h	35mi/h	55mi/h	60mi/h	75mi/h
15mi/h	SS 3sec	SS 3sec	SS 3sec	SS 3sec	SS 3sec
35mi/h	SS 3sec	TL 30sec	TL 30sec	TL 30sec	TL 45sec
55mi/h	0	TL 10sec	TL 30sec	TL 30sec	TL 30sec
60mi/h	0	TL 10sec	TL 30sec	TL 30sec	TL 30sec
75mi/h[1]	0	0	0	0	0

[1] 75-mi/h roads are interstate highways; they are not controlled by traffic light or stop signs. The values in this table is developed from [8] and its original technical report.

Although constant acceleration and deceleration are hard to achieve in real life, with limited vehicle information data, we assume that the acceleration/deceleration rates do not change within the acceleration and deceleration periods.

IV. Connection-Based Restricted Forwarding and Connectionless Geographic Forwarding: Routing in Small-Scale VANETs

With the vehicular mobility model defined, we now proceed to study packet routing in small-scale VANETs. To do so, we first need to define road topologies, populate roads with vehicles, and make them go from some starting point to a destination.

A. Implementation

We choose two 1000-m $\times$ 1000-m square areas in Orlando, FL, as shown in Figs. 1 and 2. The online maps and TIGER line file figures are presented to show two very different road layouts: 1) a downtown "grid" area and 2) a residential area. We generate the trace for the vehicle movements for 1000 s.

We implement our mobility models using the framework in [31]. We add the traffic lights and stop signs to the corresponding intersection, and let the traffic lights operate according to Table I. The simulator begins with randomly chosen source and destination spots on the roads. Given the road layouts, the simulator will calculate the path from the source to the destination for the vehicle to take using Dijkstra's single-source shortest-path algorithm. With this path planning, vehicles know where to turn at intersections. The parameters are set as intervehicle distance $d = 50$, overtaking probability $p = 0.5$, and acceleration/deceleration $(a = -b) = 4$ m/s^2.

To validate our proposed mobility model, we also implement the car-following model Street RAndom Waypoint (STRAW) described in [8]. In Fig. 3, under the same coordinate, we plot the average speed of vehicles generated by STRAW for both the downtown and residential areas. It is observed that the average speed of vehicles decreases when the number of vehicles increases. In addition, both models generate similar values for average speeds. To make the experimental study

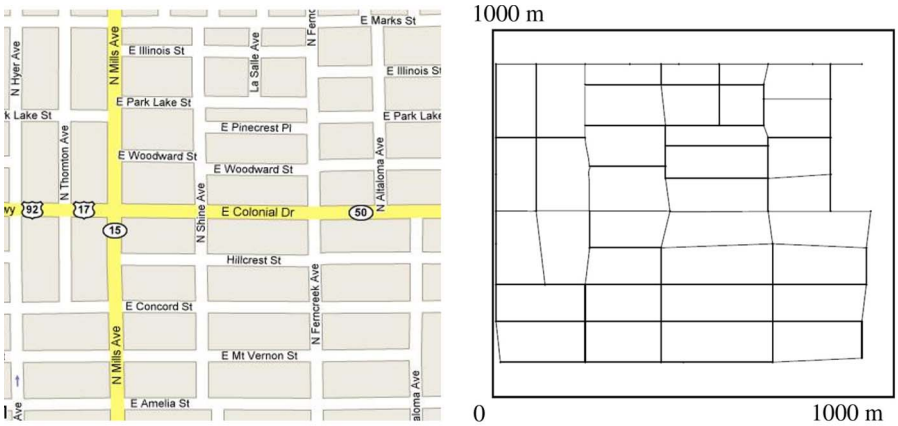


Fig. 1. Part of the Orlando, FL, downtown and the corresponding TIGER output. (a) Online map (courtesy of maps.google.com). (b) TIGER line file figure.

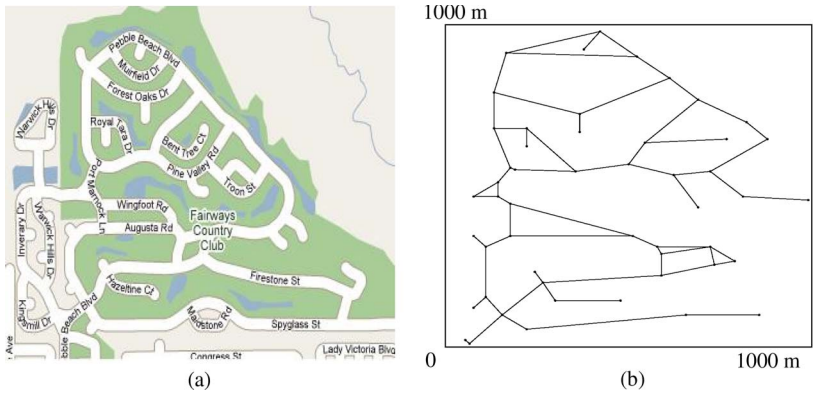


Fig. 2. Residential area in Orlando, FL, and the corresponding TIGER map. (a) Online map (courtesy of maps.google.com). (b) TIGER line file figure.

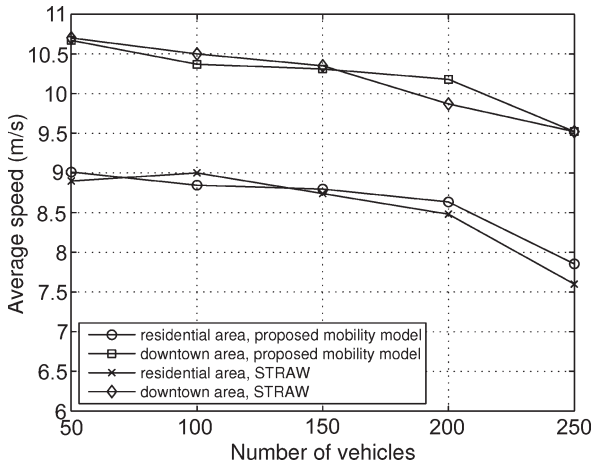


Fig. 3. Average speeds of vehicles in residential and downtown areas applying different mobility models.

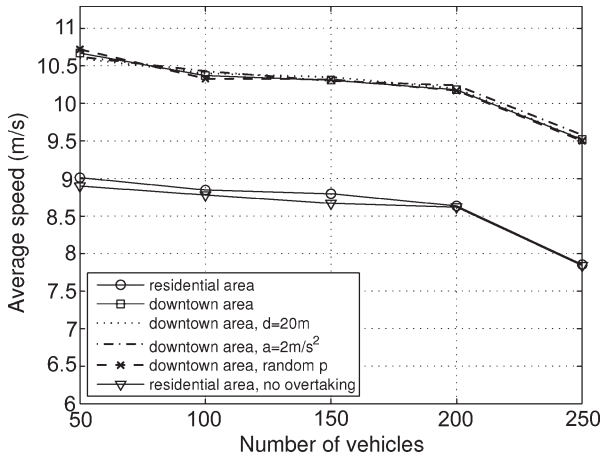


Fig. 4. Average speeds of vehicles in residential and downtown areas with different mobility parameters.

more comprehensive, we investigate how the values of the mobility parameters affect the average speed. In particular, we consider the parameters of intervehicle distance d, acceleration a, and overtaking probability p. Other than the base case with $d = 50$ m, $p = 0.5$, and $a = 4$ m/s^2, we conduct four other controlled tests. The first two tests study the effects of d and a, where $d = 20$ m, and $a = 2$ m/s^2. The last two tests consider the different tendencies of drivers, i.e., to overtake or follow. For the downtown area simulation, we assign each vehicle with a predefined p value in [0, 1], and this value is fixed in the simulation; in the residential area, no vehicle is allowed to overtake. The comparison results are shown in Fig. 4; however, the plots indicate that the mobility model parameters have a very limited effect on the vehicular average speed. Nonetheless,

the figures do suggest that the road layouts greatly affect the average speed, because most roads in both scenarios are of the same speed limit.

