Real-Time Intersection-Based Segment Aware Routing Algorithm for Urban Vehicular Networks

Yusor Rafid Bahar Al-Mayouf, Nor Fadzilah Abdullah, Omar Adil Mahdi, Suleman Khan, *Student Member, IEEE*, Mahamod Ismail, Mohsen Guizani, *Fellow, IEEE*, and Syed Hassan Ahmed, *Senior Member, IEEE*

***Abstract*— High vehicular mobility causes frequent changes in the density of vehicles, discontinuity in inter-vehicle communication, and constraints for routing protocols in vehicular ad hoc networks (VANETs). The routing must avoid forwarding packets through segments with low network density and high scale of network disconnections that may result in packet loss, delays, and increased communication overhead in route recovery. Therefore, both traffic and segment status must be considered. This paper presents real-time intersection-based segment aware routing (RTISAR), an intersection-based segment aware algorithm for geographic routing in VANETs. This routing algorithm provides an optimal route for forwarding the data packets toward their destination by considering the traffic segment status when choosing the next intersection. RTISAR presents a new formula for assessing segment status based on connectivity, density, load segment, and cumulative distance toward the destination. A verity period mechanism is proposed to denote the projected period when a network failure is likely to occur in a particular segment. This mechanism can be calculated for each collector packet to minimize the frequency of RTISAR execution and to control the generation of collector packets. As a result, this mechanism minimizes the communication overhead generated during the segment status computation process. Simulations are performed to evaluate RTISAR, and the results are compared with those of intersection-based connectivity aware routing and traffic flow-oriented routing. The evaluation results provided evidence that RTISAR outperforms in terms of packet delivery ratio, packet delivery delay, and communication overhead.**

***Index Terms*— Communication overhead, VANETs, segment aware.**

I. Introduction

THE smart city concept has recently emerged as a popular topic in wireless network technologies research. By efficiently distributing their resources through wireless communication, smart cities become more competitive than other cities and provide new, attractive, and cost-effective services to their users. Smart cities are known for their smart mobility function, which is based on their intelligent transport system (ITS) [1], [2].

Up-to-date information regarding accidents, weather, traffic situation, and road obstacles can help drivers make excellent decisions, achieve a relaxing driving experience, and avoid road mishaps [3], [4]. Vehicular ad hoc networks (VANETs) consider the timely acquisition of accurate information as a key component of ITS [5], [6]. Wireless access in vehicular environments is a state-of-the-art technique used for ITS that allows vehicles to engage in wireless communication and establish VANETs [7].

VANETs comprise a group of vehicles or nodes that communicate with one another despite the lack of a fixed infrastructure support. Given the challenging propagation channels in high-mobility VANETs, the information in these networks must be transmitted to a specific destination through multi-hop wireless communication via intermediate vehicles. The presence of intersections, buildings, and vehicles may obstruct signal propagation, especially in urban environments, and prevent a direct single-hop communication between the sending and receiving vehicles [8]. Vehicles in VANETs act as routing nodes that share information on traffic mobility, vehicle direction, traffic density, vehicle speed, and weather conditions via vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communications [9]–[11]. In V2V communication, the information is transmitted and received wirelessly via on-board units (OBUs) that serve as network nodes that are mounted on vehicles. By contrast, in V2I communication, road side units (RSUs) are mounted on a fixed infrastructure to keep drivers informed about the traffic situation and to link the OBU vehicles with an external setup through various developing wireless technologies.

Based on safety applications, companies need to enhance road safety and reduce accidents [12], [13]. Apart from offering comfort or value-added services to drivers and passengers, these networks also offer non-safety applications, such as improving traffic efficiency [14]. Given their promising applications, high effectiveness, and low-cost construction, VANETs have received increasing attention from governments, car manufacturers, and academic institutions [15], [16].

However, the successful deployment of VANETs depends on several important components, one of which is the establishment of an optimal route from sources to destinations that requires the shortest communication time and least amount of

Manuscript received August 6, 2017; revised January 7, 2018; accepted March 16, 2018. This work was supported by Universiti Kebangsaan Malaysia under Grant GUP-2016-005. The Associate Editor for this paper was M. Alam. *(Corresponding authors: Yusor Rafid Bahar Al-Mayouf; Nor Fadzilah Abdullah.)*

Y. R. B. Al-Mayouf and O. A. Mahdi are with the Department of Computer Science, College of Education for Pure Science (Ibn Al-Haitham), University of Baghdad, Baghdad 10071, Iraq (e-mail: engineer_mu2007@yahoo.com).

N. F. Abdullah and M. Ismail are with the Faculty of Engineering and Built Environment, Universiti Kebangssan Malaysia, Selangor 43600, Malaysia (e-mail: fadzilah.abdullah@ukm.edu.my).

S. Khan is with the School of Information Technology, Monash University, Subang Jaya 47500, Malaysia.

M. Guizani is with the Department of Electrical and Computer Engineering, University of Idaho, Moscow, ID 83844 USA.

S. H. Ahmed is with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816 USA.

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/TITS.2018.2823312

network resources. The inherent characteristics of VANETs, such as the frequent changes in their topology and their highly mobile vehicles, establish an unstable communication route among vehicles [17], [18], thereby presenting a key challenge in information routing and limiting the benefits of these networks [19], [20].

Although RSUs can be used to solve such problems and facilitate communication among vehicles, the deployment of these units entails a high cost. Therefore, V2V communications have a more significant role in improving network throughput than V2I communication [21]. In this case, the reliance on V2V communication presents the most critical obstacle in the routing of VANETs when the deployment of RSUs is impractical.

In VANETs, the two main types of environments are highways and urban. This study only focuses on urban environment that comprises intersections and segments between intersections. Each segment has two lanes where the traffic moves bi-directionally. Urban environment has been found to possess a number of characteristics that result in more critical communication in VANETs as follows. Signal propagation is affected by buildings located at corners and intersections. Too many obstacles cause signal interference and result in major packet loss. Many streets, paths, and squares are located near each other, providing the driver many options. Frequent alteration in vehicle speed occurs due to speed bumps, traffic lights, and crosswalks, which causes high speed variance. Thus, the connectivity time may be reduced, even among vehicles travelling in a similar direction [22].

Although there are a variety of V2V routing protocols designed purposely for VANETs [19], [23], outstanding performance results have been exhibited by geographic-based routing as the routes between source and destination do not need to be established and saved, which satisfies the condition of dynamic changes in VANETs [24], [25]. Furthermore, in comparison to other types of routing protocols, such protocols support scalable networks with reduced communication overhead [26], [27]. Finally, they are simply because they apply global positioning system (GPS) technology to determine the exact positions of a vehicle with respect to its longitude and latitude [28], [29]. Therefore, this study only focuses on geographic-based routing, which is considered the most promising, suitable, and scalable strategy for VANETs considering the dynamic nature of these networks.

For decades, several geographic routing protocols have been proposed with mainly focused on the optimal route with minimum hop count for selection best next candidate node without considering segment and traffic status. Several limitations must be addressed in these protocols. For instance, the rapid movement of highly mobile nodes in VANETs may result in route failure between consecutive relay nodes. The recurrent route breakages require a recovery process, which in turn increases packet loss, network delay, and communication overhead.

In order to eliminate conventional geographic routing protocols limitations, segment and traffic status metrics have been introduced in recent routing decisions results in segment aware routing protocols [30]–[32]. Adopt a dynamic routing decision and uses routing metrics to select the next intersection for each packet. The best segment for packet forwarding is selected by evaluating the adjacent segments at each intersection. Basically to get information about segment and traffic status, there exist many processes. However, real-time segment evaluation processes are preferred, as they adapt more to the dynamic environment of VANETs. The real-time evaluation process at each intersection is usually based on the transmission of information collector packets (CPs). This will be used to collect the network and traffic information of each segment and to forward from the current to the next intersection and vice versa by collector packets reply (CPRs). Although these protocols can be easily applied in diverse VANET conditions because of the sequential decision-making process at each intersection, they often encounter network delays and overhead during the evaluation procedure. The uninterrupted availability of routing metrics for making routing decisions presents another challenge that requires the frequent updating of routing results. However, initiating many measurement procedures will produce high communication overhead, thereby affecting on packet delivery ratio and delay.

Accordingly, the main problems focused in this study are summarized as follows:

- The selection of metrics in existing segment aware-based geographic routing protocols insufficient to execute end-to-end route selection for best utilization of the network resources in a VANET environment.
- The conventional segment aware-based geographic routing protocols still suffer to maintain a stable network without congestion.

To address these problems, a novel intersection-based routing is proposed in this paper known as real-time intersection-based segment aware routing (RTISAR) algorithm towards finding an optimal route to the destination. The major contributions of this paper are:

- RTISAR, a new optimal segment aware-based geographic routing algorithm in urban VANET that considers traffic segment status when selecting the next intersection. Apart from combining the advantages of the segment result metric, this is based on their connectivity, density, and load of segments. RTISAR also considers the cumulative distance to a specific destination. This algorithm can avoid selecting intersections with low connectivity, sparse density segment, high load segment, and low cumulative distance to the destination.
- A verity period mechanism (V_period), which considers the lifetime duration, density, and flow rate metrics to denote the projected period when a network failure is likely to occur in a particular segment. V_period can be calculated for each CP to minimize the frequency of RTISAR execution and to control the generation of CPs. A new CP cannot be generated by the vehicle until the V_period of the received CP expires. As a result, V_period mechanism minimizes the communication overhead generated during the segment status computation process.

This reminder of this paper is structured as follows: Section II discusses several intersection-based segment aware protocols proposed in the context of VANETs. Section III

TABLE I
SUMMARY OF INTERSECTION-BASED ROUTING PROTOCOLS

	Considered intersection routing metrics					**Evaluated segment status mechanism**		**Considered segment routing metrics**					
Routing protocols	**Distance**	**Density**	**Communication load**	**Communication delay**	**Connectivity**	**CPs**	**CPRs**	**Distance**	**Speed**	**Direction**	**Received signal strength indication**	**Lifetime**	**Transmission error rate**
GyTAR (2009)	✓	✓	×	×	×	✓	×	✓	✓	✓	×	×	×
EGyTAR (2011)	✓	✓	×	×	×	✓	×	✓	✓	✓	×	×	×
VDLA (2012)	✓	✓	✓	×	×	✓	✓	✓	×	×	×	×	×
iCAR (2013)	✓	✓	×	✓	×	✓	✓	✓	×	×	✓	×	×
EVDLA (2014)	✓	✓	✓	×	×	✓	✓	✓	×	×	×	✓	×
LSHR (2014)	✓	×	×	✓	×	✓	×	✓	×	×	×	×	×
TFOR (2014)	✓	✓	×	×	×	✓	×	✓	×	×	×	×	×
iCARII (2014)	✓	✓	×	✓	×	✓	✓	✓	×	×	✓	×	×
proposed (RTISAR)	✓	✓	✓	×	✓	✓	✓	✓	×	×	×	×	✓

introduces RTISAR functionality, design, and components. Section IV presents and discusses the performance evaluation of RTISAR. Finally, Section V concludes the paper.

II. RELATED WORKS

The most recent intersection-based routing protocols are discussed in the following and TABLE I summarizes the comparison between them.

