

Traffic-Aware VANETs Routing for City Environments - A Protocol Based on Ant Colony Optimization

Forough Goudarzi, *Member, IEEE*, Hamid Asgari, *Senior Member, IEEE*, and Hamed S. Al-Raweshidy, *Senior Member, IEEE*

Abstract—This paper presents a traffic-aware position-based routing protocol for vehicular ad hoc networks (VANETs) suitable for city environments. The protocol is an enhanced version of the Geographical Source Routing (GSR) protocol. The proposed protocol, named Efficient GSR (EGSR), uses an ant-based algorithm to find a route that has optimum network connectivity. It is assumed that every vehicle has a digital map of the streets comprised of junctions and street segments. Using information included in small control packets called ants, the vehicles calculate a weight for every street segment proportional to the network connectivity of that segment. Ant packets are launched by the vehicles in junction areas. In order to find the optimal route between a source and destination, the source vehicle determines the path on a street map with the minimum total weight for the complete route. The correct functionality of the proposed protocol has been verified and its performance has been evaluated in a simulation environment. The simulation results show that the packet delivery ratio is improved by more than 10% for speeds up to 70 km/h compared with VACO protocol that also uses ant-based algorithm. In addition the routing control overhead and end-to-end delay are also reduced.

Index Terms—ant colony, routing protocol, VANETs.

I. INTRODUCTION

Vehicular Ad hoc Network (VANET) is an emerging technology that aims to provide wireless communication between moving vehicles as well as between vehicles and infrastructure stations. The main motivation for use of VANETs is its potential for providing safety related information to vehicles. Vehicles exchange status information, such as speed, acceleration, and position in the periodic messages called beacons, to create awareness for surrounding vehicles, increase safety and reduce accidents. Diverse non-safety applications are also expected for VANETs, ranging from road traffic efficiency to commercial applications and infotainment, such as entertainment for road travelers making their journeys more pleasant [1]–[3].

VANETs have some characteristics that differentiate them from other types of mobile ad hoc networks (MANETs). These characteristics include fast node movements, a large network, and constrained mobility imposed by the road topology. Owing to such differences, topology-based MANET routing protocols, such as AODV [4], OLSR [5], and DSR [6], perform less efficiently in VANETs [7] [8]. These routing protocols use the broadcast mechanism to find and maintain routes. However, in VANETs, the movement of vehicles causes the communication links between vehicles to be broken frequently. Such link

failures increase the broadcasting and routing control overhead as well as leading to degradation of the protocol performance.

Position-based routing is an alternative approach for routing in VANETs. These protocols do not select a fixed set of nodes between the source and destination for routing packets and consequently do not suffer route instability. In position-based routing protocols, a greedy mechanism, such as GPSR [9], is used to forward packets. With such a mechanism, each node obtains its current location for example, through a GPS receiver, and learns the position of its one-hop neighbors by receiving periodic beacon messages. To route packets, a node sends them to the neighboring node that is nearest to the destination. This mechanism does not need route discovery and management and thus is more scalable and suitable for large and highly dynamic networks. However, there are problems if it is used for routing in VANETs in city environments. Such problems have been reported in [7] and [10] and include routing loops and incorrect directions. In order to overcome these problems, the GSR protocol [7] has been proposed. In GSR, the source node computes the shortest path to the destination by using Dijkstra's algorithm on a street map. The computed path consists of a sequence of junction IDs known as Anchor Points (AP). The list of junctions is then inserted into the header of each data packet. Using the list, source-based routing is used across junctions, while greedy-based routing is used for packet forwarding in the street segments between the junctions. The problem with GSR is that it does not consider the vehicular traffic conditions of the streets along a route to support connectivity. A number of traffic-aware or connectivity-aware routing protocols have been designed to address this issue [11]–[18]. These protocols are reviewed in the next section.

This paper proposes a position-based traffic-aware routing protocol for city environments. This protocol is called Efficient GSR (EGSR). It optimizes GSR for routing in a city environment with unevenly distributed vehicular traffic by adding traffic-awareness to GSR. Like GSR, EGSR uses the street map to compute the shortest path; however, the weight of every street segment is not the length of the street. Instead, the weights are computed and dynamically updated according to the connectivity conditions of the streets. In order to make the protocol aware of the traffic conditions of the street segments, it uses small control packets (ants) to sample traffic conditions and update vehicles routing information. The approach presented in this paper is based on ant colony

optimization (ACO).