B. Inapplicability of MANET Routing Protocols

Proactive routing protocols require that all nodes maintain a routing table for routing to other nodes. To update the routing tables, a huge amount of information exchange is needed, making large overhead a burden for the network. In a high-mobility network such as a VANET, the overhead is even larger because of the increased routing table update failure probability. As their counterpart, reactive or on-demand routing protocols discover and establish routes only when there is a need. This kind of routing protocol saves unnecessary information exchange and thus brings down the overhead cost. Nevertheless, existing routing protocols in ad hoc networks are exposed to some performance problems in VANETs [23], [37]. Moreover, the performance of well-known AODV and DSR protocols in [8] motivates us to explore the design space of existing MANET protocols and modify and tailor them for VANET applications. In this section, we propose two routing protocols based on AODV and GPSR [16], highlight their drawbacks, and analyze their performance.

We first make an assumption that all vehicles (or nodes; we interchangeably use them) are GPS enabled and equipped with wireless transceivers. With GPS, the nodes will get their positions, as well as map information. The wireless transceivers enable the nodes to communicate with surrounding nodes within some communication range and direction. For example, in the MobiSteer Project [27], directional antenna and beam steering were used to improve the communication performance between moving vehicles and roadside access points.

C. CBRF

The CBRF algorithm works like AODV. It builds routes using a route request (RREQ)/route reply (RREP) query cycle. First, RREQ packets are broadcast across the network to discover a route from the source to the destination. A node receiving an RREQ packet sends RREP packets back to the source if it has a route to the destination or is the destination itself; otherwise, it will rebroadcast the RREQ. Data packets are sent to the destination after the source node receives the RREP. The routing information is updated to ensure that the best route is chosen. If a link breaks while the route is active, a route error (RERR) message is sent to the source node. The source node may then reinitiate a route discovery process.

However, CBRF differs from AODV in the way it controls the undesired broadcast of the RREQ, i.e., the broadcast storm [18], [28]. CBRF restricts the range of such broadcast packets and makes them "multicast." We assume that the communication range for a vehicle is R and argue that only vehicles that are toward the destination would be useful in routing. Thus, the RREQ is not intended for the other half, and we can use steerable beam directional antenna to control the broadcast direction [27]. Moreover, within the useful half, we set a reference distance $r < R$, as shown in Fig. 5. The nodes within a distance

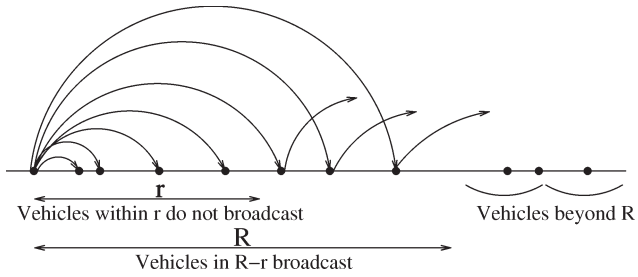


Fig. 5. Restricted forwarding.

of r are not allowed to broadcast, i.e., they are forcefully made deaf to any query packet. The reason behind this is that the vehicles in r would then rebroadcast the RREQ to range $R-r$, adding to the congestion and overhead. Of course, if the final destination happens to be within R, then it gets the packets. In CBRF, we make the broadcast RREQ only available in $R-r$ toward the destination.

In addition, due to the high mobility, a vehicle sometimes gets isolated, and packet forwarding stops. When the vehicles in front are slowed down due to traffic lights or stop signs, the data forwarding can be resumed. In such cases, when the next hop is temporarily unavailable, instead of generating an RERR and sending it back to the source, indicating a link failure, the current vehicle will act as a mobile link and cache the data packets in the buffer in a "carry-and-run" manner. Avoiding unnecessary route rediscovery, CBRF is expected to increase the packet delivery ratio and reduce the overhead.

D. CLGF

CBRF is a connection-based routing scheme that sets up the route before data can be transferred. It is designed for some VANET applications that require a route to be established and maintained. When establishing and maintaining a route is costly, applications may choose not to set up a route *a priori*. In this regard, we propose CLGF, which follows the same routine as GPSR [16] in location-based forwarding. Both CBRF and GPSR are location-based routing protocols that exploit the correspondence between geographic position and connectivity in a wireless network by using the positions of the nodes to make packet-forwarding decisions. However, GPSR is greedy, because it always forwards packets to nodes that are progressively closer to the destination, if such a node exists. If not, GPSR lets a packet traverse successively closer faces of a planar subgraph of the graph until it reaches a node closer to the destination.

Along with the position update problems described in [26], it is also possible that, in GPSR, some hotspot nodes (in our case, vehicles) may be chosen as the intermediate forwarding nodes for multiple routing sessions, resulting in congestion at those nodes. This problem is even more serious in VANETs for 1-D roads, as shown in Fig. 6. Vehicles A and B hold sessions to send packets to C and D, respectively. As the last hop node nearest to the destination, vehicle E is chosen for both A and B. If the forwarding is not that greedy, vehicle F can be chosen as an alternative to alleviate the congestion in E.

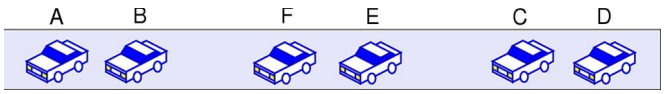


Fig. 6. Congestion caused in GPSR.

To determine the level of congestion that a node is currently experiencing in CLGF, we consider the queue length at the medium access control (MAC) layer. The MAC layer then calculates the instantaneous level of congestion as a ratio of the instantaneous queue length to the total buffer size. Finally, the node appends the level of congestion information in the periodically broadcast beacon signal (HELLO packets), along with its own identifier and position. When a forwarding node receives the beacon signal, it compares the level of congestion to a predefined threshold and only considers a subset of noncongested nodes whose congestion values are less than the threshold. The forwarding node will choose the next hop as the node progressively closest to the destination in the subset of noncongested nodes. In our implementation, the level of congestion threshold is set to be 0.8.

E. Simulation Model and Setup

Integrating the mobility traffic model traces with the investigated routing protocols, we conduct ns-2-based network simulation. We use the 2.28 version of the simulator in our experiments. To simulate a DSRC VANET environment as realistically as possible, we set the wireless signal frequency to 5.9 GHz with 10-MHz bandwidth. We also assume that vehicles are equipped with bidirectional antennas that can radiate fixed and equal transmission powers of 20.4 dBm, making the communication range about 400 m. As dictated by the standard, we tune the DSRC receiver sensitivity to -77 dBm. We adopt the IEEE 802.11 MAC and set the data rate to 6 Mb/s, which is one of the suggested data rates in the DSRC standard [36]. The simulations are conducted for both road layouts, i.e., downtown and residential, as mentioned in Figs. 1 and 2. The number of vehicles ranged from 50 to 250. The simulation time was 900 s (excluding transients) for our vehicular mobility model simulator trace. (Note that we had the trace for 1000 s.) Ten randomly chosen source nodes generated CBR traffic for their respective destination nodes. For every 50-s epoch, a new set of source/destination vehicles was chosen to characterize the randomness in network traffic flows. The default CBR data packet size was 64 B, and the rate was 2 packet/s.

We would like to emphasize that our aim is not to compare CBRF with CLGF. The reason is that CBRF is aimed at delivering packets to a specific destination, whereas CLGF is better at routing the packets in some direction (of the destination). Instead, we are interested in understanding their performance constraints and how they compare to existing MANET routing protocols, i.e., AODV and GPSR. In CBRF, we make $r = 250$ m, i.e., all vehicles within 250 m of the sender ignore the broadcast RREQ packets. Through simulation, we find that it is a suitable setting for r. In the CLGF setting, vehicles advertise their MAC layer queue lengths in periodically broadcast HELLO packets. For measuring the performance, we use three metrics to evaluate and compare routing protocols.