A geographical greedy traffic-aware routing (GyTAR) protocol proposed that dynamically chooses intermediate intersections through which packets can travel to reach their destination [33]. The number of vehicles and the curve metric distance toward the destination are considered during the selection process. The segment with the largest number of vehicles and shortest route to the destination is given the highest score. Fixed location cells are formed within each segment, and each cell shares the same radius as the transmission range. The vehicle located nearest to the center of the cell is nominated as the group leader that measures the number of vehicles within the cell. The measurement results are then transmitted from the group leader with the help of CP. The group leader generates a CP upon leaving the segment, and then the generated CP is backward transmitted to the first cell leader in the targeted segment. The corresponding motion information, including position, direction, and speed, is stored in a neighbor table and can be used to predict the position of each neighbor according to the forwarding vehicle. The next hop is then selected by nominating the neighbor with the closest predicted position. Before reaching the end of the segment, the CP traverses through the cell leaders. The density of the corresponding cell is added to the CP upon reaching a group leader, which in turn forwards the density information to the next cell. The CP is analyzed upon reaching the opposite intersection to calculate the average vehicle density of each cell. The segment under evaluation is given a score depending on its density and distance from the destination.

An enhanced GyTAR (EGyTAR) proposed that uses the direction of the vehicle to calculate the vehicle density of a segment [34]. This protocol calculates the density of vehicles that move in the direction toward the destination to estimate accurately the quality of packet transmission across the segments. Consequently, those segments with a high vehicle density and are located opposite to the direction of destination are not considered. Undoubtedly, the large number of vehicles that move toward the destination can generate a high packet delivery ratio, and a low density of vehicles that move in the opposite direction is generally preferred.

A traffic flow-oriented routing (TFOR) protocol proposed that improves the safety and organization of traffic by facilitating driving via a smart transportation system [35].

TFOR dynamically chooses the intersections by considering the directional and non-directional traffic density flow. The sending vehicular node in an intersection uses a digital map to identify the location of the neighboring intersection. The position of the destination is located using a location service. Afterward, the sending vehicle determines the curve metric distance from each neighbor intersection to the destination vehicle. Each neighbor intersection is given a score, and the intersection with the highest score is selected as the successive destination intersection. The chosen intersection must be positioned closest to the destination vehicle and must have a high traffic density flow. TFOR uses CPs to update traffic information. All data packets are marked with the position of the next intersection. The position, speed, and direction of each vehicle are recorded in the neighbor table, which entries updated periodically. The neighbor tables are consulted for their predicted positions when the source or intermediate vehicles are forwarding data packets. This strategy improves the routing position by reducing the end-to-end delay and number of hops. The forwarding node searches for the neighbor table of the two-hop neighbor, and then the packet is dispatched via greedy forwarding strategy by the neighbor that is positioned nearest the destination.

However, there are some limitations in GyTAR, EGyTAR, and TFOR protocols. Although the CP generation process is suppressed, generating a new CP is not guaranteed to change the segment status. Moreover, these protocols do not generate CPRs, thereby indicating that the evaluation results for a one-way segment are unavailable at the first intersection. Given that the best segment is selected based on the number of vehicles and distance, a high communication load and signal interference may be experienced in cases with high network density. Therefore, an efficient routing performance cannot be guaranteed. Moreover, those segments with high intermittent communications may be selected for data forwarding because the stability of the communication link is not considered.

A vehicle density and network load aware (VDLA) routing protocol proposed that uses a real-time density and traffic load collection mechanism [36]. In this protocol, if the current and next segments are placed on the similar direction, then the packets are forwarded in the greedy mode. However, if the directions are dissimilar, then the packet is forwarded toward the intersection node, thereby allowing transmissions toward different directions. Intermediate intersections are selected dynamically based on three routing parameters, namely, distance to destination, real-time density, and traffic load measurements. An intersection node assigns weight scores to all adjacent segments upon their evaluation. Subsequently, Hello packets are broadcasted by the intersection node that includes the weights of the adjacent segments. After a vehicle reaches an intersection, the measurement process is initiated by transmitting a CP to each adjacent segment. VDLA considers the number of packets in the buffer queue of a node to evaluate the load of a node and to measure the traffic load. The neighbors of the initiator vehicle and buffer queue fields are then added to the CP. When the forwarded CP is received by a node, the neighbors on its right side are added to the neighbor list of the CP. The buffer queue length is extended in a similar manner. The neighbor count is then compared with the minimum neighbor number (MNN) field, which evaluates the distribution of vehicles along the segment. In case MNN outnumbers the neighbor count, the number of neighbors of the current forwarder is added to this field and the updated packet is forwarded to the destination. After the CP packet is processed at the next intersection, another CP is backward transmitted to the first intersection using the aforementioned strategy. The number of vehicles and the measured traffic load for all adjacent vehicles are received at the initiator vehicle. The calculated data are used to evaluate the segment, and then the adjacent segments are given scores. However, if the vehicle receives the CP before reaching the intersection, then a timer will start for one second and new CP will be generated. Therefore, the communication overhead related to CP is greatly lowered under dense conditions.

An enhanced VDLA (EVDLA) protocol proposed that improves the accuracy of neighbor tables to enhance the performance of the geographic greedy forwarding strategy [37]. This protocol uses a conventional Hello packet with a static broadcast interval to describe two scenarios. First, when the neighbors exit the transmission range, their entries are not removed. Second, even after becoming neighbors, some vehicles are not listed in the neighbor table because their next Hello packet has not yet been sent. Given that the farthest node is selected as the next hop by the classical greedy forwarding strategy, any inaccuracy in the data of these nodes may prevent a successful packet transmission. Therefore, the lifetime of each neighbor is calculated and stored in the neighbor table by VDLA. Moreover, the intermediate vehicle calculates the communication time of the two vehicles that are directed toward the transmission range of each other. This information is then sent to another packet, and waiting for the arrival of the Hello packet is no longer required. However, given the influence of the variability in the speed and behavior of the driver, the lifetime prediction has a questionable dependency. EVLDA also increases routing overhead because the speed of advancements increases along with that of vehicles.

However, there are some limitations in VDLA and EVDLA protocols. Given the unavailability of a recovery strategy and the failure to consider the stability of the communication link, the data may be forwarded via segments with a high level of intermittent communication.

A link state aware hierarchical road (LSHR) routing protocol proposed that uses 3D coordinates in positioning vehicles to improve the packet delivery rate and to reduce the number of hops and transmission delay [38]. The routing metrics are measured to weigh the adjacent segments of the intersection. The segment with the lowest weight is selected for forwarding the packets. In LSHR, a sorting table is generated at an intersection node upon receiving a data packet. Communication delays and progressive distance toward the destination are included in the sorting table. The table is generated by storing the sending time in the CP during its transmission toward adjacent intersections. The communication delay of the segment is calculated by the last node at which the CP arrives. The delay and the corresponding segment ID are broadcasted

to the neighbors. The communication delay of the adjacent segments is stored in the sort table and broadcasted to the nodes within the intersection region. Each segment is given a weight score based on distance and communication delay. LSHR also calculates the virtual distance of the nodes that are positioned nearer the destination than the current node. Virtual distance refers to the longest two-hop distance. The neighbor with the longest virtual distance is selected as the next hop node. The selection of the best segment must not only depend on the forwarding delay of the segment because traffic density and communication stability are not indicated by forwarding delay, which only represents the status of the network load of the segment that is under evaluation.

An intersection-based connectivity aware routing (iCAR) protocol proposed that considers the number of vehicles in real time and the packet delivery delays for each segment [39]. Upon reaching an intersection, iCAR evaluates the adjacent segments based on their routing parameters and movement toward the destination. Afterward, the segment with the highest density, minimum delay, and closest distance to the destination is selected. iCAR forwards a CP along the segment and the promptness of segment connectivity is calculated when the CP is delivered at the next intersection. The network density is calculated between the current and next forwarder vehicles while forwarding the CP. Along with traversed node count and CP generation time, the calculation results are added to the vehicle count that is currently stored in the CP. Before sending the packet to the next hop, the forwarder computes for the validity period and updates the lifetime field in case a short validity period is obtained. The validity period is used to validate the data in a certain segment and to predict the disconnection time. The updated score is generated by the vehicle that is nearest to the segment intersection center when the CP is received. Beacon packets are used to announce the updated score along with its validity time, and the score is then transmitted back to the location where the CP packet is created. The next hop is selected not only on the basis of its closeness to the destination but also if this hop has the highest received signal strength indication.

An intersection-based connectivity aware routing (iCARII) protocol proposed that improves routing performance by increasing packet delivery ratio [40]. iCARII uses the infrastructure facilities and broadcasts one-hop beacon packets to update the driving conditions periodically, and the vehicles are equipped with GPS-based on-board and RSUs for Internet access. iCARII has four main components, namely, segment evaluation, verity period calculation, next intersection selection, and next-hop selection. In segment evaluation, the protocol dynamically senses different parts of the network to obtain real-time information about the network. In this phase, the protocol broadcasts lightweight control packets to check the traversing segments and intersections. The protocol calculates the verity period to determine the path lifetime, and then takes the decision made by the vehicles of one-hop neighbors. After calculating the verity period, iCARII selects the next intersection and determines the existence of a path toward the destination. If the packet reaches a disconnected segment because of an unexpected delay, then the current candidate node sends a new path request to the location centers for recovery.

However, the limitations of iCAR and iCARII is to use verity period as a routing metric not to evaluate the stability of a communication link but to ensure that the expired segment evaluation results are not considered in the routing process. Moreover, when computing the verity period, iCAR and iCARII do not consider the accessibility of alternate links that result from the movement of vehicles and can be used to replace failed links. In this case, iCAR and iCARII only compute a short verity period that cannot accurately adjust to changes in segment status, thereby increasing communication overhead and redundantly generating CPs.

This paper proposes the RTITAR algorithm to address the limitation of previously mentioned researchers as discussed earlier. The novelty of this work lies in its unique design based on suppressing the evaluation procedure of the segment and forwarding the result from the current to the next intersection and vice versa when selecting an optimal intersection toward the destination with low communication overhead and intermittent communications. In segment routing, the metrics related to local distance and transmission error rate are used to select the next node for forwarding the data packet to the destination (end segment). In intersection routing, the stability of the communication link is assessed based on lifetime duration, local density, and flow rate metrics, while the connectivity of the segment is evaluated based on density, load segment, and cumulative distance when choosing the next intersection toward the destination.

III. Assumption and Description of RTISAR

For the system model, RTISAR considers a vehicular network in an urban environment that comprises intersections and segments between intersections. Each segment has two lanes where the traffic moves bi-directionally. In RTISAR, each vehicle maintains a neighbor table to record the position, speed, and direction of the neighboring vehicles using GPS. The information about the neighboring vehicles is obtained by exchanging beacon packets [41]. Location services, such as hierarchical location service or grid location service, can be used to record the position of the destination vehicle [42], and an on-board navigation system installed in each vehicle can record the location of the neighboring intersections. Each vehicle must maintain a segment table to record the received RTISAR results and to make routing decisions at each intersection. A digital map can be used to obtain the segment identifier depending on the position of the vehicle as stated in [43]. RTISAR employs the carry-and-forward strategy to recover from the local optimum situation to recover from the local optimum situation [35]. This protocol involves three phases, namely, i) generation, forward, and declared CP, ii) bounded generation of CPR, iii) decision data packet routing, which all work sequentially to deliver packets from the source to the destination as shown in Figure 1.