Recently, bio-inspired networking approaches have received a great deal of interest due to their potential features such as scalability, adaptability, self-organization, robustness, and resilience to failures [19]. The architecture of bio-inspired solutions should implement key principles [20] [21] to achieve these desirable properties. Otherwise, their effectiveness or functionality might be limited [22] [23]. Thus, despite the similarities of some solutions to biological systems they fail to meet the objectives or achieve the advantages of them.

Among bio-inspired techniques, ant colony optimization has been widely used for routing in networks [15], [24]–[32]. Most of the proposed ant-based routing protocols try to find entire paths between network nodes. However, this method might not be suitable for VANETs due to rapid movement of the nodes. Some road-based ACO routing protocols have been proposed For VANET too [15], [32], [33] however they need road side units (RSUs) at every junction to keep routing information and find routes. This causes the protocols to miss two important characteristics of bio-inspired systems: self-organization and resilience to failure of specific nodes (RSUs). The main contributions of this paper are as follows:

- 1) A bio-inspired technique based on the ACO algorithm has been applied to add traffic awareness to the well-known GSR protocol.
- 2) In the resulting routing protocol, the network nodes obtain the required traffic information in a completely ad hoc manner without any central or dedicated control such as traffic sensors, road side units (RSUs), or information feed from outside the network.
- 3) The protocol is adaptive to traffic condition as the pheromone deposition and evaporation mechanisms guarantee adaptability of the protocol to traffic changes.

The remainder of this paper is organized as follows. In Section II the related works are reviewed and in Section III, the proposed protocol is described. The simulation scenario, an evaluation of the results and a comparison against the GSR and VACO (VANET routing protocol based on ACO) [15], [33] protocols are presented in Section IV. Finally, Section V concludes the paper.

II. RELATED WORK

A. Modified MANETs Routing Protocols

This class of routing protocols have been developed by some modifications to topology-based routing protocols to qualify them for routing in dynamic VANETs environments. In [34], using offline optimization, an attempt was made to optimize configuration parameters of OLSR [5] for VANETs. However, as the parameters were computed based on optimization for a selected scenario, the tuned OLSR did not perform better than the standard protocol in all scenarios. PFQ-AODV [35] is a routing protocol based on AODV that uses different metrics from hop counts. As the relative mobility of vehicles has been considered in the computation of the metric, the protocol selects links with a longer projected lifetime and therefore needs less route recovery than AODV. Nevertheless,

like AODV, it utilizes the broadcast mechanism for route finding and recovery.

VNAODV+ [36] is a cluster-based routing protocol based on AODV, which has been proposed to solve the high overhead and instability of the routes in AODV for high mobility scenarios. Only cluster leaders participate in route finding and route repair mechanisms, alleviating these problems. However, managing clusters creates extra overhead. The protocol introduces a layer between the link layer and network layer that is responsible for cluster management. The whole geographical area is divided into zones and when a vehicle enters or leaves a zone, control packets are issued to manage the clusters. Hybrid Location-Based Ad Hoc Routing (HLAR) [37] combines the features of geographical greedy forwarding and AODV to reduce the routing control overhead in the latter. Greedy forwarding is used instead of a broadcast mechanism to discover or repair a route. However, when it fails to find a neighbor node nearer to the destination for example, at a local maximum or when the location information degrades, the protocol returns to reactive routing and uses broadcast to recover from the situation.

B. Traffic-Aware Routing Protocols

To solve problems of position-based routing in city environments, a number of traffic-aware routing protocols have been proposed. Anchor-based Street and Traffic Aware Routing (A-STAR) [11], like GSR, uses Dijkstra's algorithm on a graph representing a city map to compute the shortest path between the source and destination. In order to select streets based on their connectivity, the weight of each edge (street) is assigned according to the number of bus services using that street. It assumes streets with more bus services have more traffic and therefore have lower weight. The problems with this approach are as follows: Firstly, it uses static information on city bus services, so the weights of streets are constant, while in reality traffic conditions can be very dynamic. Secondly, streets with higher traffic density are not always optimum paths because packet loss occurs not only due to low connectivity but also because of collisions along congested paths.

Spatial and Traffic-Aware Routing (STAR) [12] is another traffic-aware routing protocol with which vehicles detect abnormal traffic conditions (absence or a high number of neighbors) in four cardinal directions and then propagate this information in their beacons. In this protocol, streets with high traffic density are assigned a low weight and the weight of streets can change dynamically. However, the second problem highlighted above still exists. The algorithm tends to use more congested streets because it weighs the streets according to the number of nodes and not the packet relaying property of the streets. There is another problem with this protocol; using the propagated information, the street in which such a condition exists cannot be exactly determined.