1) *Average data packet delay:* This metric is the average of all data packet delays from the source to the destination. It reflects the latency of a route and the routing protocol. It also reflects the level of congestion.
2) *Data packet delivery ratio:* This metric measures how much data are successfully delivered by the routing protocol. It is a measure of the routing effectiveness.
3) *Routing overhead:* This metric is the ratio of the number of administrative routing packets to the number of successfully received packets.

F. Performance Study

The performance of CBRF and AODV for both downtown and residential areas is shown in Fig. 7. The average data packet delays are smaller when CBRF is used. For the data packet delivery ratios, although residential areas generally get a lower ratio of successful packet delivery, the improvement due to CBRF is clear. CBRF also lowers the overhead to almost half. The improvement is twofold: First, by controlled broadcast of RREQ, many unnecessary packet transmissions are saved. Second, if a vehicle could "cache and carry" data along the way than report a broken link and set up a new route, it can save the administrative packets for new route discovery.

The performance of CLGF and GPSR is shown in Fig. 8. We find that the delays do not increase that much with increasing number of vehicles, which shows that CLGF and GPSR are scalable. CLGF gives a better delay performance by considering the congestion information given by queuing delays. Using CLGF, we also get better packet delivery ratios for denser networks where the number of vehicles is large, and the data traffic is high.

G. Impact of Road Layouts and Mobility

It is commonly accepted that the road layouts influence the traffic pattern and, hence, the network performance. We first consider the data packet delivery ratio to show this effect. In the downtown area, where the roads are of a more regular "grid" structure, vehicles have a higher average speed. The faster the vehicles move, the larger the chance that they meet with others. This chance helps deliver packets when the vehicle density is low [see Fig. 7(b)]. In addition, the "grid" layout increases the probability of data delivery at the intersections, because vehicles reduce speed at intersections. At smaller intervehicle distances, the vehicles have a better chance of setting up good and direct wireless communication. The "grid" layout has a large number of intersections, and packet delivery benefits from this structure. Due to the irregular road layouts at the residential area, the vehicle moves slower, which hampers the intervehicle communications. However, when the number of vehicles is large, the road layout is no longer the dominating factor. The impact of road layouts is also justified in Fig. 7(a). When the vehicle density is low and vehicles get isolated, a slow-moving vehicle takes longer time to reach (be in the communication range) other vehicles, resulting in a larger delay. We observe

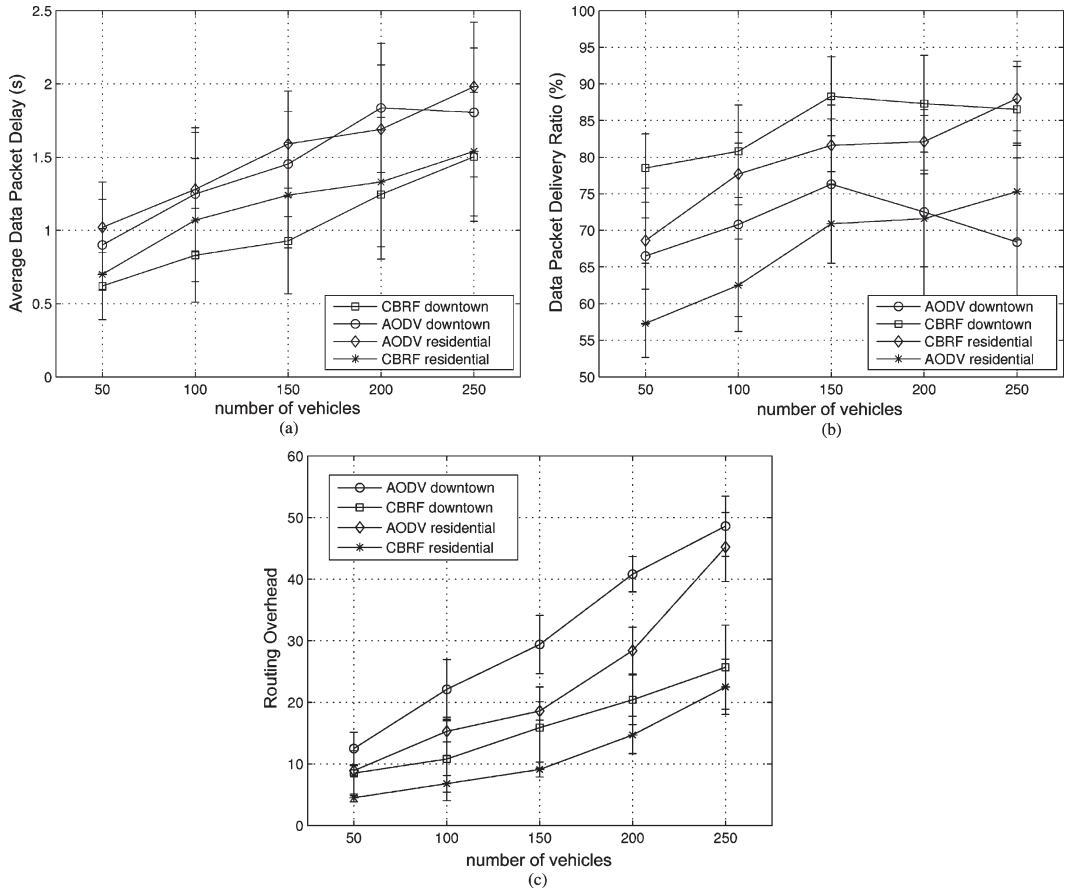


Fig. 7. Performance of CBRF and AODV. (a) Average data packet delay. (b) Average data packet delivery ratio. (c) Routing overhead.

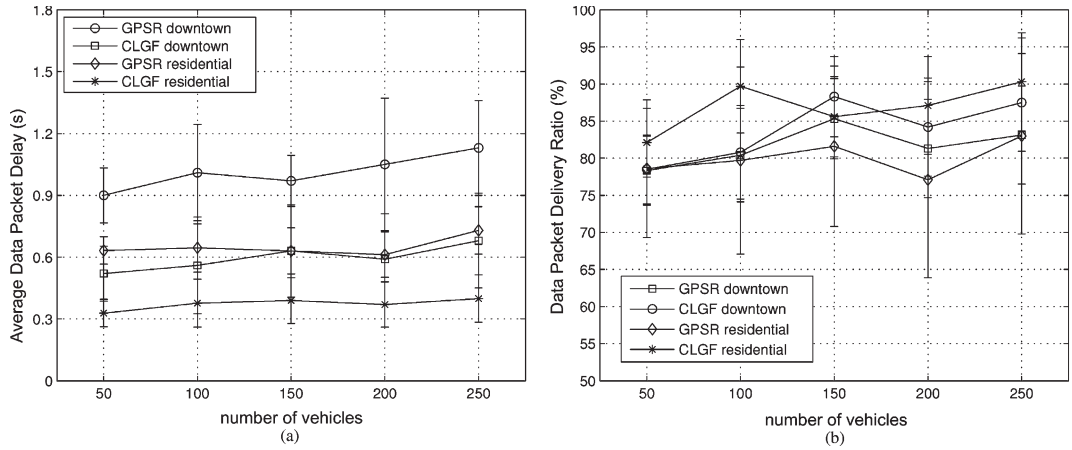


Fig. 8. Performance of CLGF and GPSR. (a) Average data packet delay. (b) Average data packet delivery ratio.

the delay difference in Fig. 7(a) when the number of vehicles is 50. Again, when the number of vehicles increases, factors such as MAC contention and backoff contribute to the delay.