A. Generate, Forward and Declared Collector Packet Phase

The GFD-CP phase is initialized by the vehicle that moves on a segment by generating a CP upon reaching the

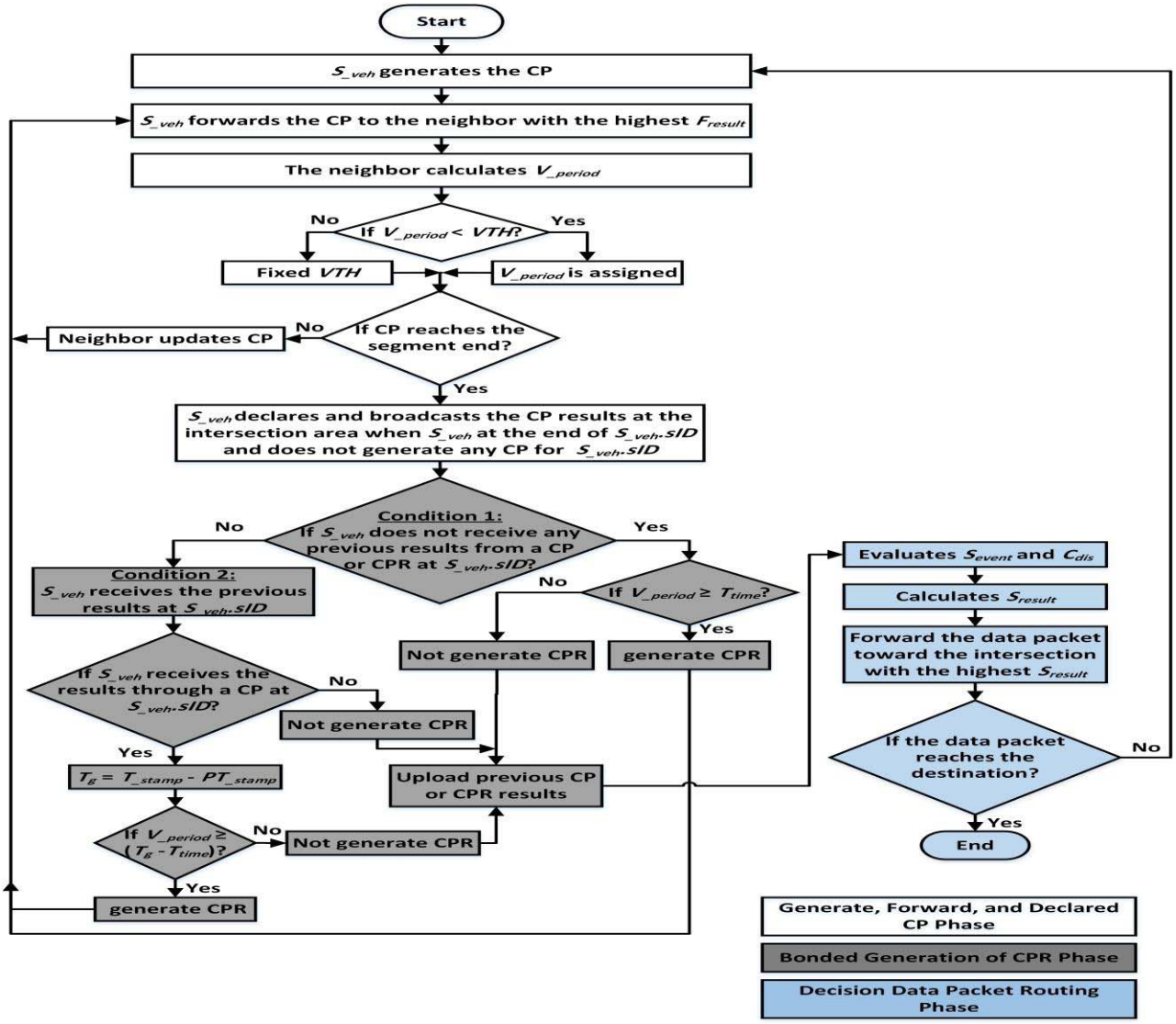


Fig. 1. Flowchart of RTISAR.

$S_{\_veh}$	$N_{\_veh}$	$S_{\_ID}$	$S_{\_dir}$	$O_{\_dir}$
$V_{\_period}$	AL_{S_dir}	AL_{O_dir}	$N_{\_hop}$	$CP_{\_type}$
$T_{\_stamp}$				

Fig. 2. Collector packet information.

intersection area. The CP collects information about the traffic segment status, which is then forwarded from the current to the next intersection. The CP contains the following information fields as shown in Figure 2.

- $S_{\_veh}$ denotes the address of the vehicle that presently has the CP;
- $N_{\_veh}$ denotes the address of the neighbor forwarder. The current position of each registered neighbor is predicted by considering mobility metrics, including speed, direction, and updated position. $S_{\_veh}$ assigns the $N_{\_veh}$ based on the F_{result} metric (explained later in details);
- $S_{\_ID}$ is the identifier of the segment, and $S_{\_veh}$ attaches its current identifier segment;
- $S_{\_dir}$ denotes the total number of vehicles within the vicinity of $S_{\_veh}$ that follow the forwarding direction of CP. The area between $S_{\_veh}$ and$N_{_veh}$ is only considered for counting neighbors to avoid errors resulting from the inaccurate count of neighbors. $O_{\_dir}$ is measured in the same way as $S_{\_dir}$, but this parameter monitors the number of vehicles that move in the opposite direction;
- $V_{\_period}$ is the verity period, which denotes the projected period when a network failure is likely to occur in a particular segment. When the CP is generated, $S_{\_veh}$ initializes the first value of $V_{\_period}$;
- AL_{S_dir} and AL_{O_dir} are the average loads for $S_{\_dir}$ and $O_{\_dir}$, respectively, which are incremented and updated at each subsequent forwarder. To understand the concept of segment load, we must know how the load for each vehicle is determined. The total number of packets in the buffer queue of a vehicle determines the load of that vehicle. Therefore, the length of the buffer queue of each vehicle is added to the total number of periodically

TABLE II
SUMMARY OF THE IMPORTANT MATHEMATICAL NOTATIONS

Symbol	Description
V_period	Verity period
LTD	Lifetime duration
$\overline{LTD}$	Average lifetime duration
LD	Local density
S_dir	Total number of vehicles within the vicinity of S_veh
O_dir	Total number of vehicles that move in the opposite direction of S_veh
$LFRC$	Local flow rate changeable
FR	Flow rate
LD	Local density
V	Average speed
R_n	Distance
D_{error}	Transmission error rate
D_{for}	Forward delivery ratio of the link
D_{rev}	Reverse delivery ratio of the link
F_{result}	Forward result
S_{event}	Segment event
S_{den}	Segment density
S_{con}	Segment connectivity
S_{load}	Segment load
N_hop	Number of hops
N_{con}	Ideal connectivity degree
D_{hop}	Delay hop
T_{time}	Travel time for the CP
AL_{S_dir}	Average loads for S_dir
AL_{O_dir}	Average loads for O_dir
N_{load}	Maximum number of packets in the buffer queue
S_{result}	Segment result
$D_{cur\text{-}dest}$	Distance between the current intersection and the destination
$D_{can\text{-}dest}$	Distance between the candidate intersection and the destination

broadcasted beaconing packets, and the result is recorded on the neighbor table. When S_veh enters a segment, this parameter calculates AL_{S_dir} and AL_{O_dir}. Afterward; the updated CP is unicasted to the next forwarder based on the F_{result} metric (explained later in details). This process continues until the CP reaches the vehicle at the other end of the segment;

- N_hop indicates the number of vehicles that have participated in the forwarding of the CP. At each succeeding forwarder, N_hop increases by 1;
- CP_type differentiates CP from CPR. The values of 0 and 1 indicate CP and CPR, respectively; and
- T_stamp indicates the time when the CP or CPR is generated.

To facilitate the understanding of technical aspects, a table of notations would be useful as illustrated in TABLE II.

The segment evaluation procedure generates high communication overhead in intersection-based segment aware routing protocols because the CPs are forwarded between two subsequent intersections to compute the traffic and segment status along the segment. Therefore, GFD-CP applies a V_period mechanism that can be calculated for each CP to reduce the frequency of RTISAR execution and to control the generation of CPs. The declared vehicle makes the decision regarding the generation of CP. A new CP cannot be generated by the vehicle until the V_period of the received CP expires. The stability of traffic and the segment status of a segment depend on the high values of V_period. GFD-CP considers some metrics when calculating V_period, such as lifetime duration, density, and flow rate.

First, the lifetime duration (LTD) defines the duration for each neighbor that remains within the radio range and belongs to the same segment within S_veh. Based on the information in the neighbor table, LTD can be computed as follows [44]:

$$LTD = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \quad (1)$$

where $a = v_s \cos\theta_s - v_n \cos\theta_n$, $b = x_s - x_n$, $c = v_s \sin\theta_s - v_n \sin\theta_n$, $d = y_s - y_n$. r denotes the radio range. (x_s, y_s), (x_n, y_n), v_s, v_n, θ_s, θ_n refer to the x coordinate, v speed, and θ movement direction angle for vehicles s and n, respectively. s represents S_veh, while n represents the neighbor.

A shorter V_period is generated if this parameter is computed based on the LTD of the link between S_veh and N_veh that is likely to fail soon. The actual connectivity state is not determined because the other links between S_veh and the alternative N_veh are ignored. Therefore, the average LTD ($L\bar{T}D$) between S_veh and all the possible N_veh is calculated using GFD-CP.

Second, the stability of the IVC links in VANETs greatly depends on density [45]. For instance, the S_veh with a high local density can have many substitute communication links to maintain network connectivity and successfully forward packets within the vicinity of S_veh. Therefore, GFD-CP considers the local density (LD) metric when calculating V_period as follows:

$$LD = S\_dir + O\_dir \quad (2)$$

However, LD is highly changeable because of the high speed and uncertain mobility of the vehicles along the network, which damage the stability of communication links. Therefore, GFD-CP considers the local flow rate changeable (LFRC) metric through the calculated V_period. $LFRC$ is computed as difference in the flow rate (FR) in a specific interval (i.e., $LFRC = FR_t - FR_{t-1}$). All vehicles use the same fixed time interval to calculate $LFRC$. FR can be computed as follows:

$$FR = LD * V \quad (3)$$

where LD and V are the local density and average speed in a specific interval, respectively.

The aforementioned metrics clearly show that the stability of a communication link is affected by $L\bar{T}D$, LD, and LFRC. The stability of the IVC link is inversely related to LFRC and is directly related to both $L\bar{T}D$ and LD. Such relationship can be formulated as follows to compute V_period:

$$V\_period = \frac{\overline{LTD} \times LD}{LFRC} \quad (4)$$

Consequently, the V_period is calculated by every forwarder. The calculated and received values are compared, and the V_period field assigns the calculated value if such value is less than the received V_period. The updated CP is then unicasted to the next forwarder.

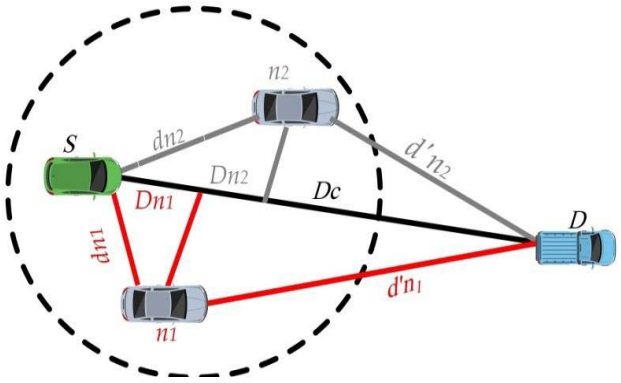


Fig. 3. Impact of distance factor in finding next-hop forwarding vehicle.

As mentioned above, the next forwarder is selected based on the forward result (F_{result}) metric, an improved greedy-based forwarding strategy that includes two parameters, namely, the distance (R_n) toward the next intersection and the transmission error rate (D_{error}).

