

# *i*CAR-II: Infrastructure-Based Connectivity Aware Routing in Vehicular Networks

Nizar Alsharif, *Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

**Abstract**—With the high demand of mobile Internet services, vehicular ad hoc networks (VANETs) have become a promising technology to enable vehicular Internet access. However, the development of a reliable routing protocol to route data packets between vehicles and infrastructure gateways is still a challenging task due to the high mobility and frequent changes of the network topology. The conventional position-based routing (PBR) in VANETs can neither guarantee the existence of a routing path between the source and the destination prior to the transmission nor provide connection duration information, which makes it unsuitable to route Internet packets. In this paper, we propose a novel infrastructure-based connectivity aware routing protocol called *i*CAR-II that enables multihop vehicular applications, as well as mobile data offloading and Internet-based services. *i*CAR-II consists of a number of algorithms triggered and run by vehicles to predict local network connectivity and update location servers with real-time network information, in order to construct a global network topology. By providing real-time connectivity awareness, *i*CAR-II improves the routing performance in VANETs by dynamically selecting routing paths with guaranteed connectivity and reduced delivery delay. Detailed analysis and simulation-based evaluations of *i*CAR-II demonstrate the validity of using VANETs for mobile data offloading and the significant improvement of VANETs performance in terms of packet delivery ratio and end to end delay.

**Index Terms**—Connectivity aware routing, link duration estimation, mobile data offloading, vehicular communications, vehicular internet access.

## NOMENCLATURE

$i, j$	Indices for vehicles, road segments, and intersections.
$v_i$	Vehicle with an identifier $i$ .
$R_i$	Road segment with an identifier $i$ .
$I_i$	Intersection with an identifier $i$ .
$R$	Transmission range for line-of-sight cases.
$\hat{R}$	Transmission range for nonline-of-sight cases.
$Loc_{v_i}$	Cartesian coordinates of $v_i$ location.
$S_{v_i}$	Reported average speed of $v_i$ .

$ES_{v_i}^*$	Predicted average speed of $v_i$ reported in a beacon message.
$ES_{v_i}$	Predicted speed of $v_i$ stored in the routing table.
$Sig_{v_i}$	Turning signal status of $v_i$ .
$RSSI_{v_i}$	RSSI value of $v_i$ .
$v_{iTable}$	Routing table of $v_i$ .
$k_{v_i}$	Vehicular density in front of $v_i$ .
$K_J$	Traffic jam density in urban environment.
$S_a$	Averaged maximum speed.
$f_{v_i}$	Binary variable indicates if $v_i$ is a <i>front</i> vehicle.
$l_{v_i}$	Binary variable indicates if $v_i$ is a <i>leading</i> vehicle.
$H_{v_i}$	Binary variable indicates if $v_i$ is moving towards a common intersection.
$d$	Distance between two vehicles.
$MLL_{v_i}$	Minimum predicted link duration with $v_i$ .
$RLL_{R_i}$	Minimum predicted road-level connectivity duration for $R_i$ .
$\mathbb{N}$	Set of neighboring vehicles.
$\mathbb{N}$	Set of potential forwarders.
$\mathbb{L}, \mathbb{F}, \mathbb{R}$	Sets of adjacent road segments representing <i>left</i> , <i>front</i> , and <i>right</i> road segments, respectively.
$\mathbb{M}_{v_i}$	Set of vehicles between two control packet forwarders.
$\mathbb{L}_{v_i}$	Set of <i>MLLs</i> with vehicles $\in \mathbb{M}_{v_i}$ .
$\mathbb{C}_{v_i, v_j}$	Set of common neighboring vehicles between $v_i$ and $v_j$ .
$\mathbb{I}_{R_i}$	Set of two intersections bounding $R_i$ .
$P$	Probability of initiating a road segment connectivity evaluation procedure.
$Last\_B$	Timestamp for the last mobility information update in the outgoing beacon messages.
$\tau_{Bt}$	Time period for updating mobility information in the outgoing beacon messages.
$\tau_{LinkUpdate}$	Time period for updating neighboring vehicles mobility information in routing tables.
$Last\_Upd_{v_i}$	Timestamp for $v_i$ mobility information update.
$\epsilon_{vel}$	Change in speed threshold to update a neighboring vehicle's mobility information in routing tables.

## I. INTRODUCTION

VEHICULAR communication allows many appealing information and traffic management applications that require Internet access. In vehicular ad hoc networks (VANETs), roadside units (RSUs) can work as Internet gateways for passing

Manuscript received November 5, 2015; revised March 31, 2016 and June 21, 2016; accepted June 22, 2016. Date of publication August 16, 2016; date of current version May 12, 2017. The review of this paper was coordinated by Prof. Y. Qian.

N. Alsharif is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada, and also with the Department of Computer Engineering, Al Baha University, Albaha 65431, Saudi Arabia (e-mail: nalshari@bcr.uwaterloo.ca).

X. Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: xshen@bcr.uwaterloo.ca).

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.2016.2600481

vehicles providing a low-cost drive-thru Internet [1]–[3]. With enabling multihop routing, vehicles forward Internet packets to extend the coverage of RSUs and Internet-based services [4]–[7]. This service, however, highly relies on the existence of forwarding vehicles and a reliable routing protocol. For sustainable and more reliable connectivity, stakeholders deploy cellular networks for in-car Internet access [8]. However, in dense areas with high vehicular traffic, and with respect to the explosive growth of mobile data traffic, the centric cellular networks can be easily overloaded. It is expected that the current mobile data demand will increase by 10 times and the monthly mobile data traffic will exceed 24 exabytes in 2019 [9]. Hence, using a hybrid network of VANETs and cellular network can support VANETs users with a more reliable low-cost Internet-based services and can enable mobile data offloading to mitigate the expected sever problem of cellular network overload.

One of the most challenging tasks to enable these features in VANETs is the design of routing protocols that cope with its highly dynamic topology. Unlike other networks, vehicles' high mobility and the frequent change of communication links between vehicles make the traditional topology-based routing protocols, such as AODV [10] and DSR [11], fail in VANETs as they flood the network with path finding and maintenance control messages [12]. Replacing this node-level network topology routing, an alternative geographical location-based routing paradigm or position-based routing (PBR) has been introduced [12]–[14]. PBR depends on routing packets among geographical locations by arbitrary nodes instead of routing among predetermined nodes. Studies confirm that this paradigm, PBR, outperforms topology-based routing in both urban and highway VANETs scenarios [12], [13].

PBR is a connectionless routing paradigm where a communication session is not required to be established before data transmission, and data packets are routed independently. Moreover, the existence of a routing path from the source to the destination during data forwarding and transmission is not guaranteed in PBR. To enable Internet access and mobile data offloading in a city VANETs, vehicles need instant information about connectivity to the core network before transmission. This information includes the existence of at least one routing path to an RSU gateway, in addition to the quality and the duration of the connection. Since PBR protocols do not support this information, a new routing paradigm, or an improved one, is required for Internet services in the heterogeneous network environment of VANETs and cellular networks.

In this paper, we propose a novel infrastructure-based connectivity-aware routing protocol, *iCAR-II*. *iCAR-II* deploys distributed algorithms to obtain real-time location and mobility information in order to estimate a minimum local network connectivity lifetime and experienced packet delivery delay (PDD) per road segment, and updates location centers (LCs) using cellular network channels. Thus, LCs can construct a city-level dynamically updated network view, or a real-time network topology, and support inquiring senders with up-to-date connectivity information, routing paths to gateways, and destination locations. With this global connectivity-awareness, *iCAR-II* significantly improves VANET performance and enables efficient mobile data offloading via RSUs.

The contribution of this paper can be summarized as follows.

- 1) We introduce a heuristic methodology to obtain a minimum communication link duration between each pair of communicating vehicles.
- 2) We introduce another methodology to obtain a road segment-level minimum connection duration.
- 3) We propose a distributed and dynamic PBR protocol that utilizes the introduced methodologies for efficient data routing and manages a cooperative operation between cellular networks and VANETs.

The remainder of this paper is organized as follows. Section II gives insight into related work. Section III describes the system model under consideration. The details of the proposed routing scheme are presented in Section IV, followed by analysis and simulation-based performance evaluation in Section V. In Section VI, we provide concluding remarks.

## II. RELATED WORK

Using multihop VANETs routing for Internet access and mobile data offloading is a recent research focus and only few works have considered its various challenges. Tan *et al.* [2], [3], [15] study Internet access in VANETs. In [2], the throughput of drive-thru Internet is studied considering one-hop vehicle-to-infrastructure scenario. In [3], a chain cluster scheme is presented for a cooperative content download and distribution among vehicles passing RSUs on highways. In [15], a strategy for RSUs placement is designed to enable multihop Internet access. Moreover, Cheng *et al.* [16] and Mouna Zhioua *et al.* [17] consider data offloading in the vehicular environment. In [16], challenges and possible solutions of offloading vehicular and cellular data traffic via drive-thru Internet are presented, while Karp and Kung [17] provide an analytical study to evaluate the potential of VANETs for cellular traffic offloading.

For the routing challenge in VANETs, many protocols have been proposed. One of the leading protocols that deploys PBR for mobile environment is GPSR [18]. GPSR uses *Greedy forwarding*, where packets are forwarded to nodes that are closer to the destination. When this strategy fails, GPSR uses *Perimeter forwarding* as a recovery strategy, where packets are forwarded around the perimeter of the failing region. In addition to the geographic location required by GPSR, other protocols [4], [14], [19]–[21] consider the availability of further network information for better routing performance. GSR [14] is an intersection-based routing that uses street maps and source routing, where the shortest path, by the means of intersections, is attached to each packet. A-STAR [19] uses a statistically rated map for street-traffic aware routing. TIGeR [20], GyTAR [21], and *iCAR* [4] deploy real-time vehicular traffic information for traffic aware intersection-based routing, where routing decisions are made at intersections based on local vehicular traffic information obtained from each road.