Furthermore, it is also interesting to know how the network performance is affected by the mobility model. In Fig. 9(a) and (b), we show the performance of CBRF and CLGF with different intervehicle distance d, acceleration a, and overtaking probability p. It is observed that, although the network is tested under different mobility parameters, the average packet delay and average packet delivery ratio are not altered much. Therefore, we can infer that the precise implementation of the mobility profiles does not have a significant impact on the network performance.

V. TOPO: Routing in Large-Scale Vehicular Ad Hoc Networks

Let us now focus on the routing in a large-scale VANET, i.e., an area of several or tens of square miles. Such a network can be used to deliver messages in a metropolitan area. The application for this large-scale routing can be either mobile to static (e.g., a vehicle queries catering information and a vehicle talks to

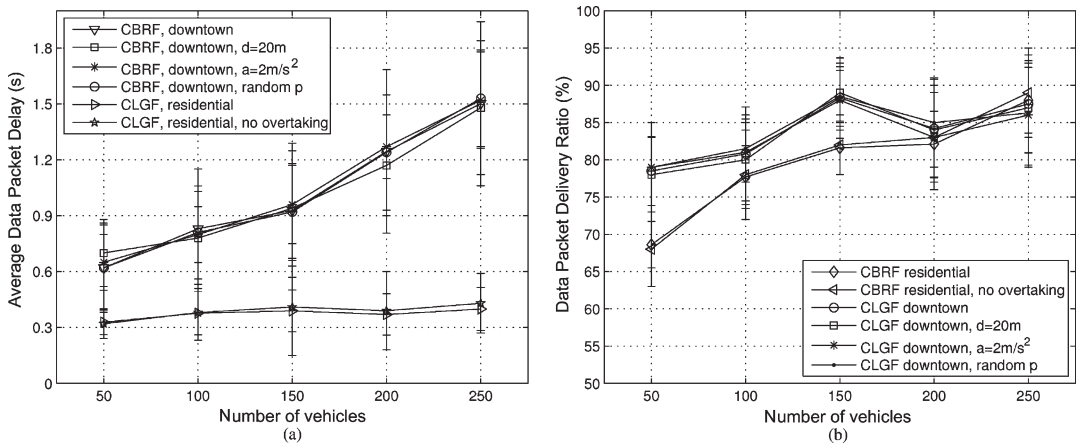


Fig. 9. Performance of CBRF and CLGF with different mobility parameters. (a) Average data packet delay. (b) Average data packet delivery ratio.

Fig. 10. Map used for large-scale simulation (courtesy of maps.google.com).

a distant road side unit for digital map update) or mobile to mobile (e.g., vehicle data transfer/exchange, online intervehicle video conference or gaming, and other real-time applications). The aforementioned small-scale routing protocols face some difficulties and are inappropriate for this scenario, as will be shown later in Fig. 11(a) and (b). Thus, a new routing protocol for a large-scale VANET is needed.

A. Definitions: Overlay and Access

We define *overlay* as high-density high-speed limit roads (e.g., state roads and interstate roads), which make the backbone of the transportation system, e.g., the thick roads in Fig. 10. It can be inferred that, in general, message delivery on the *overlay* will enjoy less delay, because there are more nodes, and they move faster. As the complement of *overlay, access* consists of the remaining roads. *Access* covers residential areas and low-speed roads. Usually, *access* areas are located along the *overlay* and intersect with the *overlay*. This way, two isolated *access* areas can be linked through the *overlay*. Given a local map, e.g., a TIGER map [34], the *overlay* can be extracted by traversing the map and getting the speed limits of each road. The *overlay* constructed from a TIGER map is a graph. It is to be noted that the graph may not be a connected graph, as the *overlays* may also be connected by *access*. With the map and GPS information, a vehicle can calculate information, such as the positions of the intersections between the *access* and *overlay*, and the distance from the intersection.

We further define a data structure called neighbor list that each of the nodes maintains. The nodes discover and keep their neighbor information in the list. This information includes neighbor IDs, next intersection IDs, and the current phase (in *access* or *overlay*). The ID of each node is unique, and it is easy to implement, as their transceivers will have different serial numbers. Intersection IDs can be obtained from the TIGER map. The phase is determined by the road where the node is on. The neighbor lists are periodically updated to refresh the surrounding information. Moreover, without loss of generality, we assume that the message exchange in the VANET-oriented system (e.g., ITSs) follows some common format: A packet can be identified with the contents in the header. Such contents will help categorize the application types delivered through the packets, e.g., safety information and commercial information.

B. Two-Phase Routing Scheme

For the routing packet from any given source to destination, there can only be four combinations of the $\langle$source, destination$\rangle$ tuple. They are $\langle access, access\rangle$, $\langle access, overlay\rangle$, $\langle overlay, access\rangle$, and $\langle overlay, overlay\rangle$. If both nodes are in the *overlay*, then only *overlay* routing is enough. If only one of them is in the *overlay*, routing will consist of both the *overlay* and *access* phases. If both of them are in *access*, routing can be done with a pure *access* phase or a mix of both phases (i.e., from *access* to *overlay* and then to *access*). When dealing with nodes that are both in *access*, the TOPO will first evaluate how far away the nodes are separated. From our experimental study, we observe that, if the scale is of 1 km, single-stage small-scale routing protocols work fine. Thus, the TOPO will switch to the use of a small-scale routing protocol if and only if the following conditions are satisfied: 1) Both the source and destination are in *access.* 2) They are connected without *overlay*, or they are no farther than 1.5 km apart.

Let us sketch how the TOPO works. The TOPO borrows the philosophy of Internet routing, where the message from one end user to another will be relayed on both the Internet backbone and local access networks. We give the most general case in VANET routing, i.e., both ends of the route in *access*. The source node is aware of the position of the destination and, thus, can calculate the shortest path on the map to that destination using path-planning algorithms. We argue that it is only necessary and efficient to do the path planning in the *overlay*. After the paths have been planned, the source node

will begin the TOPO with the *access routing phase* and deliver the messages to the *overlay*. Then, the routing switches to the *overlay routing phase*, and messages will be delivered along the path to the *access* areas. Finally, the TOPO will switch back to the *access routing phase* and send the messages to the destination node.

Before we present the TOPO in detail, we first explain why we consider the TOPO to be a framework for routing in large-scale VANETs. The two phases in the TOPO are independent of each other, which makes it easy to implement different routing schemes in either phase, provided that there are some infrastructure support at the intersections. The link between the two phases is the handover of packets, where the packets exist in one routing phase and enter the other, and the starting/ending point of any routing algorithm. If the routing scheme in *access* is R_1, it only needs to set the destination at one of the intersections with the *overlay*. Then, routing scheme R_2 gets the packet and routes it to another intersection with *access*. Finally, R_3 acts as the routing in *access* again.

C. Access Routing Phase

As previously discussed, various routing protocols can reside in the two routing phases CBRF and CLGF. The objective behind the design of *access* routing is that it must be fast and reliable and have low-complexity message delivery. We argue that, since, in the *access* areas, the node density is not high, and it is not efficient to make a message delivery along a predefined path. In this regard, we adopt CBRF and CLGF as the routing protocols in the *access routing phase* and present the TOPO accordingly. It is noted that some infrastructure support may be needed if some other routing protocol is adopted.

However, several issues need to be addressed. First, it is necessary to distinguish the stages of *access* to *overlay* from the stages of *overlay* to *access*. In the former case, the nodes just know how to forward the messages to the *overlay*, which is a kind of static infrastructure; in the latter case, the destinations may be mobile, and we need to discover them and reach there. Thus, in the first stage, CLGF is more appropriate as long as the direction of the *overlay* is given; in the other stage, CBRF will do better, because it involves a route discovery, as destinations may be mobile.