R_n tends to select neighbors that demonstrate the highest progress toward the destination. As shown in Figure 3, the line segment *SD* that joins the source with the destination is drawn to project vehicles n_1 and n_2. Dc denotes the shortest distance between the source and destination vehicles, while d and d' denote the distances from the intermediate vehicles (n_1 and n_2) to the source and destination, respectively. Dn_1 and Dn_2 are the distances that measure the progress of vehicles n_1 and n_2 from the source vehicle toward the destination vehicle, and the distance (R_n) can be calculated as follows [24]:

$$R_n = \frac{R^2(S,D) + R^2(S,n) - R^2(n,D)}{2R^2(S,D)} \tag{5}$$

where S, D, and n denote the $S_{\_veh}$, next intersection, and neighbor, respectively,

$$R(S,D) = \sqrt{(x_{Dx} - x_{Sx})^2 + (y_{Dy} - y_{Sy})^2},$$
$$R(S,n) = \sqrt{(x_{nx} - x_{Sx})^2 + (y_{ny} - y_{Sy})^2},$$
$$R(n,D) = \sqrt{(x_{Dx} - x_{nx})^2 + (y_{Dy} - y_{ny})^2}$$

The vehicle with the maximum R_n toward the next intersection is selected as the next hop. The delivery delay decreases along with the number of hops. Figure 3 shows that according to the distance parameter selection, vehicle n_2 is preferred over vehicle n_1.

However, the classic greedy-based forwarding strategy suffers from a high packet loss ratio. RTISAR prevents the use of low-quality communication links and reduces packet losses by employing the improved greedy forwarding strategy, R_n, and D_{error}.

D_{error} greatly influences link communication quality, which is used to calculate the number of packets that are successfully delivered and dropped between $S_{\_veh}$ and each neighbor. D_{error} can be computed as follows [46], [47]:

$$D_{error} = 1 - D_{ratio} \tag{6}$$

where $D_{ratio} = D_{for} * D_{rev}$. D_{for} and D_{rev} denote the forward and reverse delivery ratios of the link, respectively.

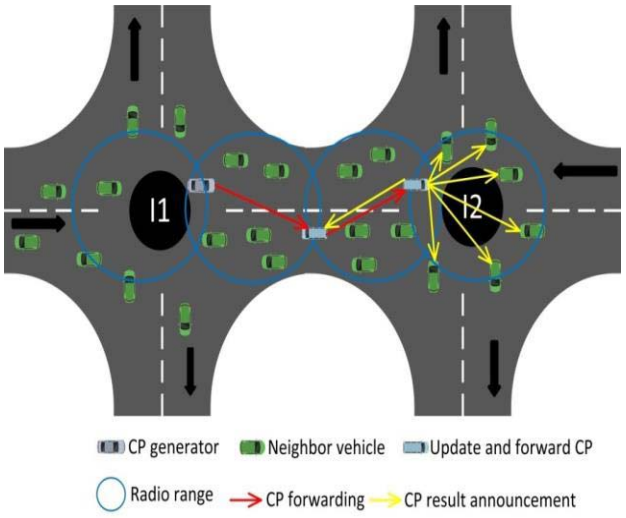


Fig. 4. GFD-CP phase.

Greedy forwarding has an inverse relationship with D_{error} and a direct relationship with R_n. These relationships can be expressed as follows to compute F_{result}:

$$F_{result} = \frac{R_n}{D_{error}} \tag{7}$$

$S_{\_veh}$ forwards the CP to the neighbor with the highest F_{result}. In this case, the forwarded CP delivers the lowest $V_{\_period}$ to the end of the segment. Upon reaching the end of the segment, the total number of neighbors for all visited vehicles is assigned to the density fields of CP. This value signifies the density of vehicles that are associated with the segment under consideration. The entire neighbor load of all visited vehicles is given by the average load fields of the CP, which signifies the load that is associated with the segment under consideration. The repeated mechanism of GFD-CP is employed until the next intersection receives the CP that processes and declares the carried RTISAR results as shown in Figure 4. The processed results are broadcasted at the intersection region by attaching them to the succeeding beacon packet. Algorithm 1 illustrates the entire GFD-CP phase.

B. Bounded Generation of Collector Packet Reply Phase

At the end of GFD-CP, RTISAR generates a CPR that is directed to the starting intersection as shown in Figure 5 in order for the computed results of RTISAR to become readily available at the intersections and to suppress the redundant generation of CPs.

When the vehicles moves in a single direction, the CPR must deliver the RTISAR results based on the first intersection (e.g., I1) because the results are carried and declared by the CPs only at the second intersection (e.g., I2). However, if CPRs are generated to lower rates, the bounded communication overhead (CO) of RTISAR is observed by employing two conditions. First, for the present segment, the CP must not be generated by the declared vehicle that is positioned at the end of the segment. Second, the decision regarding the generation of CPR is made by the declared vehicle based on the following cases. Algorithm 2 illustrates the BG-CPR phase:

Algorithm 1 GFD-CP Phase

Input: *The vehicle's position of generated CP and the information of neighbors*
Output: *Forward and declare CP in the end of segment*
Notation:
V_{TH}: *Initialization received* V_period
1. ***Begin Algorithm 1***
2. S_veh *generating a CP, when it starts entering segment;*
3. S_veh *forward CP to neighbor, which has higher Fresult;*
4. *Neighbor calculates* V_period*;*
5. ***If*** $V\_period < V_{TH}$ ***Then***
6. V_period *is assigned to the* V_period *field;*
7. ***Else***
8. *Fixed* V_{TH}*;*
9. ***If*** *CP reaches to segment end* ***Then***
10. S_veh *declares and broadcasts CP results at intersection area;*
11. ***Else***
12. *Neighbor updates CP;*
13. *Go to 3;*
14. ***End If***
15. ***End Algorithm 1***

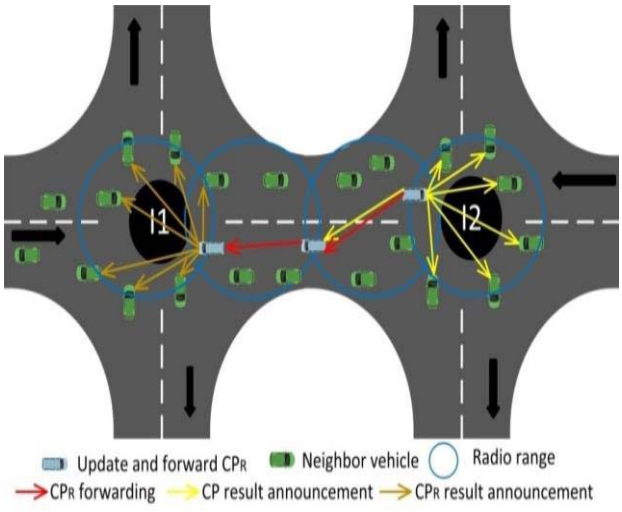


Fig. 5. BG-CPR phase.

Condition 1: For the present segment, the CP must not be generated by the declared vehicle that is positioned at the end of the segment. In addition, the declarer vehicle does not receive any previous RTISAR result through the generation of a CP or CPR at intersection I1. This situation implies that before the current receipt of CP, the vehicle does not generate CP or CPR while driving from I1 to I2. In this case, the RTISAR results can reach I1 as long as the V_period remains valid. Therefore, the declarer vehicle driving toward I1 generates and forwards the CPR.

Condition 2: For the present segment, the CP must not be generated by the declared vehicle that is positioned at the end of the segment. In addition, the declarer vehicle receives the previous RTISAR results through the generation of a CP or CPR at intersection I1. In case for the CP that is generated at I1, the CP carries valid RTISAR results by the declarer vehicle. This condition implies that two CPs have been generated successively at I1. Therefore, the declarer vehicle sends a CPR if the reply packet suppresses the newly generated CP at I1. Before generating a new CP, the CPR must arrive at I1 to prevent the generation of a new CP. This situation is only possible if the V_period of CPR arriving at the intersection is valid. Otherwise, no CPR is generated. In case for the CPR that is generated at I1, the CPR carries valid RTISAR results by the declarer vehicle. This condition implies that the RTISAR results have been declared at I1 and the CPR is sent to I2. Therefore, CPR does not need to be sent back for the provision of RTISAR results at I1.

Algorithm 2 BG-CPR Phase

Input: S_veh*'s segments table and received CP*
Output: *Bounded generation of CPR*
Notation:
$S\_veh.sID$: S_veh *identifier segment*
T_{time}: *Travel time of the current CP*
T_g: *Time between the two subsequent generations of CP at* $S\_veh.sID$
T_{stamp}: *Generation time of the current CP*
PT_{stamp}: *Generation time of the previous CP*
1. ***Begin Algorithm 2***
2. S_veh *declares the CP when* S_veh *at the end of* $S\_veh.sID$ *and does not generate any CP for* $S\_veh.sID$*;*
3. ***Condition 1****:* ***If*** S_veh *does not receive any previous result from a CP or CPR at* $S\_veh.sID$

Then
4. ***If*** $V\_period \geq T_{time}$ ***Then***
5. *Generate and forward CPR;*
6. ***Else***
7. *Not generate CPR;*
8. ***Else***
9. ***Condition 2****:* S_veh *receives the previous results at* $S\_veh.sID$*;*
10. ***If*** S_veh *receives the results through a CP at* $S\_veh.sID$ ***Then***
11. $T_g = T_{stamp} - PT_{stamp}$*;*
12. ***If*** $V\_period \geq (T_g - T_{time})$ ***Then***
13. *Generate and forward CPR;*
14. ***Else***
15. *Not generate CPR;*
16. ***Else***
17. *Not generate CPR;*
18. ***End If***
19. ***End Algorithm 2***

C. Decision Data Packet Routing Phase

As mentioned in phases GFD-CP and BG-CPR, when the vehicle receives a CP or CPR at an intersection, the carried RTISAR results are processed to evaluate the segment event (S_{event}) metric. S_{event} is based on three parameters, namely,

segment density (S_{den}), segment connectivity (S_{con}), and segment load (S_{load}).

S_{den} denotes the density of vehicles that are moving in the segment under consideration. The density field values for CP and CPR (S_dir and O_dir) and the number of hops are used to compute S_{den} as shown in Eq. 8:

$$S_{den} = \frac{(\alpha \times S\_dir) + (\beta \times O\_dir)}{N\_hop \times N_{con}} \tag{8}$$

High packet delivery ratio and low packet delivery delay are obtained if the packets are being forwarded through those segments where the vehicles moving in the same direction outnumber those that are moving in the direction in which the packets are being forwarded (i.e., S_dir) [34]. Therefore, in Eq. 8, a high weighing factor α equal to 0.75 is given to S_dir, while a weighting factor β equal to 0.25 is given to O_dir. N_{con} is a constant (set to 12) that represents the ideal connectivity degree to guarantee an excellent end-to-end connectivity. S_{den} is mapped to the range (0, 1). If the value of S_{den} is greater than 1, then this parameter is mapped to 1. Otherwise, S_{den} is in the range (0, 1).

S_{con} monitors the communication status of vehicles in the segment under consideration. RTISAR evaluates S_{con} by computing the number of CP packets that can be processed and transmitted before the status of traffic within its vicinity changes. Given that V_period gives the probable time for a network status change to occur on the segment, this parameter is used to compute S_{con} as follows:

$$S_{Con} = \frac{V\_period}{D_{hop}} \tag{9}$$

where, $D_{hop} = T_{time}/N\_hop$, D_{hop} is the delay hop that represents the delay encountered at each hop during the processing and transmission processes, and T_{time} denotes the travel time for the CP. The number of packets is mapped to the range (0, 1). If the value of S_{con} is closer to 1, then the connectivity in the segment under consideration is high. By contrast, if the value of S_{con} approaches 0, then the connectivity in the evaluated segment is poor.