Most connectivity-aware routing protocols relate vehicular traffic density with connectivity, and tend to select dense roads in routing paths for better network connectivity. Only few works consider studying key connectivity metrics, such as link duration and connection lifetime, for urban VANETs routing. Wang *et al.* [22] present a prediction model to estimate link dura-

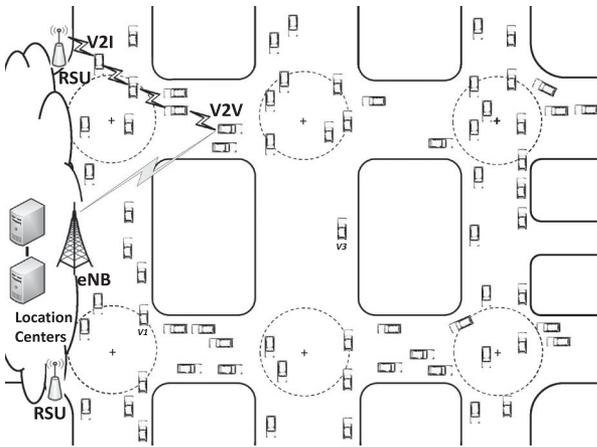


Fig. 1. Network model.

tion between two communicating vehicles considering relative speed, intervehicle distance, and the impact of traffic lights. In [23], a framework to analyze the network connectivity of urban VANETs based on link duration, connectivity duration, and re-healing time is provided. The framework considers relative velocity, traffic lights, and turning vehicles as the main causes of link breakage. In [24], a link duration estimation method is presented using cross-layer metrics. Physical layer information is used for better link duration estimation and long lifetime route construction in VANETs.

### III. SYSTEM MODEL

The network model considers a city scenario with a heterogeneous network of VANET and the fourth-generation cellular network, long-term evolution (LTE), as well as a set of location servers on the core network forming LCs, as shown in Fig. 1. City roads segments are bidirectional with variable length, width, and vehicle densities. Roads segments are bounded by either controlled or uncontrolled intersections. VANET consists of mobile vehicles equipped with on board units (OBUs), and RSUs along roads. OBUs have GPS receivers and can extract vehicles locations and velocity vectors. OBUs have access to vehicles' turn signals and are equipped with LTE interface for cellular communication and synchronization purposes. RSUs are partially deployed and do not provide full coverage to road segments; however, multihop forwarding is enabled to extend their coverage. Vehicles participate in multihop forwarding using their own OBUs, i.e., have sufficient inducements to forward packets belonging to other vehicles. RSUs work as gateways to infrastructure and both RSUs and LTE eNBs have access to LCs, e.g., via IP/UDP Internet protocols. All network entities use identical digital maps with well-defined roads and intersections.

In *i*CAR-II, vehicles frequently update LCs with their locations and local network status as described in Section IV. These updates are sent to LCs either via LTE channels or RSUs. Vehicles also periodically broadcast their locations, mobility vectors, and network status information (NSI) to their one-hop neighbors. LCs maintain tables of vehicles locations; a vehicle updates its location periodically or whenever it enters a new road segment. Moreover, LCs construct a dynamic network topology consisting of road segments weighted by experienced

PDD. Whenever a source vehicle has packets to transmit via the infrastructure, it chooses either to send via VANET or LTE, based on the available network connectivity information. If VANET disconnection is reported, the source either selects LTE mobile data or reschedules the transmission. If such information is not available, a source transmits an inquiry message to LCs via LTE to obtain network status along with the best route.

Infrastructure in the system consists of VANETs RSUs, LTE eNBs, and LCs servers. RSUs work as VANETs gateways to the core network. With respect to the deployment cost, higher RSUs deployment results in better VANETs connectivity and shorter routing paths. When connectivity to an RSU is not confirmed, LTE channels support VANETs routing with the following: Updating vehicles locations at LCs, updating local (road-level) connectivity information at LCs, and obtaining NSIs from LCs. Thus, LTE communication overhead depends on VANETs connectivity to the core network and the efficiency of *i*CAR-II to proactively confirm this connectivity. LCs play an important role in this design. They receive huge amount of updates, maintain updated network topology and vehicles locations, and respond to vehicles' inquiries. However, LCs can consider a design of distributed location servers that matches the geographically distributed nature of VANET. For example, a city road map can be divided into a number of vicinities and each server be responsible for one or more of these vicinities. Adjacent vicinities can exchange their real-time road-level network topology to have a wider network view, and a proper hierarchical server architecture will enable obtaining any destination location in the network. The details of LCs physical design such as map division and servers' management and allocation are out of this paper's scope, and LCs will be considered as one logical unit in the system hereinafter.

### IV. *i*CAR-II INFRASTRUCTURE-BASED CONNECTIVITY AWARE ROUTING

*i*CAR-II is a PBR routing scheme designed for multihop vehicular infotainment applications and Internet-based services as well as mobile data offloading. The principle of *i*CAR-II scheme is to support vehicles with instant information about VANET connectivity to infrastructure. Vehicular applications can, accordingly, decide to use VANET or LTE channels to access the core network. In order to achieve this principle, *i*CAR-II considers a number of algorithms and procedures run by vehicles' OBUs and LCs:

- 1) beaconing and neighborhood awareness;
- 2) mobility-based link lifetime estimation between each pair of neighboring vehicles;
- 3) road segment connectivity estimation;
- 4) city-level network topology construction and data routing.

For safety purposes, vehicles are required to periodically report road and driving conditions to nearby vehicles [8], [25]. This is achievable by VANETs' one-hop broadcast *beaconing* messages, which also includes vehicles location and mobility information. Using beacon information, vehicles estimate local connectivity lifetime with one-hop neighboring vehicles and achieve local neighborhood connectivity awareness. Beacons also help to exchange NSI, which includes connectivity status

to infrastructure, route to an RSU, and expected expiry time for that route. It will be shown later that routes in *iCAR-II* are represented by intersection *ID*s, and therefore, routes to infrastructure are different at different roads. Thus, NSIs are exchanged locally within road segments, while vehicles at intersections might receive NSIs from different roads.

When a vehicle  $v_i$  enters a road segment  $R_j$ , it initiates, with a probability  $P$ , a measurement procedure called road segment connectivity evaluation (RSE) by sending a unicast control packet (CP) that transverses the road segment to the other end, collecting some connectivity information from forwarders' routing tables. When failing to reach the destination intersection, CP is dropped due to local network disconnection, and a random backoff time is set in NSI. Otherwise, a vehicle at the other end reports the minimum expected connectivity lifetime of  $R_j$  and the experienced delivery delay of CP to LCs via LTE channels. The LCs' response, which includes one or more routes to RSUs, as well as a route lifetime, is attached in a beacon message and broadcasted to vehicles on  $R_j$ .

When a vehicular application or mobile data user needs to access the core network, or a nonneighboring vehicle, via *iCAR-II*, it either finds a valid route in NSI or sends an inquiry message to LCs via LTE. LCs locate the target destination, run a shortest-path algorithm, (e.g., Dijkstra) on part of the graph that includes both the source and destination, and send back an NSI message to the source. LCs include a number of RSUs in the source vehicle vicinity in its search in order to select the best route and suggest alternative routes. Upon receiving NSI with a valid route, the source starts the low-cost VANETs communication for the specified period of time and refreshes path information before the expiry time of the current path, if needed. In following, we describe the different stages of *iCAR-II* in more details.

#### A. Beaconing and Neighborhood Awareness

Every vehicle is required to broadcast road conditions periodically to its one-hop neighbors, to enable several safety applications. These messages are used by PBR in VANETs as beacons, or network heart beats, to support the awareness of a vehicle's existence, location, and communication channel status within the communication range. In addition to road and driving conditions, vehicles in *iCAR-II* are required to include some essential information to enable its functionality. Information includes

- 1) vehicle identifier ( $v_{ID}$ );
- 2) vehicle location coordinates ( $Loc_{v_{ID}}$ );
- 3) average driving speed ( $S_{v_{ID}}$ ) for the last  $m$  seconds;
- 4) driving direction ( $Dir_{v_{ID}}$ );
- 5) turning signal status ( $Sig_{v_{ID}}$ ); and
- 6) the predicted effective speed  $ES_{v_{ID}}$ , which is a function of  $S_{v_{ID}}$  and average speed of leading vehicles ( $LS_{v_{ID}}$ ) as will be shown in the next section.

Leading vehicles are the group of neighboring vehicles located in front of a transmitting vehicle, moving in the same road segment and direction, and having the same turning signal status. Leading vehicle average speed is easily calculated using information from the vehicle's routing table  $v_{IDTable}$ .

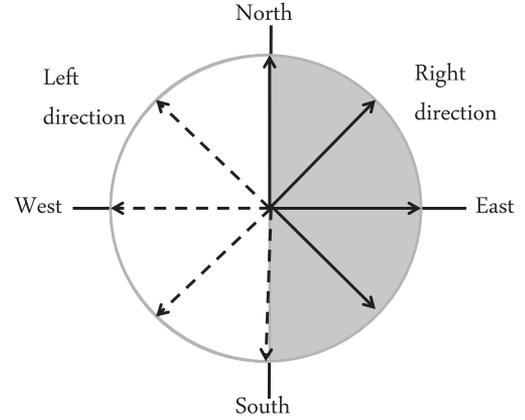


Fig. 2. Defining driving directions.

Each road segment is bounded by two intersections, has a unique identifier ( $R_{ID}$ ), and has two possible opposite directions. A vehicle is considered to be moving in a left (right) direction if it is heading any direction from north/south to west (east) as shown in Fig. 2. Turning signal variable  $Sig_{v_{ID}}$  can take one of three values representing two signaling directions *Right* and *Left* and an *Idle* status.

The routing table is a table that is maintained by each vehicle to store neighboring vehicles' information. In addition to routing information reported in beacons,  $v_{iTable}$  includes fields to track received signal strength indication  $RSSI_{v_{ID}}$ , timestamps of last recorded entries and row update  $Last\_Upd_{v_{ID}}$ , the estimation of minimum communication link lifetime  $MLL_{v_{ID}}$ , a binary variable  $l_{v_{ID}}$  to indicate if the vehicle belongs to the leading vehicles group, and another binary variable  $f_{v_{ID}}$  indicating if  $v_{ID}$  is located in front of  $v_i$  at the updating moment, regardless of its mobility direction and turning signal.

$v_{iTable}$  is maintained by adding new row information when receiving a beacon from a newly arrived vehicle to the communication range, updating row information when a beacon message is received from a neighboring vehicle, deleting a row information from the table when no beacon is received from a current neighbor for a certain period of time  $\tau_{delete\_row}$ , and updating  $l_{v_{ID}}$  and  $f_{v_{ID}}$  values with periods of time  $\tau_{l\_update}$  and  $\tau_{f\_update}$ , respectively. Row entries for an individual neighbor  $v_{ID}$  are updated periodically upon receiving a beacon message from  $v_{ID}$  with an acceptable  $RSSI$  and a period of  $\tau_{LinkUpdate} \cdot \tau_{delete\_row}$ ,  $\tau_{l\_update}$ ,  $\tau_{f\_update}$ , and  $\tau_{LinkUpdate}$  are much larger than the interbeacon interval in order to reduce  $v_{iTable}$  maintenance operations. In addition, a neighboring vehicle's information is updated if the difference between the reported predicted speed in the received beacon  $ES_{v_{ID}}^*$  and the recorded predicted speed in  $v_{iTable}$  exceeds a certain speed threshold  $\epsilon_{vel}$ , or if the remaining time before the expiry of  $MLL_{v_{ID}}$  is less than  $\epsilon_{MLL}$  as described in Algorithm 1.