Improved Greedy Traffic Aware Routing protocol (GyTAR) [13] is a traffic-aware routing protocol that relies on traffic sensors at every junction. Apart from the need for a large number of sensors, such protocols face problems such as limited coverage of detection equipment, deployment and

maintenance costs, as well as a great deal of time being consumed in collecting, processing, and disseminating traffic-related information [38]. Furthermore, due to the junction-by-junction routing approach, the protocol might not always find a path to a destination. Road-Based Vehicular Traffic routing (RBVT) [14] is another traffic-aware routing protocol that uses a beaconless mechanism in order to overcome network congestion.

Back-Bone-Assisted Hop Greedy routing (BAHG) [16] tries to select a path that minimizes the number of intermediate intersections that change the direction of the routing path, since these intersections cause more hops and poor connectivity due to buildings around them. It assigns a parameter to every street to signify its connectivity, which is then used to select connected paths. This parameter is computed based on the number of lanes, not real-time traffic conditions, and consequently, sometimes the calculated path encounters a void region. In such cases, the path should be recalculated from that point.

Intersection-Based Geographical Routing Protocol (IGRP) [17] is a routing protocol for forwarding data packets from vehicles to Internet gateways, under the assumption that the Internet gateways have up-to-date information on the position of all vehicles in their surrounding area. When a vehicle needs a route to forward data packets to the Internet gateway, it sends a request to it. The gateway then computes the intersections of the path and the required transmission range at each street segment that the vehicles should use to achieve high connectivity.

In Connectivity-Aware Routing (CAR) [18], the source broadcasts request messages to find a path to the destination. In order to estimate connectivity, every node forwarding the route request updates the hop count as well as the average and minimum number of neighbors. The destination decides the routing path and replies to the source. Despite CAR addressing connectivity issues, the gathered information on the number of nodes cannot ensure connectivity in individual road segments along a routing path, because the connectivity depends on both the number of nodes and their topology.

C. Ant-Based Routing Protocols

Mobility-Aware ant colony optimization routing DYMO (MAR-DYMO) [29] is a reactive routing protocol for VANETs that is a combination of Dynamic MANET On-demand (DYMO) [39] and ACO. DYMO itself is an improved version of AODV. Using vehicles' position and speed, it predicts their movements to find the path with the longest lifetime. MAR-DYMO has scalability problem as the performance of the protocol drops rapidly with increasing numbers of vehicles. It has higher overhead than DYMO, while the packet delivery is increased slightly. Also, because it is node-based, the overhead increases as the speed of vehicles increases due to greater link breakage and route recovery.

Trust Dependent Ant Colony Routing (TACR) [31] is a reactive ant-based routing protocol in which clusters of vehicles are created by considering direction, position, and relative speed of vehicles to manage the scalability of the protocol. Only

cluster heads contribute to launching ants and finding routes to decrease routing overhead. However, the simulation results did not show much improvement against MAR-DYMO, because managing the clusters creates overhead itself.

Mobility-Aware Zone-based Ant Colony Optimization Routing for VANET (MAZACORNET) [30] is a zone-based ACO routing for VANETs. By using ACO technique, it tries to select the links for routing that have higher lifetimes and quality. The link quality is estimated by using the Nakagami Fading Model. Interzone routing follows a proactive approach and intrazone routing is on demand. In terms of routing control overhead, it does not show much improvement against AODV due to the interzone proactive routing approach. Thus, like other node-based algorithms, it has scalability problem when used in VANETs.

VACO [15], [33] uses ant colony optimization to assess the packet-relaying quality of each street segment located between two junctions in terms of latency, bandwidth, and delivery ratio. It is assumed that there is an RSU at each junction to save routing information and find routes for packets. To set up a route, the source node forwards several ants toward a target RSU, which is the closest one to the destination vehicle. At the target RSU, backward ants are generated and sent back to the source. For route maintenance, VACO utilizes a proactive approach. Using RSU at every junction can be costly and might not be practical, at least during the initial deployment of VANETs. In addition this causes the protocol to miss two important characteristics of bio-inspired systems: self-organization and resilience to failure of specific nodes (RSUs).

AQRV [32] is a junction based QoS routing protocol where QoS metrics include connectivity probability, packet delivery ratio and delay. It is assumed that there is a static terminal intersection (TI) with Wi-Fi capability at every junction to store routing table, launch ants and relay data packets. At the beginning of data transfer a source vehicle sends its request to a TI (TIS), if TI has no route to the destination, launches a number of forward ants toward the destination TI (TID). At TID, forward ants are converted to the backward ants and are returned to the TIS. Pheromone table of every TI that the backward ants pass is updated based on the information that ants carry and the analytical expression develop for the three metrics. The analytical expression is based on the assumption that the streets are one-way. Then the routing protocol is tested on a one-way simulation scenario. Also it is assumed that the TIs know the vehicle density at each street.