S_{load} measures the status of the segment load in the evaluated segment. The received values of average load fields for CP and CPR (AL_{S_dir} and AL_{O_dir}) and the number of hops are used to compute S_{load} as follows:

$$S_{load} = \frac{(\alpha_1 \times AL_{S\_dir}) + (\beta_1 \times AL_{O\_dir})}{N\_hop \times N_{load}} \tag{10}$$

N_{load} represents the maximum number of packets that can be accommodated in the buffer queue. In the MAC layer 802.11, this value is set to 50. If the packets in the buffer queue reach their maximum number, any new packet will be discarded. Forwarding the packets through those segments where a higher load is being moved in the direction in which the packets are being forwarded (i.e. S_dir) will generate a high overhead, reduce the packet delivery ratio, and extend the delivery delay. Therefore, a high weighting factor α_1 equal to 0.75 is given to S_dir, while a weighting factor β_1 equal to 0.25 is given to O_dir. S_{load} is mapped in the range (0, 1). If the value of S_{load} is greater than 1, then this parameter is mapped to 1 and the evaluated segment has a high load. Otherwise, S_{load} is in the range (0, 1). If the value of S_{load} approaches 0, then the segment load is low.

The relationship among Eqs. 8, 9, and 10 is used to calculate S_{event} as follows:

$$S_{event} = (\alpha_2 \times S_{den}) + (\beta_2 \times S_{Con}) - (\lambda \times S_{load}) \tag{11}$$

where, α_2, β_2, and λ are the weighting factors for S_{den}, S_{con}, and S_{load}, respectively. The value of S_{event} is in the range (0, 1) as $\alpha_2 = \beta_2 = \lambda = 0.33$.

RTISAR adopts two kinds of packet routing, namely, segment and intersection routing. In segment routing, the packets are forwarded between intersections, and the next forwarder S_veh is chosen from the same segment as shown in Eq. 7. When the packet reaches an intersection, RTISAR adopts intersection routing to choose the most suitable close segment for forwarding packets. The results for each close segment (excluding the previous segment) are computed, and higher values are assigned to those segments that are located near the destination. A segment result (S_{result}) metric is calculated based on the declarer S_{event} as shown in Eq. 11. The cumulative distance toward the destination (C_{dis}) is an essential parameter in position routing that measures the closeness of a packet to the destination if the packet is forwarded through the segment under consideration. In RTISAR, C_{dis} is calculated as follows [35], [39]:

$$C_{dis} = 1 - \frac{D_{can-dest}}{D_{cur-dest}} \tag{12}$$

where $D_{cur-dest}$ and $D_{can-dest}$ denote the distance between the current intersection and the destination as well as the distance between the candidate intersection and the destination, respectively. The packet that is forwarded through the segment under consideration will reach the candidate intersection. S_{result} can be computed as follows:

$$S_{result} = (\alpha_3 \times S_{event}) + (\beta_3 \times C_{dis}) \tag{13}$$

where α_3 and β_3 are the weighting factors for the result of the two segments. The value of S_{result} is in the range (0, 1) as $\alpha_3 = \beta_3 = 0.5$. Intersection routing is reapplied to ensure that the packets are delivered to the next intersection. Consequently, RTISAR continuously performs segment and intersection routing to forward the packets toward the destination. Algorithm 3 illustrates the D-DPR phase.

IV. Performance Evaluation

The performance of RTISAR has been evaluated under an urban simulation environment using the MATLAB simulator. The simulation results are analyzed based on several performance metrics to assess the capability and efficiency of the proposed algorithm. RTISAR is then compared with iCAR and TFOR, which are implemented in the libraries of the MATLAB simulator to run all protocols on the same platform and under the same conditions and simulation parameters. Afterward, the effectiveness of RTISAR when both traffic and segment information is considered is also tested.

Algorithm 3 D-DPR *Phase*

Input: *CP results information*
Output: *Forward the data packet to the destination*

1. ***Begin Algorithm 3***
2. *Evaluates* S_{result} *and* C_{dis}*;*
3. *Calculates* S_{result}*;*
4. *Forward the data packet toward the intersection with the highest* S_{result}*;*
5. ***If*** *the data packet reaches the destination* ***Then*** *Go to 8;*
6. ***Else***
7. *Run the command line 2 until the line 15 of Algorithm 1;*
8. ***End Algorithm 3***

Several models were implemented to assess the proposed algorithms of routing. Thus, each of the utilized models is briefly described in this section to clarify the concept.

- Wi-Fi model: To implement such applications in a vehicular environment, several protocols should be induced to take full advantage of the communication between V2V. 802.11 is a set of institute for electrical and electronic engineers (IEEE) standards for WLAN. IEEE 802.11 is better known as Wi-Fi, which is an acronym of wireless fidelity. IEEE 802.11p is the most relevant to this study. IEEE 802.11p includes the WAVE protocol required to support ITS applications [48], [49]. IEEE 802.11p primarily focuses improving vehicle safety and preventing traffic congestions. The major issues with using the original IEEE 802.11 standard in VANETs is that the nodes are highly mobile, implying that two vehicles will be connected for a very short time. Hence, special features, such as the WAVE mode, are embedded in IEEE 802.11p. Instant communications between two vehicles in contact is allowed by the WAVE mode. This situation becomes possible by assigning a special basic service set identification (BSSID) to the vehicles, which is called as wildcard BSSID. The same wildcard BSSID is given to all the vehicles so that communication is started immediately upon coming into contact with each other. Moreover, IEEE 802.11p is half the IEEE 802.11a/g channel bandwidth, which also translates to half the data rates. The benefit of this is the doubling of IEEE 802.11p time-domain parameters such as guard interval, symbol time, PLCP preamble and header. Longer guard interval gives higher tolerance to inter-symbol interference from multipath propagation delay spread. Thus, a more severe vehicular operating environment with higher excess delay can be supported.
- Mobility model: The heightened pace at which wireless networks are being transformed throughout the world is the outcome of many different factors. Given that mobile users constantly demand remote access, mobility is an important driving factor for mobile networks. Node speed defines the mobility of nodes in a mobile multi-hop wireless network. The change in the topology of the network, link failure rate, and routing overhead are indicated by the average speed of network nodes. Mobility models help to achieve realistic network scenarios, as they allow the acceleration or deceleration of simulated environments as well as direction's changing nodes [50]. The car following model (CFM) [6] is a widely used mobility model in the simulation of VANETs due to its strengths such as simple design, ensures that no contact is established with the leading vehicle by using a set of rules that help to adjust the mobility of the following node. In addition, it provides a vehicular traffic framework that can help to maintain distance- and time-headway between vehicles within safe limits to avoid accidents by controlling the driving parameters. Therefore, this study uses the CFM as a mobility model.
- Propagation channel model: The most accurate means of predicting the path loss in simulations is using a 3D ray-tracing technique combined with accurate models of the geometry and RF properties of the environment. The accuracy of the ray-tracer depends both on the number of rays simulated and the availability of accurate information about the environment. In an urban environment, this information should include characteristics such as the permittivity, permeability, reflectivity, and diffractively of building materials, roads, and obstacles such as vehicles. For a given wavelength and resolution of geometry and material properties, the accuracy of the simulation asymptotically increases with an increased number of rays. In this study, a log-distance path-loss model [51] uses as a propagation channel due to its popularity in the VANET literature and its suitability for urban environments.
- Network model: A graph G (V, E) is used to model VANETs, where mobile devices are denoted by V and the arcs' set is denoted by E. The communication range intersection between two devices is modeled by the arcs. Direct communication can occur between every node $v \in V$ and in-range neighboring devices, whereas the routing protocols are used for transmitting packets to devices that are not in the communication range. Packets can be transmitted to the destination from node v by choosing any of the probable paths in the graph G (V, E). Network topology changes frequently due to mobility of the nodes. Hence, the same cardinality of the nodes V is obtained all over the period, while changes are observed in that of the edges E. The average hop length E increases due to the increase in the node set density V, which affects the performance of the routing protocol. Hence, the most important simulation parameter in evaluating the routing protocols in VANETs is the size of the network. In the current study, different numbers of nodes were involved in the simulation scenarios to analyze the effect of the network size parameter on the performance of the proposed algorithm.
- Traffic model: In conducting a simulation study, the type of the traffic pattern significantly affects the routing protocols' performance. Furthermore, a network's congestion level is affected by certain modeling parameters, such as the number of flows and the effective transmission

TABLE III
SIMULATION PARAMETERS FOR RTISAR, ICAR, AND TFOR

Parameter	Number of Value
Simulation area	2km×2km
Intersections	24
Segments	38
Number of nodes	100, 200, 300, 400, 500
Vehicular speed	35–60 km/h
Vehicular density	High density 40-50 veh/km/lane and low density 6-8 veh/km/lane
Simulation duration	600 s
Wireless radio range	300m
Bit rate	6 Mbps (QPSK 1/2)
Channel bandwidth	10 MHz
Data packet size	512 bytes
Control packet size	64 bytes
Data traffic model	2-16 CBR connections
Data packet sending rate	0.1 s
Beacon interval	1 s

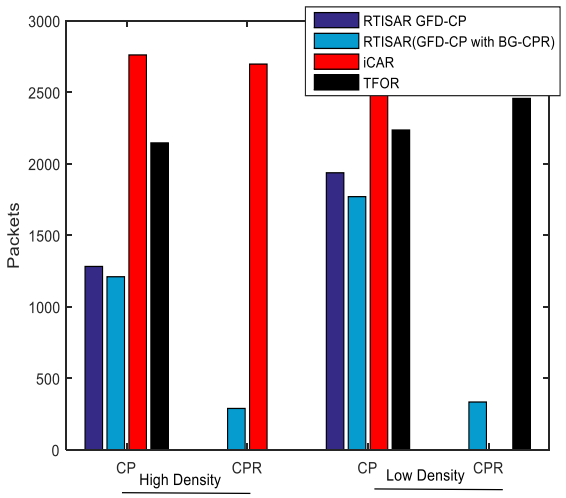


Fig. 6. Number of CPs and CPRs generated for low and high density.

rate. The effective transmission rate is calculated by dividing a measured number of data units (bits or packets) transmitted during an important measurement time interval by the time interval used for measurement in this study. Moreover, several traffic patterns, such as voice over internet protocol, file transfer protocol, and variable bit rate, can be used to connect source–destination pairs in VANET scenarios. In this study, the constant bit rate (CBR) user datagram protocol application was used. This protocol allows the creation of traffic patterns at a fixed rate via transmission of fixed-sized packets. Usually, the CBR provides the background traffic that suits the statistical features of various networking applications. Furthermore, several applications that require a predictable response time and the continuous availability of a fixed amount of bandwidth can be simulated using a CBR. These applications comprise services such as videoconferencing and telephony (voice services). Therefore, various traffic-related parameters, such as the effective transmission rate and number of flows, were considered in evaluating the performance of the proposed algorithm.

A. Simulation Environment and Parameters

An urban environment (Manhattan grid) is used as the simulation scenario. The simulation map has an area of 2 km×2 km with 38 bidirectional segments and 24 intersections. The parameter values are assigned on the basis of the values used in [34], [35], [39], and [52]. TABLE III summarizes the key parameters in the simulation.