Similarly, the routing information for a vehicle  $v_i$  is updated in the outgoing beacons periodically with respect to the timestamp of the last update  $Last\_B$  and a threshold value  $\tau_{Bt}$  to control the frequency of updating this information. Routing information in outgoing beacons is also updated upon detecting a change in NSI.

**Algorithm 1: Beaconsing.**


---

```

1: if Sending a Beacon then
2:   if  $|Current\_time - Last\_B| \geq \tau_{Bt}$  || New NSI has
   been received then
3:     Obtain  $S_{v_i}, LS_{v_i}, Sig_{v_i}, Dir_{v_i}$ 
4:     Update routing information in the Beacon
5:   else Reuse routing information
6:   end if
7:   Prepare a Beacon message with  $v_{ID}$ , road/driving
   status, routing information, timestamp
8:   Send the Beacon message for broadcasting
9: end if
10: if Receiving a Beacon then
11:   if  $RSSI_{v_{ID}} \geq RSSI_{threshold}$  then
12:     Extract  $v_{ID}$ 
13:     if  $v_{ID} \notin v_{itable}$  then
14:       Add  $v_{ID}$ , Find  $MLL_{v_{ID}}$ , and Complete  $v_{ID}$ 
   entries in  $v_{itable}$ 
15:     else Set  $Last\_velocity = ES_{v_{ID}}$  (table value)
16:       Set  $Crrnt\_velocity = ES_{v_{ID}}^*$  (beacon value)
17:       if  $|Crrnt\_velocity - Last\_velocity| \geq \epsilon_{vel}$ 
   ||  $Current\_time - Last\_Upd_{v_{ID}} \geq \tau_{LinkUpdate}$ 
   ||  $MLL_{v_{ID}} \leq \epsilon_{MLL}$  then
18:         Find  $MLL_{v_{ID}}$  using recent information
19:         Update  $v_{ID}$  entries in  $v_{itable}$ 
20:       end if
21:     end if
22:   end if
23: end if

```

---

**B. Mobility-Based Link Lifetime Estimation**

Finding the minimum link lifetime ( $MLL$ ) between two vehicles based on their mobility information exchanged in beacons is an imperative component within *i*CAR-II. Based on mobility prediction,  $MLL$  is defined as the expected remaining time duration for two communicating vehicles to stay within the communication range of each other before the first possible link breakage occurs due to their mobility, i.e., before the distance between them is predicted to exceed  $R$  meters due to a possible mobility scenario. Many vehicular mobility models can be applied in order to predict  $MLL$ , e.g., car following models [26]. In this paper, we consider a unique prediction model that takes into consideration the actual requirements for *i*CAR-II as a routing protocol, as well as the information available at, or derived from, beacons and routing tables. The  $MLL$  prediction model considers the following factors.

- 1) *Relative location*: It includes the relative distance  $d$  between two communicating vehicles  $v_i$  and  $v_{ID}$ , in addition to the road segments to which  $v_i$  and  $v_{ID}$  belong.  $v_i$  and  $v_{ID}$  can either belong to the same road segment  $R_{v_i}$  or to two adjacent road segments  $R_{v_i}$  and  $R_{v_{ID}}$ . Two adjacent roads have a common intersection and, accordingly,  $R_{v_{ID}}$  can be described to be to the right, in front, or to the left of  $R_{v_i}$ . Thus, at each road, the set of adjacent road segments can be divided into three subsets  $\mathbb{R}$ ,  $\mathbb{F}$ , and  $\mathbb{L}$ , according

to the orientation of  $v_i$  and the common intersection, regardless of the driving direction.

- 2) *Vehicles' speed*: The *predicted effective speed*  $ES_{v_i}$  is introduced in order to mitigate the effect of the frequent change in vehicle's speed and acceleration in the city environment driving pattern. The predicted speed for a vehicle  $v_j$ ,  $ES_{v_j}$  is a function of both the vehicle's average speed in the last  $m$  seconds  $S_{v_j}$  and the average speed of its leading vehicles  $LS_{v_j}$  and depends on the density of leading vehicles in front of it as follows:

$$d_L = \begin{cases} d_I, & \text{if } R > d_I \\ R, & \text{else} \end{cases} \quad (1)$$

$$k_{v_j} = \frac{1}{d_L} \cdot \sum_{i=1}^N l_{v_i} \quad (2)$$

$$LS_{v_j} = \begin{cases} \frac{1}{\sum_{i=1}^N l_{v_i}} \cdot \sum_{i=1}^N S_{v_i} \cdot l_{v_i}, & \text{if } \sum_{i=1}^N l_{v_i} > 0 \\ 0, & \text{else} \end{cases} \quad (3)$$

$$ES_{v_j} = \begin{cases} (1 - \frac{k_{v_j}}{k_J}) \cdot S_{v_j} + \frac{k_{v_j}}{k_J} \cdot LS_{v_j}, & \text{if } k_j < k_J \\ LS_{v_j}, & \text{else} \end{cases} \quad (4)$$

where  $d_I$  is the distance between the vehicle and the next intersection based on its mobility direction,  $d_L$  is the distance that the leading vehicles occupy,  $k_{v_j}$  is the leading vehicles traffic density (vehicle/m), and  $k_J$  is the traffic jam density.

- 3) *Driving direction*: Each vehicle is aware, by the means of beacons, of the driving direction of itself and its neighboring vehicles within the same road segment, i.e., either the same or the opposite driving direction. For neighboring vehicles belong to adjacent road segments, and with respect to their driving direction and common intersection  $I_j$ , the binary variable  $H_v$  is defined as follows:

$$H_{v_{ID}} = \begin{cases} 1, & \text{if } v_{ID} \text{ is heading to } I_j \\ 0, & \text{else.} \end{cases} \quad (5)$$

$H_{v_{ID}}$  information of each neighboring vehicle  $v_{ID}$  that belongs to a different road segment can be maintained in the vehicle's routing table. In addition, turning signal information  $Sig_{v_{ID}}$  gives another key indication for prospective driving direction.

Between two neighboring vehicles  $v_i$  and  $v_{ID}$ , each combination of the previous variables (i.e.,  $R_{v_{ID}}$ ,  $H_{v_{ID}}$ ,  $H_{v_i}$ ,  $Sig_{v_{ID}}$ ,  $Sig_{v_i}$ , and  $f_{v_{ID}}$ ) defines a unique case. Accordingly, *i*CAR-II mobility prediction model defines 144 possible cases. Each case is studied to predict one or more potential mobility scenarios between the communicating vehicles. Then, each scenario is further studied to derive a corresponding equation to obtain  $MLL$ . First, the different scenarios are defined according to the following rules and assumptions.

- 1) For a vehicle  $v_i$ , neighboring vehicles within the same road segment and those belonging to a *front* road segment ( $R_{v_{ID}} \in \mathbb{F}$ ) are considered to be moving in one

dimension; on the other hand, neighboring vehicles belonging to a *right* or *left* road segment ( $R_{vID} \in \{\mathbb{R} \cup \mathbb{L}\}$ ) are considered to be moving in a perpendicular direction to  $v_i$ .

- 2) A neighboring vehicle  $v_{ID}$  within the same road segment that has an idle turning signal maintains its speed of  $ES_{vID}$  for the prediction period.
- 3) Three mobility scenarios are studied for each vehicle  $v_{ID}$  that has an active turning signal: moving in the same driving direction with the speed of  $ES_{vID}$ , stopping at the reported location (waiting to make a turn), and making an instant change of direction according to  $Sig_{vID}$  and moving at the *averaged maximum speed*  $S_a$ .  $S_a$  is a constant that considers an initial speed of 0 m/s and a maximum acceleration, until reaching a maximum speed, for a total travel distance of  $R$  meters.
- 4) When the communicating vehicles  $v_i$  and  $v_{ID}$  belong to different road segments, and  $v_i$  is moving toward the common intersection, two additional scenarios are considered: instant stopping of  $v_i$  (due to a red traffic light) and proceeding of  $v_i$  at the speed of  $S_a$ .
- 5) In scenarios where vehicles move in perpendicular directions or where instant change of vehicle's driving direction is considered, a reduced effective transmission range  $\hat{R}$  is used to represent a nonline-of-sight communication environment.
- 6) When there are more than one mobility scenarios for a certain case, only the scenario/scenarios that can cause earlier communication disconnection is/are considered. If the first disconnection depends on the actual values of the case variables in more than one scenario, equations from the different scenarios are considered, and  $MLL$  takes the minimum result. The predicted  $MLL$  might be obtained from a different scenario than the actual one, or from a misinterpreted turning signal, i.e., an active turning signal for a lane change only. This can result only in a shorter  $MLL$  and is corrected via the frequent  $MLL$  updates, as shown in Algorithm 1, to maintain valid  $MLL$  information.
- 7) The prospective mobility scenario is predicted for a short period of time to ensure the validity of the given mobility information; thus,  $MLL$  is upperbounded by  $R/S_a$ .

The aforementioned rules determine one or more mobility scenarios for each case under consideration. Each scenario is associated with a predicted link lifetime  $t$  between the communicating vehicles. To obtain  $t$ , a corresponding equation to each scenario is derived as follows.