To find an efficient way to guide the messages to the *overlay*, a node in *access* needs to find the nearest intersection of the *overlay* and *access*. This can easily be done by current GPS technology or simple calculation of the Euclidean distances from the source node to each of the intersections. The source node will invoke CLGF to forward the message to the nearest intersection. However, while checking its neighbor list and forwarding the message to the next hop, a node will also check if it has a neighbor in the *overlay*. If so, the message will directly be forwarded to the node in the *overlay*, and the *access routing phase* ends. Although there might be a chance that a message forwarded to the intersection cannot further be forwarded to a node in the *overlay*, we find it very unlikely to happen, as later discussed in Section IV-E.

We consider the second stage, when the message is delivered to a node in *access* from the *overlay*. A node will start route discovery to establish a route to the destination node. If the destination node is either static or mobile within the same *access*, CBRF will easily find the destination node. However, the worst case occurs when the destination node is mobile and moves out of *access* to the *overlay*. In the TOPO, we make the destination node acknowledge its neighbors about its position change at the time it moves on to the *overlay*. In particular, the destination node broadcasts packets containing its direction in the *overlay* (next intersection ID) and a time-to-live (TTL) value. When a node in *access* gets an RREQ for the destination and this request is received within the TTL, it will reply with the next intersection ID to guide the packet back to the *overlay* and follow the movement of the destination node. After the destination node receives the message from the source node, it can include its updated position information in the messages back; however, this process is outside the scope of the TOPO.

D. Overlay Routing Phase

The process between the two stages of the access phase is the overlay routing. Again, the routing in the overlay can employ any arbitrary routing protocol; we present a simple yet efficient one in this paper. We argue that, in the overlay, the message is delivered and forwarded along the precalculated path to the destination node (if it is in the *overlay*) or to an intersection with *access* (if destination is in *access*). The forwarding is done in a greedy manner for each node: as long as the destination (intersection) is not within the communication range, it just forwards the message to the farthest node along the path, provided that the node is not overwhelmed. However, to make it work, we add some rules for the nodes.

1) If there are no nodes ahead of the current message-carrying node C along the path and it is moving along the path, it carries the message until it finds some other node to forward.
2) At an intermediate intersection A in the *overlay*, if there is no traffic along the path (to intersection B) and C is not heading in that direction, it forwards this message to any node heading intersection A and sets a TTL to that message. The node receiving that message will do the forwarding again. If it is not able to forward the message to intersection B, it will redo this step, except for setting the TTL. If the TTL value in a message exceeds a predefined threshold, the node currently holding the message will delete edge $e = \langle A, B \rangle$ on overlay graph G, recalculate the path for this message, and continue forwarding.
3) At the final intersection F, where the message should get off the *overlay* and continue in *access*, if there is no node in *access* to relay the message, C will forward this message to any node heading F, making the message bounce around F until some node in *access* is able to get it.

It is to be mentioned that the *overlay* network may not be a connected graph, i.e., two or more *overlays* can be connected by *access*. Thus, during path planning, in such scenario, the TOPO will set the end intersection of the *overlay* as the temporary destination and invoke the TOPO routing process again when the packet reaches that point. However, in our experimental study,

we find that the disconnected *overlay* is unlikely to happen. We summarize the TOPO as pseudocodes in Algorithm 1.

Algorithm 1 TOPO

procedure TPO (s, d, p) ▷ The source, destination, and packet
  $r \leftarrow distance(s, d)$
  if $s,\ d \in access$ && $(r < 1.5$ km$\|$ $no\ overlay\ between\ s, d)$ **then**
    CLGF(s, d, p)
  else
    if $s \in access$ **then**
      ACCESS-ROUTING-PHASE (s, d, p)
    else OVERLAY-ROUTING-PHASE (s, d, p)
    end if
  end if
end procedure
procedure ACCESS-ROUTING-PHASE (s, d, p)
  if p is not from overlay **then**
    $n \leftarrow$ nearest intersection
    while neighbor list ! = NULL **do**
      if overlay node $o \in$ neighbor list **then**
        SEND (o, p)
      else $o \leftarrow$ CLGF(s, n, p)
      end if
    end while
    OVERLAY-ROUTING-PHASE (o, d, p)
  else RREP $\leftarrow$ CBRF (s, d, p)
    if RREP == destination left **then**
      TOPO (here,new_des, p)
    end if
    Update_Position_Info (d, s)
  end if
end procedure
procedure OVERLAY-ROUTING-PHASE (s, d, p)
  Calculate the path to d.
  If $d \in overlay$ **then**
    GREEDY-FORWARDING (s, d, p)
  else get the last intersection id i on the path
    GREEDY-FORWARDING (s, i, p)
    ACCESS-ROUTING-PHASE (i, d, p)
  end if
end procedure
procedure GREEDY-FORWARDING (s, d, p)
  while $d \notin$ neighbor list do
    if OVERWHELMED($n_f \leftarrow$ farthest node) **then**
      $n_f \leftarrow$ second farthest node
    end if
    Forward p to n_f along the path subject to the rules in V-D.
  end while
end procedure
procedure OVERWHELMED (n)
  $l \leftarrow$ MAC Queue Length of n
  if $l < f_o \times$ buffer size **then return** False
  else return True
  end if
end procedure

E. ITS Friendliness

We further discuss how the TOPO can be used in ITSs, which is targeted to deliver information between vehicles and/or infrastructure. Since the backbone of the TOPO relies on the *overlay* (highways), where the wireless channel may highly be congested and the bandwidth is limited due to ITS applications, the main challenge for the TOPO is how to perform routing without compromising other services in ITSs.

The basic approach in addressing ITS friendliness is prioritization. From the header of the packets in ITSs, we can get the application type of the message. Based on the service type, we prioritize packets, e.g., packets with safety information will have higher priority than packets with commercial information. The prioritization categorizes packets by giving them different weights, and scheduling is done using the weighted fair queuing technique. When there are N traffic flows with weights $w_1, w_2, \ldots, w_N$, the data rate for flow i is given by

$$R_i = \frac{Rw_i}{\sum_i w_i}$$

where R is the available link data rate.

VI. PERFORMANCE EVALUATION OF TOPO

We conduct simulations to study the performance of the TOPO. As indicated in [44], it is not efficient to use the current ns-2 release on a single computer to simulate a large number of nodes in a large topology; thus, we choose another memory-efficient network simulator, i.e., GTNetS [10]. GTNetS is proven to be competent of simulating as many as 50 000 nodes in a mobile ad hoc scenario [44]. We use the mobility model presented in Section III on a large topology, as we have shown in Fig. 10. The number of nodes is set to be 6000 unless specifically mentioned, and the area measures 10 km $\times$ 3 km. In our simulation, the *overlay* is chosen as the thick roads on the map, and we use Dijkstra's single-source shortest path algorithm in path planning. The rest of the environmental settings are the same as those for small-scale VANETs in Section IV-F. To test the performance of the TOPO, we choose different route lengths or source–destination pairs with different separation distances (i.e., 2000, 3000, 4000, and 6000 $\pm$50 m). It is worth mentioning that, since the nodes are mobile, the separation distance is the average distance from the source to the destination over a session, as the calculation of distance is evoked once a packet is received. Overwhelm factor $f_o =$ (MAC layer queue length/overall buffer size) is set to 0.8, and TTL $= 3$ s. We first allow ten CBR flows generating 2 pps. Each packet is 64 B. The effects of different packet sizes and rates are also studied. The simulation lasts 900 s, with each session lasting 100 s.