B. Results and Discussion

The number of CPs and CPRs that are generated and forwarded must be investigated to show how the GFD-CP and BG-CPR phases affect the evaluation of traffic and segment status process performance. The data packets must not be transmitted between vehicles. Figure. 6 shows that both RTISAR GFD-CP and RTISAR (GFD-CP with BG-CPR) significantly contribute to reducing traffic and measuring segment status. Although RTISAR GFD-CP does not implement the CPR procedure, this method produces 1281 and 1936 CPs in high- and low-density scenarios, respectively. CPRs must be generated to make the results available at both segments. Consequently, the CPRs are generated using RTISAR (GFD-CP with BG-CPR) in some cases, and the generation of CPs is suppressed. RTISAR (GFD-CP with BG-CPR) clearly reduces the number of CPs to 1209 and 1769 as well as generates 288 and 333 CPRs in high- and low-density scenarios, respectively. To consider benchmarking, RTISAR GFD-CP is compared with TFOR because both of these protocols do not follow the CPR procedure. Figure 6 shows that RTISAR GFD-CP generates fewer CPs compared with TFOR. Specifically, RTISAR GFD-CP reduces the CPs to 40.2% in the high-density scenario, but only to 13.3% in the sparse density scenario. This finding may be attributed to the fact that more CPs are generated in the sparse scenario than in the dense scenario because density and network intermittent connectivity are highly varied in the former. Therefore, evaluation traffic segment process increased in sparse density. By contrast, RTISAR (GFD-CP with BG-CPR) is compared with iCAR to evaluate the traffic and segment status measurement process while following the CPR procedure. Figure 6 shows the number of CPs and CPRs that are generated by RTISAR (GFD-CP with BG-CPR) and iCAR in the two considered scenarios. RTISAR (GFD-CP with BG-CPR) generates 56.2% less CPs and 89.3% less CPRs in the high-density scenario, thereby reducing the total packets generation by up to 72.7%. In the sparse scenario, RTISAR (GFD-CP with BG-CPR) reduces the total packets generated by 57.9% and reduces the generated CPs and CPRs by 29.5% and 86.4%, respectively. Therefore, RTISAR (GFD-CP with BG-CPR) has a remarkable influence on the reduction of CPs and CPRs and subsequently affects CO.

Figure 7 shows the performance of RTISAR, iCAR, and TFOR in the low-density scenario when the number of connections of CBR increases from 2 to 16. RTISAR obtains a higher

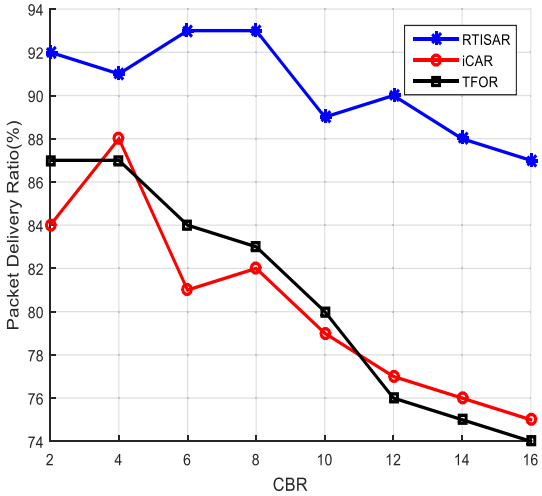


Fig. 7. Packet delivery ratio vs. CBR connections in low density.

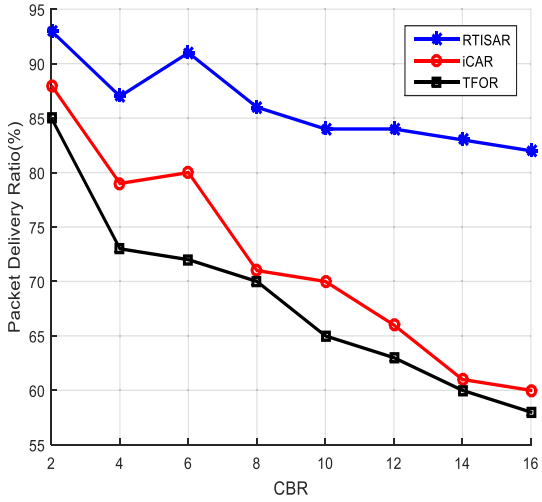


Fig. 8. Packet delivery ratio vs. CBR connections in high density.

packet delivery ratio (PDR) than iCAR and TFOR. Specifically, RTISAR obtains 92% PDR for 2 CBR connections and 87% PDR for 16 CBR connections. By contrast, iCAR and TFOR obtain 84% and 87% PDR for 2 CBR connections, but only obtain 75% and 74% PDR for 16 CBR connections. Figure 7 implies that when the CBR has a large number of connections, RTISAR outperforms both iCAR and TFOR. Specifically, for 2 CBR connections, the PDR of RTISAR is 9.5% and 5.7% higher than that of iCAR and TFOR, respectively. Moreover, for 16 CBR connections, the PDR of RTISAR is 16% and 17.5% higher than that of iCAR and TFOR, respectively. Therefore, RTISAR increases the total PDR by 12.7% and 11.6% for iCAR and TFOR by reducing its evaluation segment status process and suppressing the unwanted generation of CPs and CPRs.

Figure 8 presents the PDR when the three routings in the high-density scenario have varied CBR connections. RTISAR outperforms both iCAR and TFOR in terms of PDR. Specifically, RTISAR, iCAR, and TFOR deliver 93%, 88%, and 85% of the sent packets for 2 CBR connections, and deliver

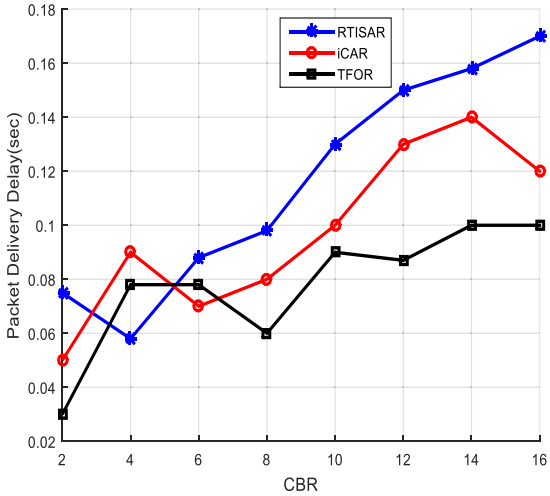


Fig. 9. Packet delivery delay vs. CBR connections in low density.

82%, 60%, and 58% of the packets for 16 CBR connections, respectively. In the high-density scenario, the PDR of RTISAR is 5.6% and 9.4% higher than that of iCAR and TFOR for 2 CBR connections and 36.6% and 41.3% higher for 16 CBR connections. As a result, RTISAR increases the total PDR by 21.2% and 25.3% for iCAR and TFOR, respectively. Suppressing the unwanted generation of CPs and CPRs significantly contributes to the reduction of the RTISAR evaluation segment status procedure. The inevitable signal interference causes collisions and packet losses, which in turn result in the below average performance of the three routings in the high-density scenario. However, signal interference and contention-based transmission have minimal effects on RTISAR because the mechanism of this algorithm reduces the occurrence of CO.

Figure 9 shows that the packet delivery delay (PDD) of RTISAR is higher than that of iCAR and TFOR in the low-density scenario. Specifically, RTISAR obtains a PDD of 0.17 seconds, while iCAR and TFOR obtain a PDD of less than 0.12 seconds for 16 CBR connections. This finding may be attributed to the fact that RTISAR chooses high connectivity segments even if these segments are located far away from the destination and cause high delivery delays. Consequently, RTISAR obtains a high PDR by slightly extending the delivery delay as shown in Figure 7.

In a scenario with high density and large number of CBR connections, the PDD of RTISAR significantly decreases compared to that of iCAR and TFOR as shown in Figure 10. Specifically, the PDD of RTISAR, iCAR, and TFOR reache 0.08, 0.07, and 0.15 seconds for 2 CBR connections, respectively. In other words, the PDD of RTISAR is 14.2% higher than that of iCAR and is 46.6% lower than that of TFOR for 2 CBR connections. However, the PDDs of iCAR and TFOR significantly increase when a large number of CBR connections is present and as the influence of high CO—resulting from the generation of CPs and CPRs—becomes more prominent. In this case, the PDR of both protocols decreases. By contrast, the PDD of RTISAR does not exhibit a rapid increase even with a larger number of CBR connections because the traffic segment procedure of this

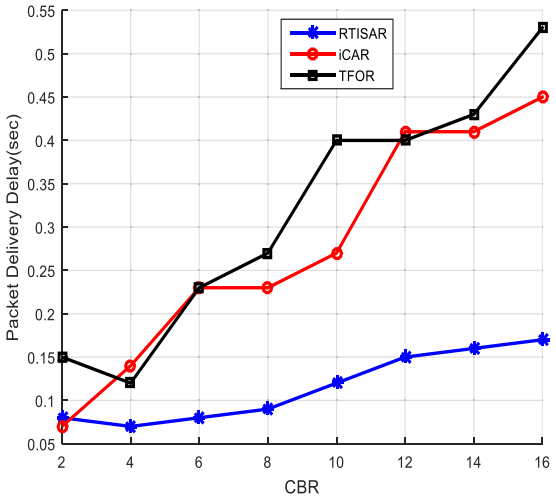


Fig. 10. Packet delivery delay vs. CBR connections in high density.

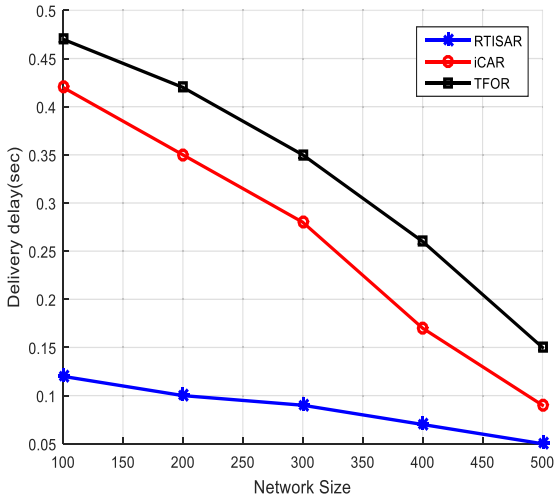


Fig. 12. Packet delivery delay vs. network size.

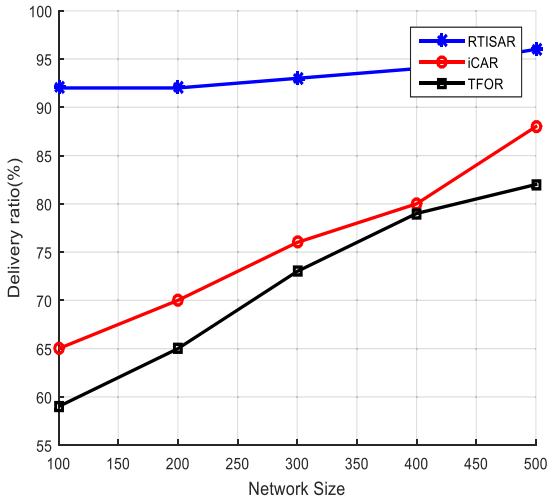


Fig. 11. Packet delivery ratio vs. network size.

protocol generates a low overhead. For 16 CBR connections, the PDD of RTISAR reaches 0.17 seconds, while that of iCAR and TFOR gradually increase to 0.45 and 0.53 seconds, respectively. These longer delays in sending packets can be attributed to contention-based transmissions. Therefore, the PDD of RTISAR is 62.2% and 67.9% lower than that of iCAR and TFOR for 16 CBR connections. The RTISAR then decreases the total PDD by 24% and 57.2% for iCAR and TFOR, respectively.