- 1) A diagram for the potential mobility scenario is created; a case example for mobility in two dimensions is presented in Fig. 3 with two potential mobility scenarios.
- 2) According to the aforementioned rules and a certain scenario under consideration, the different variables of the scenario are determined, e.g., using  $R$ ,  $\hat{R}$ ,  $S_a$ ,  $ES_v$ , etc.
- 3) For mobility in one dimension, simple Kinematic equations are used to find  $t$ . For example, for a scenario of two vehicles moving toward each other with predicted speeds  $ES_{v_i}$  and  $ES_{v_{ID}}$ , and with an initial distance  $d$  between

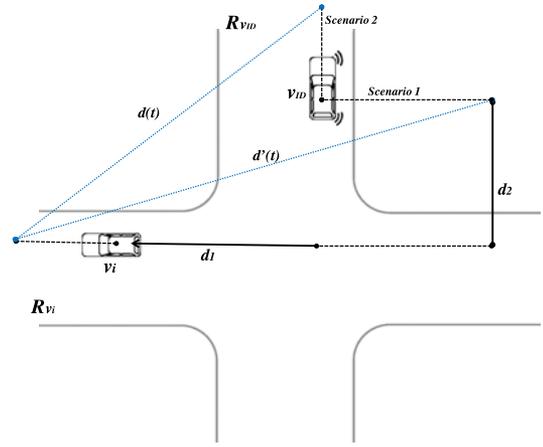


Fig. 3. Example case with two mobility scenarios.

them, we would have

$$t = (R + d)/(ES_{v_i} + ES_{v_{ID}}). \quad (6)$$

- 4) For mobility in two dimensions, the Parametric equations for the predicted trajectory of each vehicle are defined with respect to the parameter  $t$ , i.e., defining  $x_{v_i}(t)$ ,  $y_{v_i}(t)$ ,  $x_{v_{ID}}(t)$ , and  $y_{v_{ID}}(t)$  as functions in time. Then, the Pythagorean theorem is used to find the predicted change in distance between the communicating vehicles

$$d(t) = \sqrt{(x_{v_i}(t) - x_{v_{ID}}(t))^2 + (y_{v_i}(t) - y_{v_{ID}}(t))^2}. \quad (7)$$

By substituting  $\hat{R}$  for  $d(t)$  and solving for  $t$  to find the required link lifetime, we obtain an equation associated with the mobility scenario to predict  $MLL$ . For example, considering *Scenario 1* in Fig. 3, the variables under consideration are  $\hat{R}$ ,  $d_1$ ,  $d_2$ ,  $ES_{v_i}$ , and  $S_a$ . The parametric equations for this scenario are  $x_{v_i}(t) = -d_1 - ES_{v_i}t$ ,  $y_{v_i}(t) = 0$ ,  $x_{v_{ID}}(t) = S_a t$ , and  $y_{v_{ID}}(t) = d_2$ . By applying the Pythagorean theorem

$$d(t) = \sqrt{(-d_1 - ES_{v_i}t - S_a t)^2 + (-d_2)^2}. \quad (8)$$

Replacing  $d(t)$  by  $\hat{R}$  and solving for  $t$  in the case that  $\hat{R} \geq d$

$$t = \begin{cases} \frac{-d_1 + \sqrt{\hat{R}^2 - d_2^2}}{S_a + ES_{v_i}}, & \hat{R} \geq d \\ 0, & \hat{R} < d. \end{cases} \quad (9)$$

Similarly, different scenarios have been studied for the different cases and a set of equations have been determined. When a vehicle  $v_i$  needs to update the value  $MLL_{v_{ID}}$  in  $v_{iTable}$  upon receiving a beaconing message from  $v_{ID}$ ,  $v_i$  determines the mobility case based on the available information and calculates the predicted link lifetime  $t$ . When more than one scenarios are considered, the minimum value of  $t$  is maintained. Then, the  $MLL$  between  $v_i$  and  $v_{ID}$  is updated in  $v_{iTable}$

$$MLL_{v_{ID}} = \min\left(t, \frac{R}{S_a}\right). \quad (10)$$

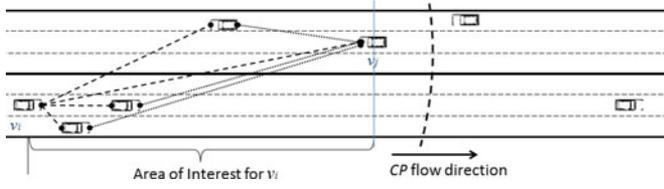


Fig. 4. Calculating  $RLL$  in road segment evaluation.

$MLL_{v_{ID}}$  is updated frequently at  $v_{i_{Table}}$  with respect to three criteria, as shown in Algorithm 1:

- 1) periodically with a period of  $\tau_{LinkUpdate}$ ;
- 2) if a major change in  $v_{ID}$ 's predicted speed has been detected; and
- 3) if  $v_i$  is receiving beacon messages, with acceptable  $RSSI_{v_{ID}}$ , after, or close to, the expiry time of the expected  $MLL_{v_{ID}}$ .

### C. Road Segment Connectivity Evaluation

RSE is a heuristic procedure dynamically initiated by some vehicles to sense the different parts of the network and update the NSI. NSI includes road segment connectivity to infrastructure (RSU) status, the best route to infrastructure, the expected PDD via that route, and the expiry time of it. NSI is shared locally within a road segment and exchanged via beacon messages. In RSE, a light-weight CP traverses the road segment via relaying forwarders and collects connectivity and link lifetime information at each intermediate forwarder. When reaching the target intersection  $I_{ID}$ , a vehicle  $v_j$  at  $I_{ID}$  reports the connectivity status of  $R_{ID}$  and its predicted  $MLL$  road-level link lifetime ( $RLL_{R_{ID}}$ ) to  $LCs$  via LTE and obtains an updated  $NSI_{R_{ID}}$  accordingly.

RSE procedure is initiated, with a probability  $P$ , by a CP originator  $v_i$  entering a road segment  $R_{ID}$  toward an intersection  $I_{ID}$ . The principle of RSE is predicting an  $MLL$  per road segment based on link lifetime information,  $MLLs$ , between individual vehicles available at their routing tables. RSE divides  $R_{ID}$  into smaller vicinities, or areas of interest ( $AoIs$ ), between CP forwarders as shown in Fig. 4. While passing CP, each forwarder finds the maximum link lifetime between itself and the previous forwarder, directly or via one-hop relay vehicle. The originator and each forwarder  $v_{lf}$  attach in CP the set  $\mathbb{L}_{v_{lf}}$  that includes  $MLLs$  values for all neighbors in its  $AoI$  along with their identifiers' set  $\mathbb{M}_{v_{lf}}$ . A receiver forwarder  $v_{cf}$  extracts the set of common neighbors  $\mathbb{C}_{v_{lf}, v_{cf}}$  and finds the maximum possible link between itself and the last forwarder  $RLL_{v_{lf}, v_{cf}}$ , as indicated below:

$$\hat{l}_{\max} = \max_{v_i \in \mathbb{C}} (\min(MLL_{v_{lf}, v_i}, MLL_{v_{cf}, v_i})) \quad (11)$$

$$RLL_{v_{lf}, v_{cf}} = \max(\hat{l}_{\max}, MLL_{v_{lf}, v_{cf}}). \quad (12)$$

Intermediate forwarders update  $\mathbb{M}$  and  $\mathbb{L}$  while keeping only the minimum value of  $RLL_{R_{ID}}$ . The last CP receiver  $v_j$ , which is the closest to  $I_{ID}$ , reports the total delivery delay of CP,  $D_{R_{ID}}$ , along with  $RLL_{R_{ID}}$ , to  $LCs$ .  $LCs$  update the network

graph, find the route(s) to  $RSU(s)$ , and send  $NSI_{R_{ID}}$  back to  $v_j$ . Then,  $v_j$  unicasts the updated NSI to CP originator and broadcasts it via its beacons. Every vehicle within  $R_{ID}$  updates  $NSI_{R_{ID}}$  and includes it in its beacons.

As greedy routing without store-carry-forward is used to deliver CP, reaching  $I_{ID}$  indicates local network connectivity at  $R_{ID}$  for a period of time registered in CP. The delivery delay of CP also gives an indication of PDD in the road as it experiences similar transmission and queuing delay in addition to interference and fading conditions in  $R_{ID}$ . For a disconnected road segment, CP is dropped when a forwarder, or an originator,  $v_i$  fails to find a next forwarder.  $v_i$  creates an NSI indicating dis-connectivity with a small random validity period, which works as a back-off time to prevent multiple RSE calls by vehicles entering  $R_{ID}$ .

When  $v_i$  enters  $R_{ID}$ , it is expected to receive an NSI from its neighbors, which includes the expiry time of NSI. To ensure the availability of a valid NSI,  $P$  is designed to be a function of the remaining validity time of  $NSI_{R_{ID}}$ ,  $t_{rem}$ , and  $R_{ID}$  length  $|R_{ID}|$ . When  $v_i$  receives no valid NSI, it also initiates the RSE procedure. In the following, we present one way to design  $P$  [7]:

$$P = \begin{cases} e^{-\frac{t_{rem}-C}{2}}, & t_{rem} \geq C \\ 1, & t_{rem} < C \end{cases} \quad (13)$$

$$C = 2 \cdot t_{\max} \cdot \lceil |R_{ID}|/R \rceil + \epsilon \quad (14)$$

where  $t_{\max}$  and  $\epsilon$  are design constant parameters representing the maximum acceptable delay per forwarder, including average transmission delay and queuing delay, and the expected time to obtain NSI from  $LC$ , respectively.

### D. City-Level Network Topology and Data Routing

The frequent distributed calls of the RSE procedure and the associated connectivity and delay information sent to  $LCs$  enable  $LCs$  to draw a real-time network graph providing a city-level network topology awareness, where the graph consists of vertices, representing road intersections, and weighted edges, representing road segments where each edge is weighted by the experienced delay. As  $LCs$  receive RSE update messages for only connected roads, the graph represents only real-time network view of the map, and edges with expired validity lifetime can be removed. With a known set of  $RSUs$  locations in a city, each road segment has a subset of nearby  $RSUs$ ; thus, after receiving an RSE update message related to a certain road  $R_{ID}$ , a shortest path algorithm, e.g., Dijkstra, is run on the subgraph of the network that has the road segment  $R_{ID}$  and the subset of nearby  $RSUs$  to find the best route to the core network.  $LCs$  send back a response message to the sender, which has an NSI. Then, the sender broadcasts the NSI in  $R_{ID}$  via beacons, which enables connectivity awareness to all vehicles in the vicinity of  $R_{ID}$ . NSI includes the path, by the means of intersections, the path's lifetime, which is the minimum  $RLL_{R_{ID}}$  among road segments constructing the path, and the expected delivery delay, which is the summation of experienced delivery delay for road segments constructing the path.