A. Performance

The results for the delay and packet delivery ratio are presented in Fig. 11(a) and (b), respectively. As a comparison, we also test the performance of two single-phase routing schemes AODV and GPSR. Fig. 11(a) shows the average end-to-end

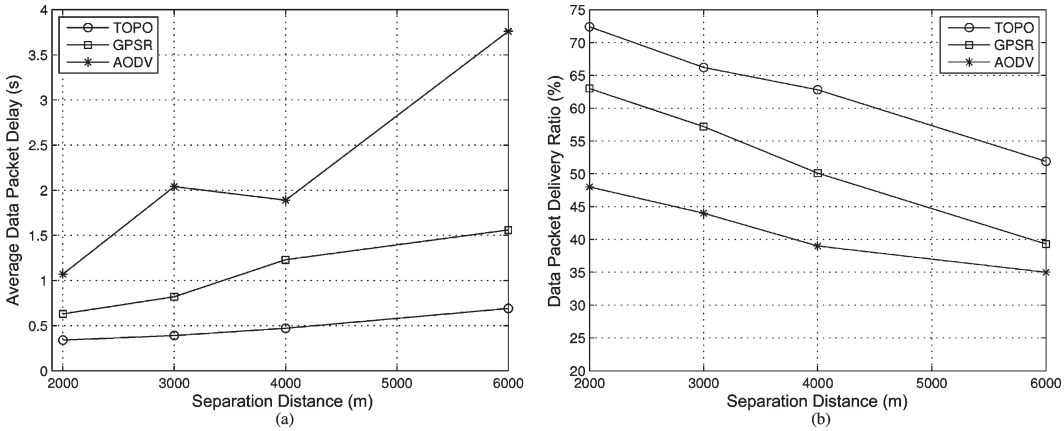


Fig. 11. (a) Delay performance. (b) Packet delivery ratio.

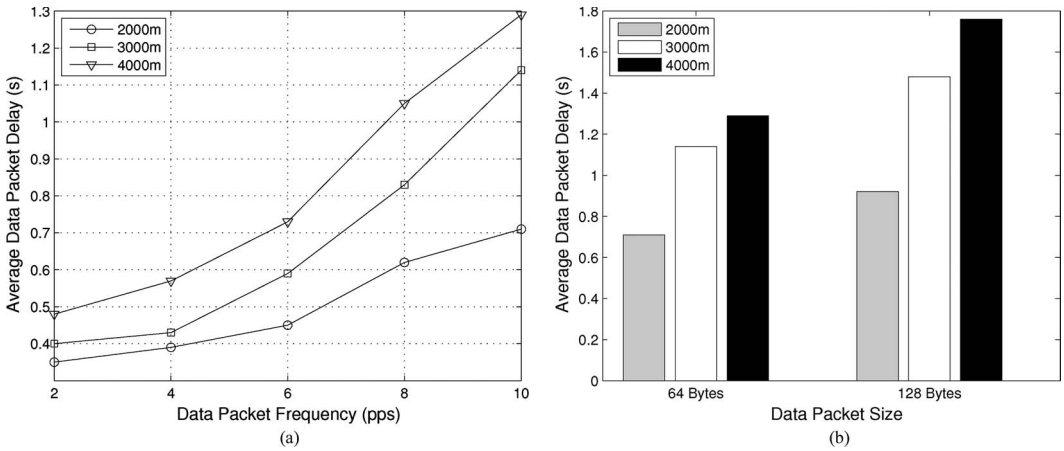


Fig. 12. Packet delay w.r.t. (a) packet rate and (b) packet size.

delay in packet delivery. Not only does the TOPO yield smaller delays but it also keeps a slow increase in delay with increasing distance. The reason behind the small delay is that the TOPO routes packets along an *overlay* path, which has high traffic density and high connectivity. Unless there are extreme cases when some portion of the *overlay* has very low traffic density, routing in the *overlay* will not generate higher delay. GPSR and AODV discard the traffic information and route packets from a pure networking perspective; thus, the delay becomes larger if the route is longer. However, AODV performs even poorer because of the cost associated with the route maintenance.

Fig. 11(b) shows the packet delivery ratio of the protocols. The performances of all protocols degrade with longer routes. However, the TOPO has a better delivery ratio than the other two. It is to be noted that, if there is no delivery failure in the *overlay*, which is true if the traffic density and direction diversity are high, the performance of the TOPO will mainly be determined in *access*. In other words, the performance of *access phase routing* (i.e., CLGF and CBRF) dictates the overall performance of the TOPO.

B. Effect of Packet Size and Packet Rate

We vary the amount of data packets in the network, as well as the packet size, in testing the performance of the TOPO. In particular, we are interested in the network delay performance under DSRC standards. It is suggested in [35] that typical DSRC applications generate packets of sizes larger than 100 B with more than 3 pps. In Fig. 12(a), we increase the data traffic volume by generating more packets per second. As expected, the delay shows an increasing trend when the traffic load is heavy. However, considering the length of the route, the delay performance is acceptable, particularly when compared with other protocols, as shown in Fig. 11(a), where the load is low. In Fig. 12(b), we set the packet generation rate at 10 pps and vary the data packet size. Plots indicate that an increase in data packet size does not significantly cause a larger delay.

C. Effect of Vehicle Density

We further study the effect of vehicle density on the network performance. The benchmarks used are 2000, 6000, and 8000 vehicles (66.7, 200, and 266.7 vehicle/km^2). Fig. 13(a) presents a drastic decrease in delay when the vehicle density increases. However, the delay gaps between 6000 and 8000 vehicles are not very impressive. The reason is given as follows: Although the higher node density decreases the delay by increasing the connectivity, possible packet collision due to channel access contention nullifies some of the effect and

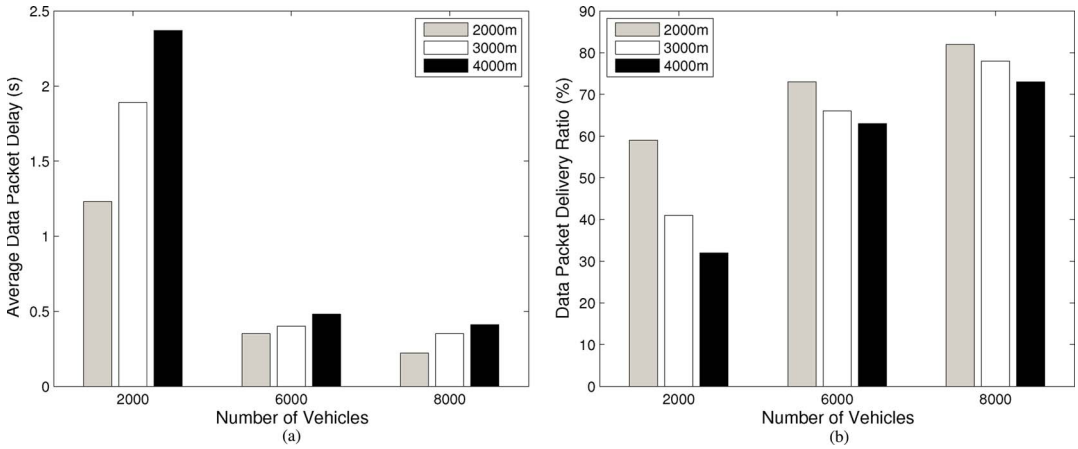


Fig. 13. (a) Packet delay. (b) Packet delivery ratio.