Figure 11 describes the performance of RTISAR, iCAR, and TFOR in terms PDR when the number of nodes increases from 100 to 500. RTISAR achieves the highest PDR (92% and 96%) for 100 and 500 nodes, respectively. In this algorithm, the route is selected progressively in consideration of the connectivity, lifetime, density, and load of the segments. Reducing the evaluation segment process of RTISAR by suppressing the unwanted generation of CPs and CPRs can also result in lower CO. By contrast, the PDRs of iCAR and TFOR increase from 65% to 88% and from 59% to 82% for 100 nodes. Those segments with a large number of nodes experience high connectivity and successful movement of packets toward the destination. However, in iCAR and TFOR, depending on density as a routing metric for selecting the best segment for forwarding packets does not essentially guarantee a satisfactory routing performance because the network loss and signal interference resulting from high density can increase packet loss. Moreover, the failure to consider the lifetime metric can forward the data via segments with increased levels of intermittent communications. The increased number of unnecessary CPs and CPRs can also increase CO. Therefore, iCAR and TFOR demonstrate a lower performance than RTISAR. Figure 11 shows that the PDR of RTISAR is 41.5% and 9% higher than that of iCAR and is 55.9% and 17% higher than that of TFOR for 100 and 500 nodes, respectively. As a result, RTISAR contributes to 25.2% and 36.4% of the increase in the total PDR for iCAR and TFOR, respectively.

Figure 12 shows that the PDD of RTISAR has significantly decreased compared with that of iCAR and TFOR for a large number of nodes. Specifically, for 100 nodes, the PDD of RTISAR, iCAR, and TFOR reach 0.12, 0.42, and 0.47 seconds, respectively. Those scenarios with a low number of nodes experience a high network intermittent connectivity, thereby increasing PDD and packet loss as shown in Figure 11. In this case, the PDD of RTISAR is 71.4% and 74.4% lower than that of iCAR and TFOR for 100 nodes. Moreover, the PDD of RTISAR decreases when a large number of nodes is present because the connectivity is increased, thereby increasing PDR and reducing delay. Specifically, the PDD of RTISAR, iCAR, and TFOR reaches 0.05, 0.09, and 0.15 seconds for 500 nodes. Therefore, the PDD of RTISAR is 44.4% and 66.6% lower than that of iCAR and TFOR for 500 nodes. RTISAR contributes to 57.9% and 70.5% of the decrease in the total PDD for iCAR and TFOR, respectively.

Figure 13 describes the performance of RTISAR, iCAR, and TFOR in CO as the number of nodes increases. The number of nodes increases along with CO because the control packet rate is directly related to the number of nodes. The control packets comprise CPs, CPRs, and a beacon. RTISAR

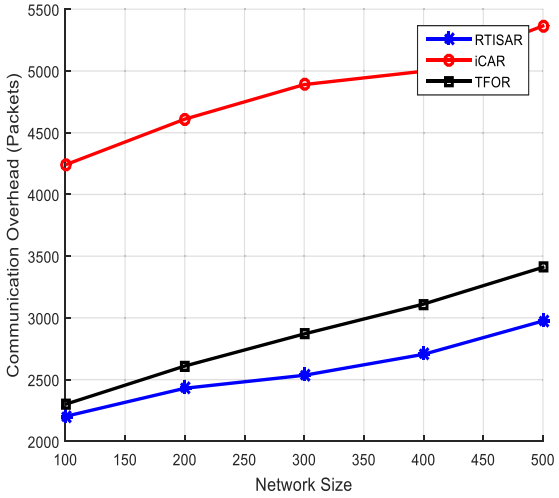


Fig. 13. Communication overhead vs. network size.

obtains the lowest CO for all cases. Specifically, RTISAR obtains 2202 and 2975 packets when 100 and 500 nodes are considered, respectively. However, RTISAR has a beacon, CPs, and CPRs as its control packets, all of which increase CO. Nevertheless, RTISAR can reduce and control unnecessary CPs and CPRs unlike iCAR and TFOR. Therefore, RTISAR outperforms both iCAR and TFOR in terms of CO. By contrast, iCAR and TFOR obtain 4240 and 2300 control packets for 100 nodes, and obtain 5365 and 3410 control packets for 500 nodes. TFOR only has a beacon and CPs as its control packets, while iCAR has a beacon, CPs, and CPRs as its control packets. Therefore, TFOR has a lower CO than iCAR. Figure 13 shows that the CO of RTISAR is 48% and 44% lower than that of iCAR and is 4.2% and 12.7% lower than that of TFOR for 100 and 500 nodes, respectively. As a result, RTISAR contributes to 46% and 8.4% of the reductions in the CO for iCAR and TFOR, respectively.

V. Conclusion

The paper presented the assumption, description, phases, and simulation results of the performance evaluation of RTISAR, an intersection-based routing algorithm that provides the optimal route for forwarding data packets toward the destination by considering the traffic segment status when choosing the next intersection. S_{result} was used to assess the segment status based on connectivity, density, load segment, and cumulative distance toward the destination. A V_period mechanism in the GFD-CP phase and segment status computation process in necessary conditions in BG-CPR phase were used to minimize the CO generated. The simulation results of RTISAR were analyzed and compared with those of iCAR and TFOR. The influence of the number of flows with different density scenarios and network sizes on the performance of various routings was studied by conducting extensive simulations. The evaluation results from the studied simulation scenarios showed that RTISAR outperformed both iCAR and TFOR in terms of PDR, PDD, and CO.

References

[1] R. Jalali, K. El-Khatib, and C. McGregor, "Smart city architecture for community level services through the Internet of Things," in *Proc. 18th Int. Conf. Intell. Next Generat. Netw. (ICIN)*, 2015, pp. 108–113.

[2] A. Ghazal, C.-X. Wang, B. Ai, D. Yuan, and H. Haas, "A nonstationary wideband MIMO channel model for high-mobility intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 885–897, Apr. 2015.

[3] S. A. A. Shah, M. Shiraz, M. K. Nasir, and R. B. M. Noor, "Unicast routing protocols for urban vehicular networks: Review, taxonomy, and open research issues," *J. Zhejiang Univ. Sci. C*, vol. 15, no. 7, pp. 489–513, 2014.

[4] Y. Chen, M. Fang, S. Shi, W. Guo, and X. Zheng, "Distributed multi-hop clustering algorithm for VANETs based on neighborhood follow," *EURASIP J. Wireless Commun. Netw.*, vol. 1, p. 98, Apr. 2015.

[5] S. Misra, I. Woungang, and S. C. Misra, *Guide to Wireless Ad Hoc Networks*. London, U.K.: Springer, 2009.

[6] H. Hartenstein and K. Laberteaux, *VANET: Vehicular Applications and Inter-Networking Technologies*, vol. 1. Hoboken, NJ, USA: Wiley, 2009.

[7] W. Chen, R. K. Guha, T. J. Kwon, J. Lee, and Y.-Y. Hsu, "A survey and challenges in routing and data dissemination in vehicular ad hoc networks," *Wireless Commun. Mobile Comput.*, vol. 11, no. 7, pp. 787–795, Jul. 2011.

[8] A. Fonseca and T. Vazão, "Applicability of position-based routing for VANET in highways and urban environment," *J. Netw. Comput. Appl.*, vol. 36, no. 3, pp. 961–973, 2013.

[9] M. C. Weigle and S. Olariu, *Vehicular Networks: From Theory to Practice*. London, U.K.: Chapman & Hall, 2009.

[10] M. R. Jabbarpour, R. M. Noor, R. H. Khokhar, and C.-H. Ke, "Cross-layer congestion control model for urban vehicular environments," *J. Netw. Comput. Appl.*, vol. 44, pp. 1–16, Sep. 2014.

[11] Y. R. B. Al-Mayouf, M. Ismail, N. F. Abdullah, S. M. Al-Qaraawi, and O. A. Mahdi, "Survey on Vanet technologies and simulation models," *ARPN J. Eng. Appl. Sci.*, vol. 11, no. 15, pp. 9414–9427, 2016.

[12] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *J. Netw. Comput. Appl.*, vol. 37, pp. 380–392, Jan. 2014.

[13] R. Chen, W.-L. Jin, and A. Regan, "Broadcasting safety information in vehicular networks: Issues and approaches," *IEEE Netw.*, vol. 24, no. 1, pp. 20–25, Jan./Feb. 2010.

[14] H. Moustafa and Y. Zhang, *Vehicular Networks: Techniques, Standards, and Applications*. New York, NY, USA: Auerbach, 2009.

[15] N. Kumar, N. Chilamkurti, and J. J. P. C. Rodrigues, "Learning automata-based opportunistic data aggregation and forwarding scheme for alert generation in vehicular ad hoc networks," *Comput. Commun.*, vol. 39, no. 2, pp. 22–32, 2014.

[16] A. Caragliu, C. del Bo, and P. Nijkamp, "Smart cities in europe," *J. Urban Technol.*, vol. 18, no. 2, pp. 65–82, 2011.

[17] M. H. Eiza and Q. Ni, "An evolving graph-based reliable routing scheme for VANETs," *IEEE Trans. Veh. Technol.*, vol. 62, no. 4, pp. 1493–1504, May 2013.

[18] Y. R. B. Al-Mayouf *et al.*, "Efficient and stable routing algorithm based on user mobility and node density in urban vehicular network," *PLoS ONE*, vol. 11, no. 11, p. e0165966, 2016.

[19] B. T. Sharef, R. A. Alsaqour, and M. Ismail, "Vehicular communication ad hoc routing protocols: A survey," *J. Netw. Comput. Appl.*, vol. 40, pp. 363–396, Apr. 2014.

[20] C. Chen, Y. Jin, Q. Pei, and N. Zhang, "A connectivity-aware intersection-based routing in VANETs," *EURASIP J. Wireless Commun. Netw.*, vol. 1, p. 42, Dec. 2014.

[21] F. Malandrino, C. Casetti, C.-F. Chiasserini, and M. Fiore, "Optimal content downloading in vehicular networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 7, pp. 1377–1391, Jul. 2013.

[22] H. Wu *et al.*, "An empirical study of short range communications for vehicles," in *Proc. 2nd ACM Int. Workshop Veh. Ad Hoc Netw.*, 2005, pp. 83–84.

[23] Y.-A. Daraghmi, C.-W. Yi, and I. Stojmenovic, "Forwarding methods in data dissemination and routing protocols for vehicular Ad Hoc networks," *IEEE Netw.*, vol. 27, no. 6, pp. 74–79, Nov./Dec. 2013.

[24] M. R. Jabbarpour, A. Marefat, A. Jalooli, R. M. Noor, R. H. Khokhar, and J. Lloret, "Performance analysis of V2V dynamic anchor position-based routing protocols," *Wireless Netw.*, vol. 21, no. 3, pp. 911–929, 2015.

[25] K.-H. Cho and M.-W. Ryu, "A survey of greedy routing protocols for vehicular ad hoc networks," *SmartCR*, vol. 2, pp. 125–137, Apr. 2012.

[26] K. Z. Ghafoor, M. A. Mohammed, J. Lloret, K. A. Bakar, and Z. M. Zainuddin, "Routing protocols in vehicular ad hoc networks: Survey and research challenges," *Netw. Protocols Algorithms*, vol. 5, no. 4, pp. 39–83, 2013.

[27] O. A. Mahdi *et al.*, "A comparison study on node clustering techniques used in target tracking WSNs for efficient data aggregation," *Wireless Commun. Mobile Comput.*, vol. 16, no. 16, pp. 2663–2676, 2016.