**Algorithm 2:** Next-hop Selection.

---

**Require:**  $v_{cf_{table}}$ ,  $R_{vcf}$ ,  $I_{Crnt\_Target}$ ,  $I_{Nxt\_Target}$ ,  $RSSI_{threshold}$

```

1: for  $z=1$  to  $|\mathbb{N}|$  do           (check all neighbors)
2:   if  $R_{vz} == R_{vcf} || I_{Nxt\_Target} \in \mathbb{I}_{R_{vz}}$  then
3:     if ( $v_{cf}$  moving towards  $I_{Crnt\_Target}$  &  $f_{vz} == 1$ )
       || ( $v_{cf}$  moving away from  $I_{Crnt\_Target}$  &  $f_{vz} == 0$ )
       ||  $I_{Nxt\_Target} \in \mathbb{I}_{R_{vz}}$  then ( $v_z$  makes forwarding progress)
4:       if  $RSSI_{vz} \geq RSSI_{threshold}$  then
5:          $\mathbf{N} = \mathbf{N} \cup v_z$  ( $v_z$  is a potential forwarder)
6:       end if
7:     end if
8:   end if
9: end for
10: if  $\mathbf{N} \neq \emptyset$  then
11:   Find  $v_{nf}$  s.t.  $d_{v_{nf}} = \max(d_{v_z} \forall v_z \in \mathbf{N})$ 
12: else
13:   Next-forwarder is not found (packet will be
       dropped)
14: end if

```

---

According to the direction of data forwarding, either toward RSU or a destination vehicle, data routing can be described as uplink routing or downlink routing. For uplink routing, vehicles that have data to send find connectivity and expected delay information available in NSI. According to this information, vehicles either use VANETs, LTE, or reschedule transmission for better VANETs conditions. In the case of a connected network, the path from a source road segment to a destination RSU is predetermined by the means of consecutive intersections. Thus, *iCAR-II* deploys source PBR where the path is attached to the header of each packet, which reduces cost, delay, and overhead of multiple route enquiries via LTE. In case a packet has reached a disconnected road, a forwarder can encapsulate the packet and forward it via a new path using a more recent NSI available at its road segment, if any, otherwise the packet is dropped. Disconnection can occur due to an unexpected delivery delay beyond the path lifetime, or an unexpected local network disconnection in the routing path during its lifetime. On the other hand, vehicle's location, an associated RSU, and the path from RSU to the vehicle, by the means of intersections, are determined by LCs in the downlink routing case. Data packets are forwarded from the core network to the RSU, and VANETs data routing takes place from RSU to the destination vehicle using source PBR.

For routing within roads, *iCAR-II* uses a greedy-based next-hop selection method. Algorithm 2 shows a light-weight next-hop selection procedure to filter one-hop neighbors based on their location and the latest received *RSSI*. The location filter in the forwarding process aims to maximize the progress toward the target intersection. Such greedy forwarding protocol selects next-forwarders that are farther from a sender, which are more likely to leave the communication range causing transmission interruption, or have bad signal quality. Thus, *RSSI* filter excludes neighbors with *RSSI* below a certain threshold. A vehicle's mobility direction is not considered in order to

maximize the number of potential forwarders, taking into consideration that vehicle's mobility can be negligible compared to data transmission speed, and the distance between vehicles are updated frequently on routing tables.

While forwarding data packets, a next forwarder  $v_{nf}$  is chosen only from the current road segment or from the road segment connecting to the next target intersection  $I_{Nxt\_Target}$  in the packet's path. When a packet reaches the last road segment in its path, each forwarder  $v_{cf}$  looks up the packet's destination in its routing table. In Algorithm 2,  $\mathbb{I}_{R_{v_z}}$  defines the set of two intersections that bounds the road segment  $R_{v_z}$ .

## V. PERFORMANCE EVALUATION

This section presents the evaluation of *iCAR-II*. First, the individual components of *iCAR-II* are considered in brief analysis and discussion. Then, the overall performance is evaluated using a special MATLAB-based simulation program developed to evaluate VANETs routing protocols performance. In addition to *iCAR-II*, we considered three other VANETs routing protocols, which have been slightly modified in order to have a fair comparison with *iCAR-II*, i.e., having the same infrastructure resources. In the following, we present the evaluation methodology, the simulation setup, the evaluation metrics for comparing the different protocols, the protocol components analysis, and the analysis of simulation results.

### A. Evaluation Methodology

We first study the different components of *iCAR-II*, namely

- 1) beaconing and neighborhood awareness;
- 2) node-level link lifetime;
- 3) road-level connection lifetime;
- 4) city-level network connectivity;
- 5) location service and data routing.

Then, we compare the performance of *iCAR-II* with following three common PBR protocols;

- 1) GPSR [18], which is a MANET protocol used widely as a performance benchmark for geographic routing protocols;
- 2) GSR [14], which is a VANETs PBR protocol and it uses source intersection-based routing;
- 3) GyTAR [21], which is another PBR VANETs intersection-based routing protocol that takes into consideration vehicular traffic. GyTAR has been also deployed widely for comparing routing performance in the VANETs context.

Those protocols are modified to use LTE channels to report vehicles location periodically and acquire the location of the destination, or the closest RSU, from LCs.

### B. Simulation Setup

VANETs city scenario has been implemented in MATLAB to represent 7000 m  $\times$  7000 m grid area of bidirectional roads. Roads vary in terms of length, width, and vehicles density to represent major roads and residential areas in the city. Each road segment has a predefined maximum speed. Fig. 5 shows approximate map that represents the roads grid that has a total of 165 intersections, with 45 of them being traffic-light controlled.

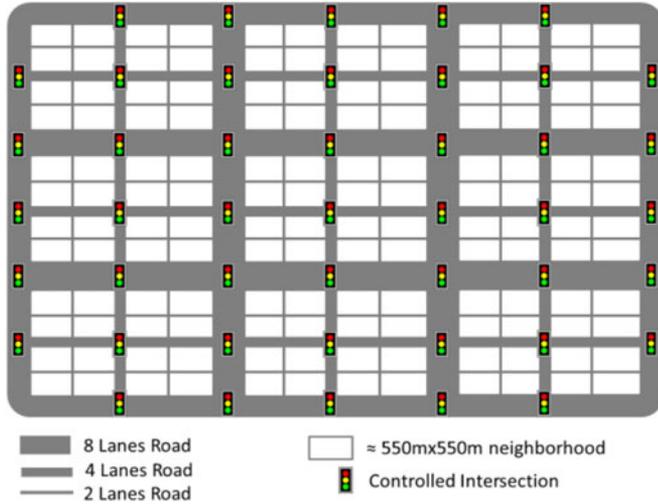


Fig. 5. Approximate simulation map.

TABLE I  
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
Scenario Duration	40 s	$RSSI_{\text{threshold}}$	$0.6 \times RSSI_{\text{max}}$
Scenario Repetition	5–12 times	$\tau_{\text{Bt}}, \tau_{\text{deletrow}}$	
$R$	250 m	$\tau_{\text{update}}, \tau_{\text{fupdt}}, \tau_{\text{LinkUpdate}}, m$	3 s
$\hat{R}$	150 m	$\epsilon_{\text{vel}}$	7 m/s
Packet Size	512 Byte	$S_a$	15 m/s
Transmission Rate	12 Mb/s	$K_J$	0.4 vehicle/m
Packet Lifetime	1500 ms	$C$	1.5 s

The open-source microscopic vehicular traffic generator SUMO [27] is used to generate vehicles movement files. SUMO uses car-following model and the input of our grid map including the number of lanes and traffic densities.

For wireless consideration, a simple DCF MAC is applied for MAC contention, a FIFO packet queue, such as the ac queues design for WAVE's MAC layer [28], is implemented for packet buffering, and a free space model with urban area path loss exponent [29],[30] is deployed for  $RSSI$  estimation. Source vehicles are randomly selected, where source vehicles are always 10% of the total number of vehicles for the different vehicles density scenarios. Each source vehicle continuously sends data packets to the core network via RSUs, where packets are routed independently. LTE channels are assumed to have ideal communication and represented by a fixed delay of 200 ms for one-way communication. Fetching information from LCs is also represented by a fixed delay of 500 ms. Each simulation scenario has been repeated several times for accurate results. Table I presents the general simulation parameters used in this evaluation.

### C. Evaluation Metrics

The performance evaluation of the routing protocols has considered variable network density, packet generation rate, and

the number of deployed RSUs. The performance metrics are as follows.

- 1) Packet delivery ratio (PDR): We define two forms of PDR to show the ability of a routing protocol to successfully transfer data from a source to a destination on an end-to-end basis: a) PDR1: number of successfully received data packets by destinations per number of sent data packets per sources and b) PDR2: number of successfully received data packets by destinations per the total number of data packets sent, or ready to be sent, at sources.
- 2) Average PDD: This metric shows the latency of data packet delivery introduced by each routing protocol and defined as the average end-to-end delivery delay of all successfully delivered data packets.
- 3) Average routing overhead: This metric shows the extra communication overhead required by routing protocols. Two types of routing overhead are defined: a) average LTE routing messages, e.g., location updates and enquiry messages, per second, and b) average unicast routing CPs which is the average of extra unicast packets sent by vehicles to maintain the routing protocol per second per road segment.

### D. Simulation Results

1) *Beaconing and Neighborhood Awareness*: Beaconing is one of the main components in any PBR protocol. Beaconing rate can be either fixed or dynamic with respect to speed, vehicle density, or other parameters. Since we have considered a fixed beaconing rate in this study, any other beaconing scheme is still valid as long as it enables *i*CAR-II to predict  $MLL$ s and exchange NSIs. *i*CAR-II considers two strategies to reduce the computation and communication overhead that can occur to update  $MLL$ s and share NSIs. First, instead of updating  $MLL$  value for each neighboring vehicle upon receiving its beacon, *i*CAR-II reduces the number of  $MLL$  updates for each neighbor per second by the factor of  $1/\tau_{\text{LinkUpdate}}$ . Simulation shows that the effect of updating  $MLL$  values on the *i*CAR-II performance is negligible if the  $\tau_{\text{LinkUpdate}}$  is less than 3 s. Second, *i*CAR-II uses beacons to share NSIs in order to preserve the bandwidth.

Fig. 6 shows the delay required to deliver NSI to all vehicles within a road segment of 1 km length and 4 lanes width. It is shown that NSI distribution time is generally decreased by the increased vehicular density and/or beaconing rate. In light and moderate traffic densities, increasing vehicular density or beaconing rate significantly decrease the delay to deliver NSI. However, in dense areas, and with high static beaconing rate, the delivery of beaconing packets is delayed due to packets' collisions, or rescheduling, causing slightly delayed NSI delivery. Thus, NSI delivery delay in such situations highly depends on the performance of the deployed MAC protocol. In general, results in Fig. 6 show acceptable delay taking into consideration that (13) and (14) preserve time for NSI delivery.