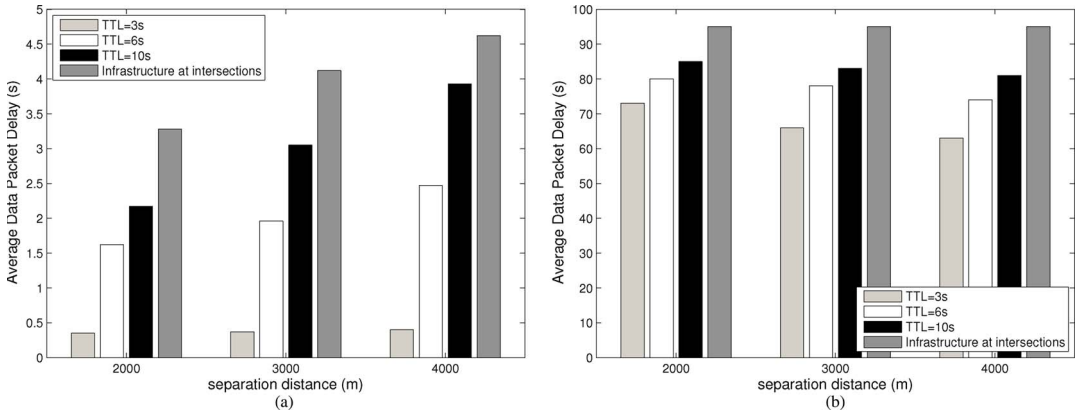


Fig. 14. Effect of packet-caching schemes on the (a) packet delay and (b) packet delivery ratio.

contributes to the delay. Fig. 13(b) clearly shows that a higher vehicle density means more successful data packet delivery ratio.

D. Effect of Packet Caching

As mentioned earlier, when the packet-carrying vehicle leaves the precalculated data-forwarding route at an intersection, it will send the packet to other vehicles on the route. A TTL value is set to indicate the lifetime of the packet. If we assume that a certain infrastructure, such as a data buffer [43], is available at the intersection, we can use the infrastructure to cache the packets and deliver them to other vehicles on the route at a later time. Fig. 14(a) and (b) explores these possibilities.

Fig. 14(a) indicates that the delay increases to a great extent when a larger TTL value is set or the infrastructure is assumed. This is because, when the next hop is not available, after waiting TTL time, the packet is rerouted. A larger TTL value means that the packet should wait at the intersection longer before it can be rerouted or continued on the forwarding process. The longer waiting time leads to a larger delay. The presence of infrastructure implies that the packets will always be cached and never be rerouted; thus, the delay is the largest. Fig. 14(a) also indicates that the increased amount in delay increases with the separation distance. However, in Fig. 14(b), the delivery ratios show the tradeoffs. With the support of the infrastructure, most of the packets can successfully be delivered when the delay is compromised, and smaller TTL values yield less of a delivery ratio. It can be inferred from the plots that, although the packet delivery ratio (PDR) decreases when the separation distance increases, the change rate is smaller when a larger TTL value is adopted. The tradeoffs can be used to tune the parameters of the TOPO when the awareness of the delay and/or delivery ratio is different for various applications.

E. Performance Under ITS

We also investigate the performance of the TOPO when other ITS data transfer sessions occur in parallel. Since the actual application type in ITSs is unknown, we do not distinguish data transfers in ITSs. This way, all ITS packets are treated equally and have higher priority than the TOPO routing packets. First, we define $w = (\text{weight of ITS}/\text{weight of the TOPO})$ and assume that there is only one saturated ITS session in the system. The throughput of the TOPO and ITS sessions are shown in Fig. 15(a). Then, we fix $w = 2$, vary the number of simultaneous ITS flows, and record the throughput in Fig. 15(b). Plots indicate that, although the throughput is not proportional to w or the number of ITS flows, the throughput of the ITS sessions is not compromised when the TOPO is implemented. Thus, we claim the TOPO as an ITS-friendly protocol.

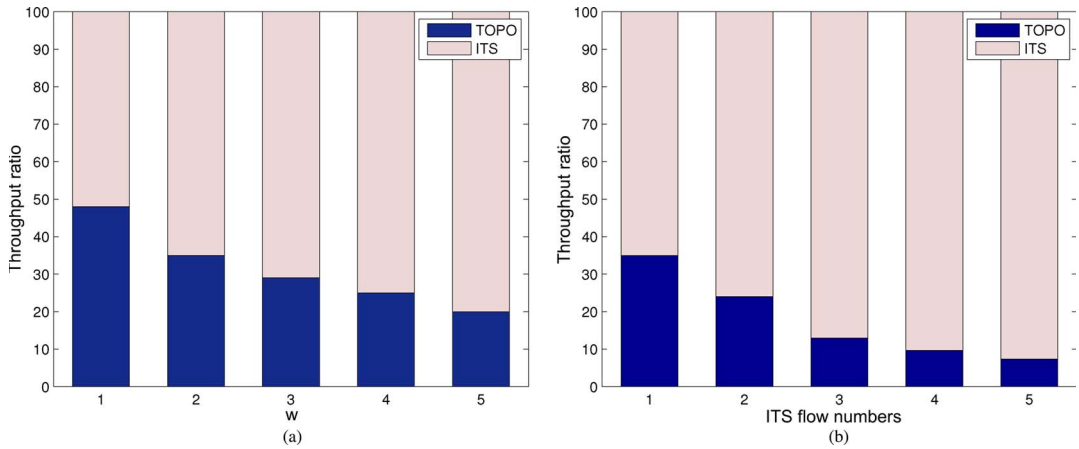


Fig. 15. Throughput ratio of the TOPO and ITS sessions w.r.t. (a) w and (b) the number of flows.

VII. Conclusion

Research on VANETs needs integrated study of vehicular mobility models and network protocols. Physical-world traffic rules, such as road layouts and traffic regulations, have a significant impact on the networking performance and, hence, deserve careful investigation. In this paper, we have proposed a vehicular mobility model based on real-life scenarios. We have simulated vehicle movement traces using real-world maps. Based on the traces, we have designed new routing protocols for VANETs.

Considering the routing issues of VANETs, we have first proposed two small-scale VANET routing schemes and carefully studied their performance. We have shown how the packet delivery ratio and delay are affected by different schemes, vehicle densities, and road layouts. However, when applying the existing protocols to large-scale VANETs, the performance degrades. We have argued that it is possible to exploit the road diversity that could potentially provide insights into designing better VANET routing schemes. To this end, we have distinguished the *overlay* roads from the *access* roads, based on the vehicle density and speed limit, and suggested that routing can independently be done on both types of roads. We have presented a TOPO and addressed the packet-routing issue similar to a vehicle moving on a map. The proposed TOPO can also be regarded as a framework in large-scale VANET routing that is compatible with various single-stage routing protocols. We have conducted simulations to verify our ideas, and results have shown much better performance, compared with those of existing methods. Furthermore, results have also shown how the performance of the TOPO is affected by the packet size, packet rate, vehicle density, and packet-caching schemes. As an added benefit, the TOPO has also achieved ITS friendliness with packet prioritization.

References

[1] A. Abdrabou and W. Zhuang, "Statistical QoS routing for IEEE 802.11 multihop ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1542–1552, Mar. 2009.

[2] L. Abusalah, A. Khokhar, and M. Guizani, "A survey of secure mobile ad hoc routing protocols," *Commun. Surveys Tuts.*, vol. 10, no. 4, pp. 78–93, 4th Quart. 2008.

[3] F. Bai, N. Sadagopan, and A. Helmy, "IMPORTANT: A framework to systematically analyze the impact of mobility on performance of routing protocols for ad hoc networks," in *Proc. IEEE INFOCOM*, 2003, pp. 825–835.

[4] M. Brackstone and M. McDonald, "Car-following: A historical review," *Trans. Res. F*, vol. 2, no. 4, pp. 181–196, Dec. 1999.

[5] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden, "A measurement study of vehicular Internet access using in situ wi-fi networks," in *Proc. ACM Mobicom*, 2006, pp. 50–61.

[6] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," in *Proc. ACM Mobicom*, 2002, pp. 483–502.

[7] L. Chen and W. B. Heinzelman, "A survey of routing protocols that support QoS in mobile ad hoc networks," *IEEE Netw.*, vol. 21, no. 6, pp. 30–38, Nov./Dec. 2007.

[8] D. R. Choffnes and F. E. Bustamante, "An integrated mobility and traffic model for vehicular wireless networks," in *Proc. ACM VANET*, 2005, pp. 69–78.