[28] S. M. Bilal, C. J. Bernardos, and C. Guerrero, "Position-based routing in vehicular networks: A survey," *J. Netw. Comput. Appl.*, vol. 36, no. 2, pp. 685–697, 2013.

[29] M. Altayeb and I. Mahgoub, "A survey of vehicular ad hoc networks routing protocols," *Int. J. Innov. Appl. Stud.*, vol. 3, no. 3, pp. 829–846, 2013.

[30] Y. R. B. Al-Mayouf, N. F. Abdullah, M. Ismail, S. M. Al-Qaraawi, O. A. Mahdi, and S. Khan, "Evaluation of efficient vehicular ad hoc networks based on a maximum distance routing algorithm," *EURASIP J. Wireless Commun. Netw.*, vol. 1, p. 265, Dec. 2016.

[31] T. Darwish and K. A. Bakar, "Traffic aware routing in vehicular ad hoc networks: Characteristics and challenges," *Telecommun. Syst.*, vol. 61, no. 3, pp. 489–513, 2016.

[32] O. A. Mahdi, A. W. A. Wahab, M. Y. I. Idris, A. A. Znaid, Y. R. B. Al-Mayouf, and S. Khan, "WDARS: A weighted data aggregation routing strategy with minimum link cost in event-driven WSNs," *J. Sensors*, vol. 2016, May 2016, Art. no. 3428730.

[33] K. N. Qureshi, A. H. Abdullah, J. Lloret, and A. Altameem, "Road-aware routing strategies for vehicular ad hoc networks: Characteristics and comparisons," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 3, p. 1605734, 2016.

[34] M. Jerbi, S. M. 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.

[35] S. M. Bilal, S. A. Madani, and I. A. Khan, "Enhanced junction selection mechanism for routing protocol in VANETs," *Int. Arab J. Inf. Technol.*, vol. 8, no. 4, pp. 422–429, 2011.

[36] I. A. Abbasi, B. Nazir, A. Abbasi, S. M. Bilal, and S. A. Madani, "A traffic flow-oriented routing protocol for VANETs," *EURASIP J. Wireless Commun. Netw.*, vol. 1, p. 121, Dec. 2014.

[37] C. Zhao, C. Li, L. Zhu, H. Lin, and J. Li, "A vehicle density and load aware routing protocol for VANETs in city scenarios," in *Proc. Int. Conf. Wireless Commun. Signal Process. (WCSP)*, 2012, pp. 1–6.

[38] C. Li, C. Zhao, L. Zhu, H. Lin, and J. Li, "Geographic routing protocol for vehicular ad hoc networks in city scenarios: A proposal and analysis," *Int. J. Commun. Syst.*, vol. 27, no. 12, pp. 4126–4143, 2014.

[39] Y. He, C. Li, X. Han, and Q. Lin, "A link state aware hierarchical road routing protocol for 3D scenario in VANETs," in *Proc. Int. Conf. Internet Veh.*, 2014, pp. 11–20.

[40] N. Alsharif, S. Céspedes, and X. Shen, "iCAR: Intersection-based connectivity aware routing in vehicular ad hoc networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 1736–1741.

[41] N. Alsharif and X. S. Shen, "iCARII: Intersection-based connectivity aware routing in vehicular networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 2731–2735.

[42] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (VANETs): Status, results, and challenges," *Telecommun. Syst.*, vol. 50, no. 4, pp. 217–241, 2012.

[43] H. Saleet, O. Basir, R. Langar, and R. Boutaba, "Region-based location-service-management protocol for VANETs," *IEEE Trans. Veh. Technol.*, vol. 59, no. 2, pp. 917–931, Feb. 2010.

[44] M. Hadded, P. Muhlethaler, R. Zagrouba, A. Laouiti, and L. A. Saidane, "Using road IDs to enhance clustering in vehicular ad hoc networks," in *Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2015, pp. 285–290.

[45] N. Brahmi, M. Boussedjra, J. Mouzna, and M. Bayart, "Adaptative movement aware routing for vehicular ad hoc networks," in *Proc. Int. Conf. Wireless Commun. Mobile Comput., Connecting World Wirelessly*, 2009, pp. 1310–1315.

[46] K. Shafiee and V. C. M. Leung, "Connectivity-aware minimum-delay geographic routing with vehicle tracking in VANETs," *Ad Hoc Netw.*, vol. 9, no. 2, pp. 131–141, 2011.

[47] D. S. J. de Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Netw.*, vol. 11, no. 4, pp. 419–434, 2005.

[48] Q. Yang, A. Lim, S. Li, J. Fang, and P. Agrawal, "ACAR: Adaptive connectivity aware routing for vehicular ad hoc networks in city scenarios," *Mobile Netw. Appl.*, vol. 15, no. 1, pp. 36–60, Feb. 2010.

[49] D. Eckhoff, N. Sofra, and R. German, "A performance study of cooperative awareness in ETSI ITS G5 and IEEE WAVE," in *Proc. 10th Annu. Conf. Wireless On-Demand Netw. Syst. Serv. (WONS)*, Mar. 2013, pp. 196–200.

[50] B. Schweiger, C. Raubitschek, B. Bäker, and J. Schlichter, "ElisaTM–Car to infrastructure communication in the field," *Comput. Netw.*, vol. 55, pp. 3169–3178, Oct. 2011.

[51] E. Spaho, L. Barolli, G. Mino, F. Xhafa, and V. Kolici, "VANET simulators: A survey on mobility and routing protocols," in *Proc. Int. Conf. Broadband Wireless Comput., Commun. Appl. (BWCCA)*, 2011, pp. 1–10.

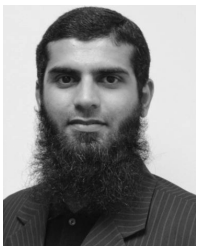
[52] P. K. Sharma and R. Singh, "Comparative analysis of propagation path loss models with field measured data," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 6, pp. 2008–2013, 2010.

[53] Y. R. B. Al-Mayouf, N. F. Abdullah, M. Ismail, A. W. A. Wahab, and O. A. Mahdi, "Efficient routing algorithm for VANETs based on distance factor," in *Proc. Int. Conf. Adv. Electr., Electron. Syst. Eng. (ICAEES)*, Nov. 2016, pp. 567–571.

Yusor Rafid Bahar Al-Mayouf was born in Baghdad, Iraq, in 1980. She received the B.Sc. degree in systems and control engineering from University of Technology, Iraq, in 2002, the M.Sc. degree in mechatronics from University of Siegen, Germany, in 2007, and the Ph.D. degree in electrical, electronic, and systems engineering from Universiti Kebangsaan Malaysia, Malaysia, in 2017. She is currently a Lecturer in networks and communications engineering with the Department of Computer Science, College of Education for Pure Science (Ibn Al-Haitham), University of Baghdad, Iraq. Her main research interests include wireless networking communications.

Nor Fadzilah Abdullah received the B.Sc. degree in electrical and electronics from Universiti Teknologi Malaysia in 2001, the M.Sc. degree (Hons.) in communications engineering from The University of Manchester, U.K., in 2003, and the Ph.D. degree in electrical and electronic engineering from University of Bristol, U.K., in 2012. She was with major telecommunication companies, such as Ericsson, Malaysia, and mobile operator Maxis Communications Berhad, Malaysia, from 2003 to 2008. She was a Researcher with the Communication, System and Network Group, University of Bristol, from 2008 to 2014. She is currently a Senior Lecturer with Universiti Kebangsaan Malaysia, Selangor, Malaysia, and an Honorary Staff with the University of Bristol. She has been involved in a number of collaboration between the university and a number of industries on various projects related to her research area. Her research interests include 5G, millimeter wave, vehicular networks, MIMO, space time coding, fountain code, channel propagation modelling, and estimation.

Omar Adil Mahdi was born in Baghdad, Iraq, in 1980. He received the bachelor's degree in computer science from University of Baghdad, Iraq, in 2003, the M.S. degree in computer science from University of Al-Anbar, Iraq, in 2010, and the Ph.D. degree from the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia, in 2017. He is currently a Lecturer in networks and communications engineering with the Department of Computer Science, College of Education for Pure Science (Ibn Al-Haitham), University of Baghdad. His current research area is data aggregation and data routing in wireless sensor network, and wireless *ad hoc*, vehicular, sensor network, and next generation networks.



Suleman Khan (S'14) received the M.Sc. degree in computer science from University of Peshawar, Pakistan, in 2006, the M.B.A. degree in human resource development from the Institute of Management of Sciences, Hayatabad, Pakistan, in 2007, the M.S. degree in distributed systems from the COMSATS Institute of Information Technology, Abbottabad, Pakistan, in 2011, and the Ph.D. degree (Hons.) from the Faculty of Computer Science and Information Technology, University of Malaya, Malaysia, in 2017. He is currently a Faculty Member with the School of Information Technology, Monash University, Malaysia. He has published over 40 high-impact research articles in reputed international journals and conferences. His research interests include but are not limited to network security, network forensics, software-defined networks, Internet of Things, cloud computing, and vehicular communications.

Mahamod Ismail received the B.Sc. degree in electrical and electronics engineering from University of Strathclyde, U.K., in 1985, the M.Sc. degree in communications engineering and digital electronics from Institute of Science and Technology, The University of Manchester, Manchester, U.K., in 1987, and the Ph.D. degree from University of Bradford, U.K., in 1996. He is currently a Professor of communications engineering and the Head of the Department of the Electrical, Electronic, and Systems, Universiti Kebangsaan Malaysia. His current research interests include mobile communications and wireless networking. He is an Executive Member of the Communications/Vehicular Technology Society and the IEEE Malaysia Chapter.

Mohsen Guizani (S'85–M'89–SM'99–F'09) received the B.S. degree (Hons.) and the M.S. degree in electrical engineering and the M.S. and Ph.D. degrees in computer engineering from Syracuse University, Syracuse, NY, USA, in 1984, 1986, 1987, and 1990, respectively. He is currently a Professor and the ECE Department Chair with the University of Idaho, Moscow, ID, USA. He has authored nine books and over 500 publications in refereed journals and conferences. His research interests include wireless communications and mobile computing, computer networks, mobile cloud computing, security, and smart grid. He is a Senior Member of ACM. He served as a member, the chair, and the general chair of a number of international conferences. He received the Wireless Technical Committee's Recognition Award in 2017. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the TAOS Technical Committee. He served as the IEEE Computer Society Distinguished Speaker from 2003 to 2005. He was the Founder and an Editor-in-Chief of the *Wireless Communications and Mobile Computing* journal from 2000 to 2016. He guest edited a number of special issues in the IEEE journals and magazines. He is currently an Editor-in-Chief of IEEE NETWORK. He serves on the editorial boards of several international technical journals.

Syed Hassan Ahmed (SM'18) received the B.S. degree in computer science from Kohat University of Science and Technology, Pakistan, and the combined master's and Ph.D. degrees from the School of Computer Science and Engineering (SCSE), Kyungpook National University (KNU), South Korea. In 2015, he was also a Visiting Researcher with the Georgia Institute of Technology, Atlanta, GA, USA. He is currently a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL, USA. He has authored or co-authored over 100 international publications, including journal articles, conference proceedings, book chapters, and two books. His research interests include sensor and ad hoc networks, cyber-physical systems, vehicular communications, and future Internet. From 2014 to 2016, he was a recipient of the Research Contribution awards by SCSE at KNU. In 2016, his works on robust content retrieval in the future vehicular networks lead him to win the Qualcomm Innovation Award at KNU.