2) *Node-Level MLL*: The  $MLL$  finding procedure predicts the worst possible case scenario for future movement of two neighboring vehicles based on their mobility vectors, distance

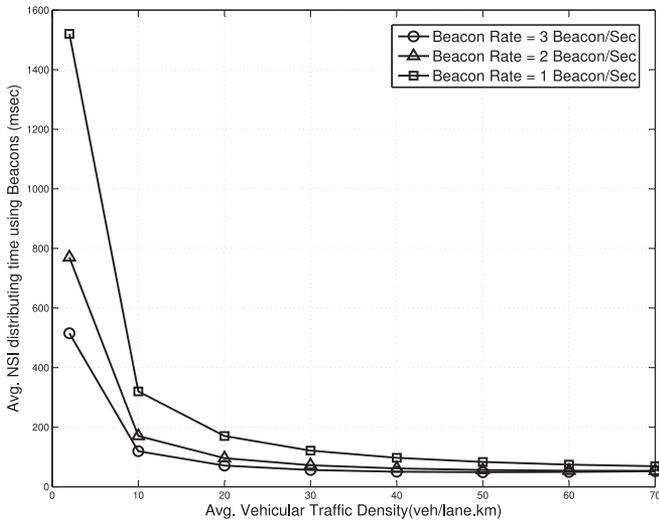


Fig. 6. Average delay for network information dissemination within a road segment using beacons.

between them, distance to a common intersection, and their turn-signal status, in order to assign a lower-bound of link lifetime between them.  $MLL$  is frequently updated, while vehicles are exchanging beacons, every  $\tau_{LinkUpdate}$ . Simulation results show that this procedure succeeds in putting a lower bound of link lifetime in all cases. In other words, it successfully predicts link breakage before it happens. However, in real-life situations, other cases can occur causing link breakage within the predicted minimum lifetime. For example, a parking vehicle on the side of the road can have a high  $MLL$  value; however, it is more likely to turn-off its OBU causing a communication termination. Such situations represent a small percentage and can be ignored.

Equations (1)–(4) present the expected speed  $ES$  of vehicles based on its average speed and the average speed of its leading group. Differentiating between leading vehicles based on their turn-signal status makes  $ES$  more accurate. The mean percentage error of  $ES$  prediction, with  $\tau_{LinkUpdate}$  defined in Table I, is 4.3%. However, with large  $\tau_{LinkUpdate}$  value, this mean increases significantly, considering urban scenario with controlled intersections where drivers change their speed frequently due to traffic lights status. To avoid the effect of such error when using larger  $\tau_{LinkUpdate}$  values, Algorithm 1 calls  $MLL$  procedure when a major change in speed is detected.

3) *Road-Level Minimum Connection Lifetime*: Minimum road connectivity lifetime algorithm is a heuristic algorithm that uses the link lifetime information available at nodes' routing tables to assign a minimum  $RLL$  to each road segment. First, one possible routing path is considered to check instantaneous connectivity, and then, one-hop relay between each pair of consecutive forwarders in the path is considered to predict future connectivity. Among each pair, the maximum predicted link lifetime is selected, and among the selected set, the  $MLL$  is considered to be the  $RLL$ . Intuitively, the road segment has at least one connected path from end-to-end during  $RLL$  second. More than one initial path can be considered, and more than one-hop possible intermediate relay can be calculated, which

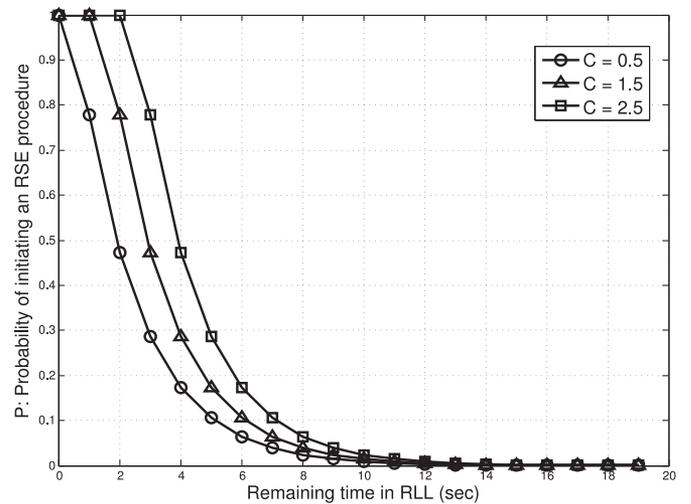


Fig. 7. Probability of initiating RSE procedure for a vehicle entering the road segment.

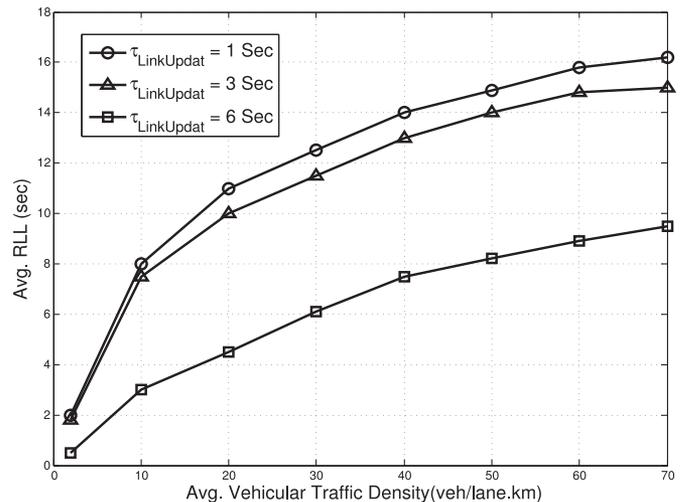


Fig. 8. Average  $RLL$  in a road segment.

increase the predicted  $RLL$ . However, this increase comes at the cost of communication overhead to share more than one-hop neighboring  $MLL$  information, and calculation overhead to find all possible future links among those vehicles. However,  $iCAR-II$  considers only the previously calculated one-hop  $MLL$ , available at vehicles' routing tables, along with dynamic updating procedure using  $P$  in (13).  $P$  is able to maintain valid  $RLL$ , while the road segment has a connected path. It takes into consideration the time required to generate  $RLL$  and obtain and distribute  $NSI$ , before the expiry time of the current one.

Fig. 7 shows the probability of initiating an RSE call by a vehicle entering the road segment as a function of  $RLL$ 's remaining time considering different values of  $C$ .  $C$  is a design parameter related to road length, transmission range, maximum acceptable transmission time per one-hop, and the time required to obtain  $NSI$  from LCs. Fig. 8 shows the average reported  $RLL$  values with respect to different vehicles density. It shows that even with low vehicles density, roads can maintain connected

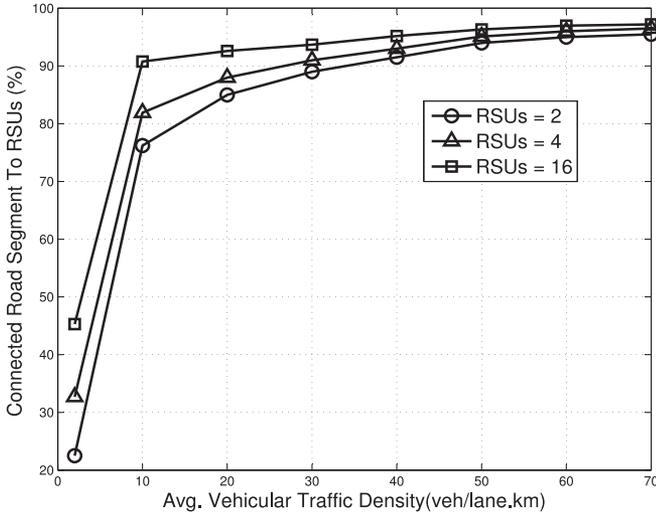


Fig. 9. Percentage of connected road segments to the core network using *i*CAR-II.

paths for a considerable duration of time, and RSE procedure enables source vehicles to instantaneously utilize these paths. Moreover, the results show that the average  $RLL$  is directly proportional to vehicles density within road segment. This can be related to the decrease in average vehicles speed in the high density scenario, as well as the availability of more intermediate nodes between each pair of CP forwarders.  $RLL$  is also inversely proportional to  $\tau_{LinkUpdate}$  as with large  $\tau_{LinkUpdate}$  values, the remaining time of  $MLLs$  decrease before updates, and  $RLL$  decreases accordingly.

4) *City-Level Network Connectivity*: *i*CAR-II is a proactive protocol that enables vehicles to have immediate global network condition information by making NSIs available at vehicles' beacons. Vehicles, via NSI, can know about the road connectivity to the core network, the route to an RSU, the expected delivery delay, and the expiry time of that route. Routes are dynamically updated on LCs by probabilistically initiating RSE procedures among different network edges. LCs maintain updated network values as (13) and (14) ensure that.

Fig. 9 shows the percentage of the connected road segments to the infrastructure with respect to different network node densities and number of deployed RSUs. It can be seen that *i*CAR-II can construct connected networks, even with low deployment of RSUs. This can be related to the global view of connected road segments at LCs. With respect to road segments length, number of lanes per road segment, and the transmission range under consideration, it is shown that the number of connected road segments to the core network is increasing rapidly with the increase in vehicular density in the light traffic densities test points (less than 10 vehicles/lane-km) as the network connectivity becomes more sensitive to vehicular densities in this range. With higher vehicular densities, the increase becomes slower as most main roads are already connected to RSUs, and only a few roads are joining the network when the number of vehicles increases.

5) *Packet Delivery Ratio*: As packets are transmitted by source vehicles using *i*CAR-II only when connected path is

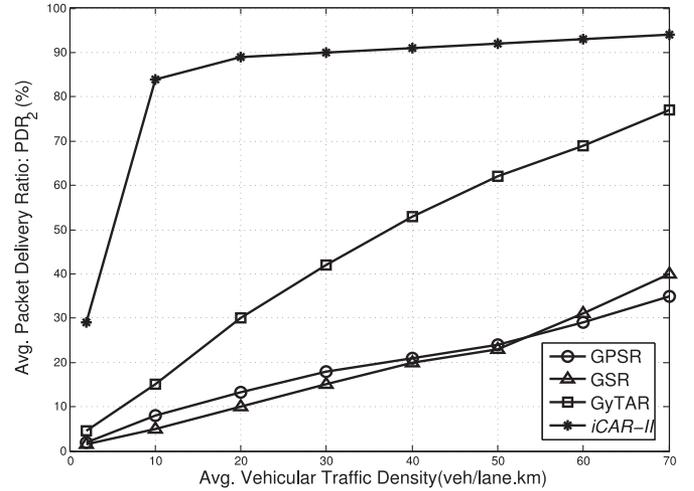


Fig. 10. Average PDR ( $PDR_2$ ) using 4 RSUs and  $PGR = 10$  packet/s.