III. EFFICIENT GEOGRAPHIC SOURCE ROUTING (EGSR) PROTOCOL

A. System Model

The assumptions to design the EGSR protocol are presented in this section. The wireless technology deployed for exchange of packets is dedicated short range communication (DSRC). DSRC uses IEEE 802.11p standard at the PHY and MAC layers. Each vehicle is equipped with a GPS receiver, digital map, and navigation system. Thus, vehicles are aware of their position through the GPS and can map their positions on

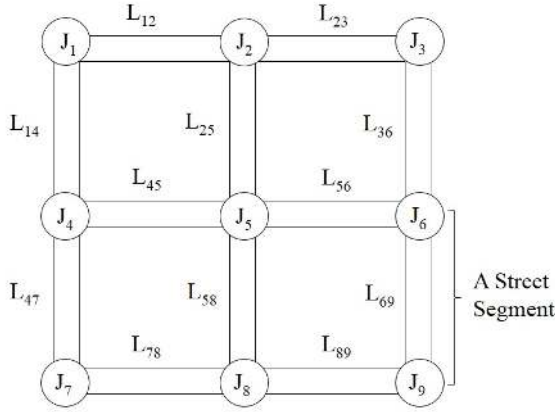


Fig. 1. A section of a city map as an example. J_n : Junction n, L_{ij} : Length of the segment between J_i and J_j

roads using the navigation system. The clocks of all vehicles are synchronized. Synchronization has been considered in IEEE Std 1609.0-2013 [40] and IEEE Std 1609.4-2016 [41] and is necessary for multi-channel operation and security purposes; it can be provided by GPS. Vehicles can be equipped with a sufficient number of computational resources, such as processors and a large memory capacity [42].

B. The Protocol Design

Using a digital map of the streets, each vehicle can obtain an adjacency matrix of the graph that models the city map. As a simple example, Fig. 1 shows a part of a city map with specified junctions in circles (J_n) and street segments with lengths L_{ij} . This map can be represented by a graph, with junctions as vertices and streets as edges. Fig. 2 shows the corresponding adjacency matrix. According to the GSR protocol, whenever a vehicle wants to send a packet to a destination, it initially adds two vertices, which correspond to the source and the destination, to the matrix. Then, it computes the shortest path using Dijkstra's algorithm, adds the ordered list of junctions (anchor points) to the packet header, and then sends it. In EGSR, like GSR, the source vehicle computes an ordered list of the junctions of the route and stores it in the packet header. The list of junctions is computed using Dijkstra's algorithm on a graph representing the city map in which the weight of every edge (street) is proportional to the connectivity of that street segment. In order to make the weight of every edge proportional to the network connectivity of the corresponding street and not just its length, the elements of the matrix in Fig. 2 are redefined as L_{ij}/P_{ij} , where $0 < P_{ij} < 1$. P_{ij} is a variable showing the connectivity condition of the street segment between junctions J_i and J_j . In other words, P_{ij} is the pheromone value related to the street segment between junctions J_i and J_j . A low P_{ij} demonstrates a poor connectivity due to low traffic density. Vehicles update the P_{ij} s according to the information in the ant packets that they receive. This mechanism is described further in the following sections.

Ant packets are launched by the vehicles in junction areas and are forwarded toward the next junction. On arrival, the

	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9
J_1	∞	L_{12}	∞	L_{14}	∞	∞	∞	∞	∞
J_2	L_{12}	∞	L_{23}	∞	L_{25}	∞	∞	∞	∞
J_3	∞	L_{23}	∞	∞	∞	L_{36}	∞	∞	∞
J_4	L_{14}	∞	∞	∞	L_{45}	∞	L_{47}	∞	∞
J_5	∞	L_{25}	∞	L_{45}	∞	L_{56}	∞	L_{58}	∞
J_6	∞	∞	L_{36}	∞	L_{56}	∞	∞	∞	L_{69}
J_7	∞	∞	∞	L_{47}	∞	∞	∞	L_{78}	∞
J_8	∞	∞	∞	∞	L_{58}	∞	L_{78}	∞	L_{89}
J_9	∞	∞	∞	∞	∞	L_{69}	∞	L_{89}	∞

Fig. 2. Adjacency matrix of the map in Fig. 1.