[9] *DSRC*. [Online]. Available: http://www.standards.its.dot.gov/Documents/advisories/dsrc_advisory.htm

[10] *The Georgia Tech Network Simulator (GTNetS)*. [Online]. Available: http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/

[11] J. Haerri, M. Fiore, F. Filali, and C. Bonnet, "VanetMobiSim: Generating realistic mobility patterns for VANETs," in *Proc. ACM VANET*, 2006, pp. 96–97.

[12] H. Hartenstein, B. Bochow, A. Ebner, M. Lott, M. Radimirsch, and D. Vollmer, "Position-aware ad hoc wireless networks for inter-vehicle communications: The fleetnet project," in *Proc. ACM/IEEE MobiHoc*, 2001, pp. 259–262.

[13] H. Huang, P. Luo, M. Li, D. Li, X. Li, W. Sun, and M. Wu, "Performance evaluation of SUVnet with real-time traffic data," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3381–3396, Nov. 2007.

[14] M. Jerbi, R. Meraihi, S.-M. Senouci, and Y. Ghamri-Doudane, "GyTAR: Improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments," in *Proc. ACM VANET*, 2006, pp. 88–89.

[15] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Norwell, MA: Kluwer, 1996, pp. 153–181.

[16] B. Karp and H. T. Kung, "Greedy perimeter stateless routing for wireless networks," in *Proc. ACM/IEEE Mobicom*, 2000, pp. 243–254.

[17] M. Khabazian and M. Ali, "A performance modeling of connectivity in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2440–2450, Jul. 2008.

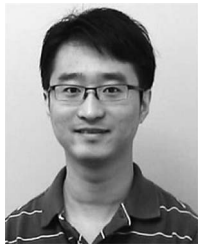
[18] J. Li, C. Blake, D. De Couto, H. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *Proc. ACM Mobicom*, 2001, pp. 61–69.

[19] C. Lochert, M. Mauve, H. Fusler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 9, no. 1, pp. 69–72, Jan. 2005.

[20] J. Luo, D. Ye, X. Liu, and M. Fan, "A survey of multicast routing protocols for mobile ad-hoc networks," *Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 78–91, Jan. 2009.

[21] A. Mahajan, "Urban mobility models for vehicular ad hoc networks," M.S. thesis, Florida State Univ., Tallahassee, FL, Spring 2006.

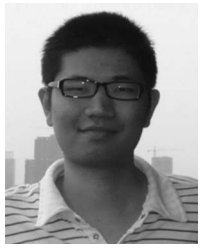
[22] R. Mangharam, D. S. Weller, D. D. Stancil, R. Rajkumar, and J. S. Parikh, "GrooveSim: A topography-accurate simulator for geographic routing in vehicular networks," in *Proc. ACM VANET*, 2005, pp. 59–68.
[23] H. Menouar, M. Lenardi, and F. Filali, "Improving proactive routing in VANETs with the MOPR movement prediction framework," in *Proc. 7th Int. Conf. ITS Telecommun.*, 2007, pp. 1–6.
[24] Z. Mo, H. Zhu, K. Makki, and N. Pissinou, "MURU: A multi-hop routing protocol for urban vehicular ad hoc networks," in *Proc. MobiQuitous*, 2006, pp. 1–8.
[25] G. H. Mohimani, F. Ashtiani, A. Javanmard, and M. Hamdi, "Mobility modeling, spatial traffic distribution, and probability of connectivity for sparse and dense vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1998–2007, May 2009.
[26] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces," in *Proc. ACM Mobihoc*, 2006, pp. 108–119.
[27] V. Navda, A. P. Subramanian, K. Dhanasekaran, A. Timm-Giel, and S. Das, "MobiSteer: Using steerable beam directional antenna for vehicular network access," in *Proc. ACM Mobisys*, 2007, pp. 192–205.
[28] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. ACM Mobicom*, 1999, pp. 151–162.
[29] M. Piorkowski, M. Raya, A. Lugo, P. Papdimitratos, M. Grossglauser, and J. Hubaux, "TraNS: Realistic joint traffic and network simulator for VANETS," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 12, no. 1, pp. 31–33, Jan. 2008.
[30] E. M. Royer and C. E. Perkins, "An implementation study of the AODV routing protocol," in *Proc. IEEE WCNC*, 2000, pp. 1003–1008.
[31] A. Saha and D. B. Johnson, "Modeling mobility for vehicular ad-hoc networks," in *Proc. ACM VANET*, 2004, pp. 91–92.
[32] W. Sun, H. Yamaguchi, K. Yukimasa, and S. Kusumoto, "GVGrid: A QoS routing protocol for vehicular ad hoc networks," in *Proc. IWQoS*, 2006, pp. 130–139.
[33] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support ITS services in VANET networks," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3337–3347, Nov. 2007.
[34] *TIGER: Topologically Integrated Geographic Encoding and Referencing System*. [Online]. Available: http://www.census.gov/geo/www/tiger/
[35] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Vehicle-to-vehicle safety messaging in DSRC," in *Proc. ACM VANET*, 2004, pp. 19–28.
[36] J. Yin, T. ElBatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, and T. Talty, "Performance evaluation of safety applications over DSRC vehicular ad hoc networks," in *Proc. ACM VANET*, 2004, pp. 1–9.
[37] H. F. Wedde, S. Lehnhoff, and B. van Bonn, "Highly dynamic and scalable VANET routing for avoiding traffic congestions," in *Proc. ACM VANET*, 2007, pp. 81–82.
[38] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: A mobility-centric data dissemination algorithm for vehicular networks," in *Proc. ACM VANET*, 2004, pp. 47–56.
[39] H. Wu, R. Fujimoto, G. F. Riley, and M. Hunter, "Spatial propagation of information in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 1, pp. 420–431, Jan. 2009.
[40] S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Analytical model for connectivity in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3341–3356, Nov. 2008.
[41] J. Yoon, M. Liu, and B. Noble, "Sound mobility models," in *Proc. ACM Mobicom*, 2003, pp. 205–216.
[42] J. Zhao and G. Cao, "VADD: Vehicle-assisted data delivery in vehicular ad hoc networks," in *Proc. IEEE INFOCOM*, 2006, pp. 1–12.
[43] J. Zhao, Y. Zhang, and G. Cao, "Data pouring and buffering on the road: A new data dissemination paradigm for vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3266–3277, Nov. 2007.
[44] X. Zhang and G. F. Riley, "Performance of routing protocols in very large-scale mobile wireless ad hoc networks," in *Proc. IEEE/ACM MASCOTS*, 2005, pp. 115–122.
[45] *The Network Simulator—ns-2*. [Online]. Available: www.isi.edu/nsnam/ns/



Wenjing Wang (M'09) received the B.Sc. degree from the University of Science and Technology of China, Hefei, China, in 2005 and the M.S. degree from the University of Central Florida (UCF), Orlando, in 2007, both in electrical engineering. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and Computer Science, UCF.

His research interests include applied game theory, wireless access networks, ad hoc networks, and vehicular internetworking.

Mr. Wang was a recipient of the Best Electrical Engineering Graduate Student Award in UCF and the AT&T Graduate Fellowship.



Fei Xie (M'09) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2004 and the M.S. degree in computer science from the University of Central Florida (UCF), Orlando, in 2008. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and Computer Science, UCF.

Mainak Chatterjee received the B.Sc. degree (Hons.) in physics from the University of Calcutta, Calcutta, India, in 1994, the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bangalore, India, in 1998, and the Ph.D. degree from the University of Texas, Arlington, in 2002.

He is currently an Associate Professor with the School of Electrical Engineering and Computer Science, University of Central Florida, Orlando. He serves on the executive and technical program committees of several international conferences. His research interests include economic issues in wireless networks, applied game theory, resource management and quality-of-service provisioning, ad hoc and sensor networks, code-division multiple access data networking, and link-layer protocols.