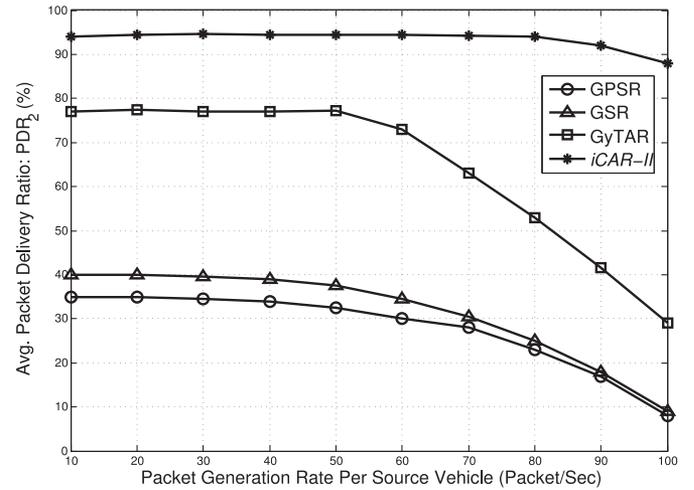


Fig. 11. Average PDR ( $PDR_2$ ) using 4 RSUs and vehicle density of 70 vehicle/lane-km.

detected, the ratio of delivered data packets to the sent packets ( $PDR_1$ ) is expected to be high regardless of network node density. With  $PGR = 10$  packets/s and the deployment of four RSUs, simulation shows that  $PDR_1$  always exceed 97%. Data packets that have not been delivered during the lifetime of the path might be dropped due to an expected network disconnection. Also, packets that have not been delivered during their lifetime due to delivery delay are dropped. Notice that *i*CAR-II conserves network bandwidth by buffering data packets when VANETs is not connected to the core network.

In order to have a valid performance comparison between *i*CAR-II and GPSR, GSR, and GyTAR, which do not require prior determination of path existence before transmission, we define  $PDR_2$  to be the ratio of packets successfully delivered to packets that are ready to be sent. Figs. 10 and 11 show that *i*CAR-II still has a significantly higher  $PDR_2$  than the other PBR protocols. Fig. 10 shows that increasing vehicles density, with a low data packet traffic in the network, improves PDR, as

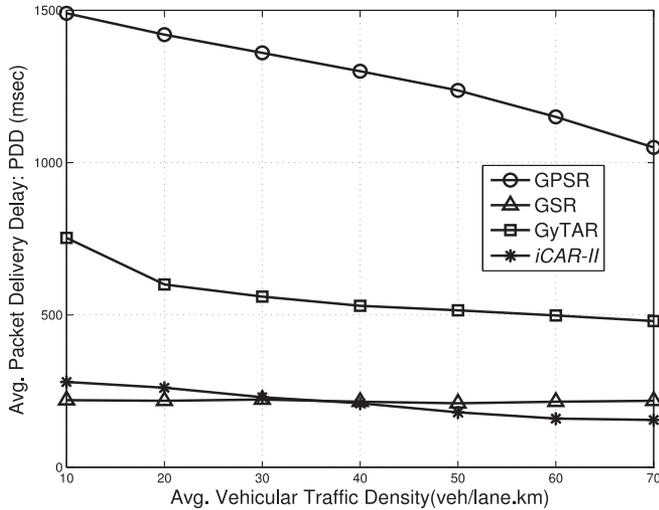


Fig. 12. Average PDD using four RSUs and PGR = 10 packet/s.

VANETs become more connected. With low vehicles densities, GPSR, GSR, and GyTAR show a very low PDR2 as they blindly route data packets through an intermitted network, while *iCAR-II* has a noticeably high PDR2 due to its connectivity awareness feature. The curve trend of *iCAR-II* is analogous to the network connectivity curve in Fig. 9. Data packets might be routed along paths that are not the shortest curvemetric routes yet connected. It is observed that GSR performs better than GPSR only in high vehicular density, when VANETs are connected, as GSR does not consider vehicular traffic in the routing decision. In high vehicular densities, GPSR suffers from higher routes length compared to other protocols as it does not use map information or anchor routing. In such cases, GPSR packets reach their expiry time before delivery.

Fig. 11 shows PDR2 of the different protocols with respect to a variable PGR and a high vehicular density. Results show that *iCAR-II* maintains high performance, even with an increasing PGR. As *iCAR-II* considers delivery delay per road segment in its route constructing level, new paths are dynamically suggested by LCs to maintain network performance. On the other hand, GSR does not consider dynamic routing, while GPSR and GyTAR considers only local connectivity and distance to destination, which result in routing convergence to dense roads, which causes data traffic congestions and high queuing delay. Delay is associated with PDR as delayed packets can reach their expiry time before delivery and be dropped. The slight dropping in *iCAR-II* PDR2 shown in Fig. 11, with high PGR, is due to reaching the communication capacity of RSUs.

6) *Packet Delivery Delay*: As *iCAR-II* considers packet experienced delivery delay of CPs a major metric in route calculation, average PDD using *iCAR-II* is expected to be low. Simulation results show that *iCAR-II* significantly reduces PDD compared to other routing protocols as shown in Figs. 12 and 13. With low vehicle densities, *iCAR-II* selects connected paths even with long trajectories to achieve higher PDR with the cost of slightly high PDD. PDD of GSR is analogous to that of *iCAR-II* in the case of light data traffic as packets are routed

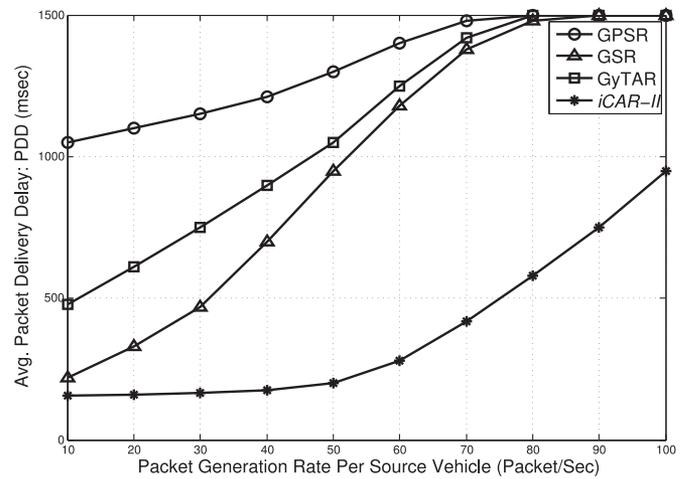


Fig. 13. Average PDD using four RSUs and vehicle density of 70 vehicle/lane-km.

along predetermined paths. However, these paths are either connected to the core network or the packets are dropped, causing very low GSR-PDR, as shown in Fig. 10.

Simulation results also show that PDD in GPSR is high. Packets forwarded using GPSR encounter a high number of intermediate forwarders due to the perimeter recovery routing strategy and the experience of long routing paths. Moreover, increasing PGR leads to a significant PDD increase in GPSR, GSR, and GyTAR, as shown in Fig. 13. These protocols do not consider delivery delay in its routing, and when routes converge to a limited number of roads, data traffic congestion increases the delivery delay. It can be shown from Figs. 11 and 13 that a considerable portion of data packets have been dropped due to reaching their expiry lifetime, which is set to be 1500 ms in our study. As *iCAR-II* uses dynamic route selection considering the experienced delivery delay, it has a significantly reduced PDD.

7) *Routing Overhead*: We consider the additional routing control messages to measure and compare the introduced overhead by the different routing protocols. These CPs can be classified into three categories:

- 1) beaconing messages;
- 2) LTE routing messages;
- 3) unicast routing control messages.

Beacons are the main communication overhead introduced by any PBR protocol, as the broadcast beaconing messages use control channels periodically. However, beacons are required by safety applications, and as long as different protocols use the same interbeacon interval/beaconing protocol, the effect of beacons on the networks is the same for the different protocols.

LTE communication overhead is an important evaluation metric, as accessing LTE channels cost more than VANETs DSRC channels. Fig. 14 shows the simulation results of average LTE control messages used for each vehicular density scenario. In GSR and GyTAR, LTE routing messages are used to report entering new road segments (location updates) and to enquire about a destination location. *iCAR-II* has a slightly higher average of LTE control messages as it uses LTE channels to update

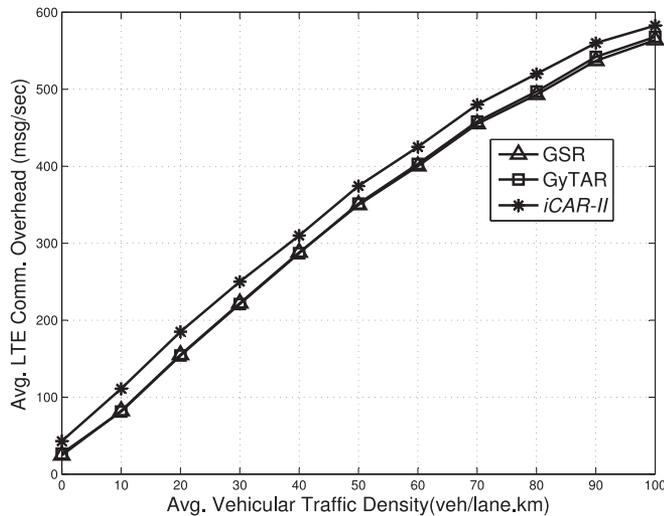


Fig. 14. LTE routing-control messages.

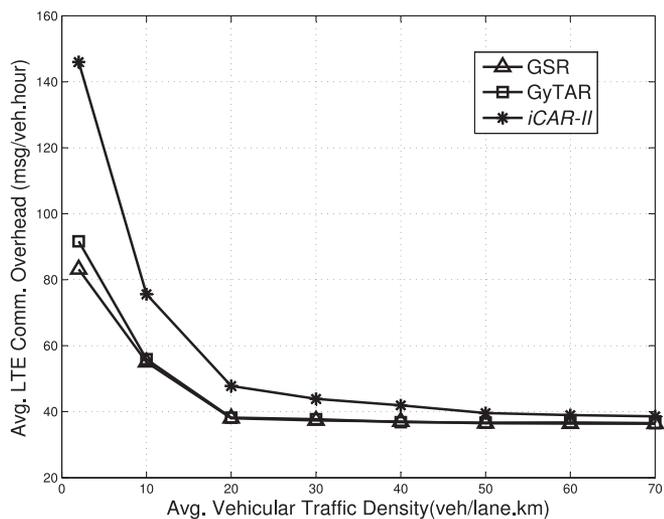


Fig. 15. Average LTE routing messages per vehicle per hour.

road condition information. In GPSR, vehicles use LTE channels for location updates and location inquiry messages. The average LTE communication overhead, when GPSR is deployed, depends on the location update period, and it is always higher than the average overhead generated by the other protocols. This shows one of the advantages of intersection-based routing.

To better understand the LTE communication overhead, Fig. 15 shows the average number of LTE control messages sent by each vehicle per hour. It is shown that *i*CAR-II introduces higher LTE overhead especially in the sparse network cases. In fact, the increase of LTE overhead in *i*CAR-II, as compared to the other protocols, is still insignificant with respect to the increase in PDR shown in Fig. 10. In the worst case, the difference is about 60 messages per vehicle per hour, which is a small cost to obtain connectivity information in a sparse network for data offloading or VANETs Internet access. With higher vehicle densities, roads become more connected with higher average *RLL* and fewer *LCs* updates accordingly.

The third type of communication overhead for routing control is the unicast packets sent locally within roads to collect traffic information in GyTAR and to examine connectivity, collect links lifetime, and calculate delivery delay in *i*CAR-II. Although *i*CAR-II introduces about double the number of these CPs compared to GyTAR, this overhead can be neglected as these packets are unicast, distributed, and, in the worst case, the average number of CPs does not exceed two packets per second for each road segment.

## VI. CONCLUSION

In this paper, an efficient framework *i*CAR-II has been introduced to integrate VANET, cellular network, and *LCs*, in order to improve VANETs data routing and enable cellular network mobile data offloading. *i*CAR-II enables mobile users to proactively obtain VANET connectivity to the core network information. The availability of this information preserves VANET's bandwidth, in the cases of disconnectivity or data traffic congestions, and enables users to enjoy the low-cost VANET-based Internet access and mobile data offloading. *i*CAR-II utilizes the reliable communication channel of cellular network to construct a global real-time view of VANET's topology. It has been demonstrated that *i*CAR-II algorithms can provide real-time VANET information to mobile users, and an efficient and dynamic data routing, with a limited use of LTE messages per vehicle. With respect to the current framework, *RSUs* placement can be optimized in order to increase VANETs connectivity to the core network and reduce the cost of *RSUs* deployment, which will be considered in our future work.

## REFERENCES

- [1] T. H. Luan, X. Ling, and X. Shen, "MAC in motion: Impact of mobility on the MAC of drive-thru Internet," *IEEE Trans. Mobile Comput.*, vol. 11, no. 2, pp. 305–319, Feb. 2012.
- [2] W. L. Tan, W. C. Lau, O. Yue, and T. H. Hui, "Analytical models and performance evaluation of drive-thru Internet systems," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 1, pp. 207–222, Jan. 2011.
- [3] H. Zhou *et al.*, "Chaincluster: Engineering a cooperative content distribution framework for highway vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2644–2657, Dec. 2014.
- [4] N. Alsharif, S. Cespedes, and X. Shen, "*i*CAR: Intersection-based connectivity aware routing in vehicular ad hoc networks," in *Proc. IEEE Int. Conf. Commun.*, 2013, pp. 156–161.
- [5] S. Cespedes, S. Taha, and X. Shen, "A multihop-authenticated proxy mobile ip scheme for asymmetric VANETs," *IEEE Trans. Veh. Technol.*, vol. 62, no. 7, pp. 3271–3286, Sep. 2013.
- [6] W. Viriyasitavat, F. Bai, and O. Tonguz, "Toward end-to-end control in VANETs," in *Proc. IEEE Veh. Netw. Conf.*, 2011, pp. 78–85.
- [7] N. Alsharif and X. Shen, "*i*CARII: Intersection-based connectivity aware routing in vehicular networks," in *Proc. IEEE Int. Conf. Commun.*, 2014, pp. 2731–2735.
- [8] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: A survey," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 148–157, May 2013.
- [9] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 20132018," 2014. [Online]. Available: [http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white\\_paper\\_c11-520862.html](http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html)
- [10] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop Mobile Comput. Syst. Appl.*, 1999, pp. 90–100.
- [11] D. B. Johnson, D. A. Maltz, and J. Broch, "The dynamic source routing protocol for multihop wireless ad hoc networks," in *Ad Hoc Networking*. Boston, MA, USA: Addison-Wesley Longman, 2001, pp. 139–172. [Online]. Available: <http://dl.acm.org/citation.cfm?id=374547.374552>

- [12] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *IEEE Veh. Technol. Mag.*, vol. 2, no. 2, pp. 12–22, Jun. 2007.
- [13] C. A. T. H. Tee and A. C. R. Lee, "Survey of position based routing for inter vehicle communication system," in *Proc. IEEE Distrib. Framework Appl.*, 2008, pp. 174–182.
- [14] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve, "A routing strategy for vehicular ad hoc networks in city environments," in *Proc. IEEE Int. Veh. Symp.*, 2003, pp. 156–161.
- [15] H. Omar, W. Zhuang, and L. Li, "Gateway placement and packet routing for multihop in-vehicle internet access," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 3, pp. 335–351, Sep. 2015.
- [16] N. Cheng, N. Lu, N. Zhang, X. Shen, and J. W. Mark, "Vehicular WiFi offloading: Challenges and solutions," *Veh. Commun.*, vol. 1, no. 1, pp. 13–21, 2014.
- [17] G. El Mouna Zhioua, H. Labiod, N. Tabbane, and S. Tabbane, "VANET inherent capacity for offloading wireless cellular infrastructure: An analytical study," in *Proc. IEEE 6th Int. Conf. New Technol. Mobility Security*, 2014, pp. 1–5.
- [18] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th ACM MobiCom*, 2000, pp. 243–254.
- [19] B.-C. Seet, G. Liu, B.-S. Lee, C.-H. Foh, K.-J. Wong, and K.-K. Lee, "A-star: A mobile ad hoc routing strategy for metropolis vehicular communications," in *Proc. Networking*, 2004, pp. 989–999.
- [20] R. Tavakoli and M. Nabi, "Tiger: A traffic-aware intersection-based geographical routing protocol for urban VANETs," in *Proc. IEEE 77th Veh. Technol. Conf.*, 2013, pp. 1–5.
- [21] M. Jerbi, S. Senouci, T. Rasheed, and Y. Ghamri-Doudane, "Towards efficient geographic routing in urban vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 5048–5059, Nov. 2009.
- [22] X. Wang, C. Wang, G. Cui, and Q. Yang, "Practical link duration prediction model in vehicular ad hoc networks," *Int. J. Distrib. Sensor Netw.*, vol. 2015, 2015, Art. no. 2.
- [23] W. Viriyasitavat, F. Bai, and O. Tonguz, "Dynamics of network connectivity in urban vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 3, pp. 515–533, Mar. 2011.
- [24] N. Sofra, A. Gkelias, and K. Leung, "Route construction for long lifetime in VANETs," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3450–3461, Sep. 2011.
- [25] H. T. Cheng, H. Shan, and W. Zhuang, "Infotainment and road safety service support in vehicular networking: From a communication perspective," *Mech. Syst. Signal Process., Special Issue Integr. Veh. Dyn.*, vol. 25, no. 6, pp. 2020–2038, Aug. 2011.
- [26] J. Harri, F. Filali, and C. Bonnet, "Mobility models for vehicular ad hoc networks: a survey and taxonomy," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 4, pp. 19–41, Dec. 2009.
- [27] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO—Simulation of urban mobility," *Int. J. Adv. Syst. Meas.*, vol. 5, nos. 3/4, pp. 128–138, 2012.
- [28] C. Suthaputachakun and Z. Sun, "Routing protocol in intervehicle communication systems: A survey," *IEEE Commun. Mag.*, vol. 49, no. 12, pp. 150–156, Dec. 2011.
- [29] M. Boban, T. T. V. Vinhoza, M. Ferreira, J. Barros, and O. K. Tonguz, "Impact of vehicles as obstacles in vehicular ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 1, pp. 15–28, Jan. 2011.
- [30] F. K. Karnadi, Z. H. Mo, and K.-C. Lan, "Rapid generation of realistic mobility models for VANET," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2007, pp. 2506–2511.



**Nizar Alsharif** (M'09) received the B.Eng. degree in 2005 from the Department of Computer Engineering, Umm al-Qura University, Mecca, Saudi Arabia, and the M.A.Sc. degree in 2010 from the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, where he is currently working toward the Ph.D. degree.

He is a Lecturer with the Department of Computer Engineering, Al Baha University, Al Bahah, Saudi Arabia. His research interests include network's security and privacy, vehicular ad hoc networks, the Internet of things, and E-healthcare systems.



**Xuemin Shen** (M'97–SM'02–F'09) received the B.Sc. degree in 1982 from Dalian Maritime University, Dalian, China, and the M.Sc. and Ph.D. degrees in 1987 and 1990 from Rutgers University, New Brunswick, NJ, USA, all in electrical engineering.

He is a Professor and the University Research Chair with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. He is also the Associate Chair for graduate studies. His research interests include resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks.

Dr. Shen is an Elected Member of the IEEE ComSoc Board of Governor and the Chair of Distinguished Lecturers Selection Committee. He served as the Technical Program Committee Chair/Cochair for IEEE Globecom'16, Infocom'14, IEEE VTC'10 Fall, and Globecom'07; the Symposia Chair for IEEE ICC'10; the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC'08; the General Cochair for ACM Mobihoc'15, Chinacom'07, and QShine'06; the Chair for the IEEE Communications Society Technical Committee on Wireless Communications and Peer-to-Peer Communications and Networking. He also serves/served as the Editor-in-Chief for IEEE NETWORK, *Peer-to-Peer Networking and Application*, and *IET Communications*; a Founding Area Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS; an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Computer Networks*, *ACM/Wireless Networks*, etc.; and as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS MAGAZINE, *ACM Mobile Networks and Applications*, etc. He received the Excellent Graduate Supervision Award in 2006 and the Outstanding Performance Award in 2004, 2007, 2010, and 2014 from the University of Waterloo; the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada; and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. He is a registered Professional Engineer in the province of Ontario, Canada; an Engineering Institute of Canada Fellow; a Canadian Academy of Engineering Fellow; a Royal Society of Canada Fellow; and a Distinguished Lecturer of the IEEE Vehicular Technology and Communications Societies.