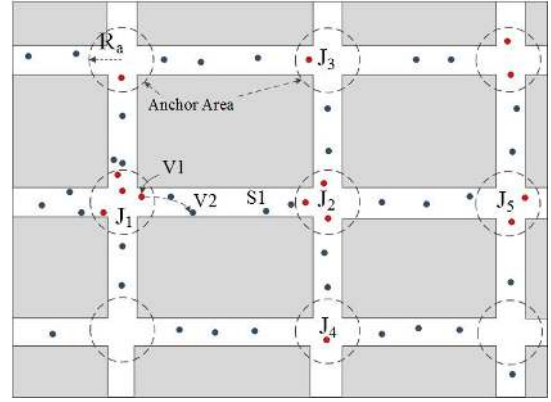


Fig. 3. Anchor areas at every junction.

junction ID is recorded in the ant packet and the next street is selected randomly with a probability proportional to the number of vehicles in the street in the neighborhood of the current ant holder. Then, the ant is forwarded to the junction located at the end of the selected street segment. Between two junctions, ants are broadcast similar to POCA [43] so as to prevent a broadcast storm, but using a simpler approach than POCA. When a node wants to broadcast an ant, it selects its nearest neighbor to the next junction for rebroadcasting it. If an ant packet passes completely through a street segment, there is connectivity in that segment. Every node that receives an ant updates its adjacency matrix: the P_{ij} related to the street segment between the junctions J_i and J_j traversed by the ant will be increased (see section III-D). In other words, the ant deposits pheromone. In this way, vehicles have an adjacency matrix in which the weight of each street is proportional to its length and network connectivity.

A mechanism for pheromone evaporation, that decreases the P_{ij} s in regular intervals so as to make the adjacency matrix adaptive to traffic changes is presented in section III-D. The number of ants that traverse a street and the length of time it takes reflect the packet-relaying condition of the street segment. If there are not enough vehicles in a street or it is congested, fewer ants and over a longer period of time can pass through that street. When this occurs, the evaporation mechanism decreases the P_{ij} related to that street more rapidly than ants can increase it.

C. Launching the Ants

An area with radius R_a at every junction is called an anchor area (see Fig. 3). The time interval between launching successive ants at a junction is called t_{ant} . If a vehicle in an anchor area during time interval t_{ant} , does not receive a new ant (an ant that has been launched in this junction), it creates one and broadcasts it toward the next junction. This vehicle also selects one neighbor as the next ant forwarder. For example, in Fig. 3, vehicle $V1$ located in the anchor area of street $S1$ launches an ant and broadcasts it toward junction J_2 . It selects $V2$, which is its closest neighbor to the next junction (J_2), to rebroadcast the ant. Then, only one vehicle will rebroadcast an ant in every forwarding step. When the ant reaches junction J_2 , J_2 is recorded in the ant packet. Then, the ant is sent to one of the street segments J_2-J_3 , J_2-J_4 , or J_2-J_5 , selected randomly, with a higher probability of selection given to the street that has more vehicles in the neighborhood of the current ant holder. Algorithm 1 presents the ant-launching process.

t_{ant} is the time between launching successive ants. It determines how quickly the algorithm adapts to changes in vehicular and data traffic of the streets. A too-small t_{ant} causes the network to be flooded by ants, and with a large t_{ant} , the protocol cannot adapt to the changes in the network. In both cases, performance of the protocol decreases. Analysis of the parameters of ant-based protocols in ad hoc networks has been presented in [44]. We have selected this parameter experimentally and using the results in [44].

Algorithm 1 Ant-launching process

```
// Vehicle  $V_i$  upon entering anchor area of  $J_i$ 
1: Set timer  $T_a = t_{ant}$ 
2: Upon  $V_i$  receives an ant do
3: if (the only junction recorded in the ant== $J_i$ ) then
4:   Reset timer  $T_a$  (timer  $T_a = t_{ant}$ )
5: end if
6: end do
7: Upon timer  $T_a = 0$  do
8:   Launch ant
9:   Reset beacon timer
10:  Set timer  $T_a = t_{ant}$ 
11: end do
12: Upon  $V_i$  left the anchor area do
13:   Cancel timer  $T_a$ 
14: end do
```

D. Updating the Adjacency Matrix

When a vehicle receives an ant, it updates its adjacency matrix. If junctions J_i and J_j have been recorded in the ant packet as two consecutive junctions, it means there was connectivity to pass the ant through the street segment between junctions J_i and J_j so P_{ij} and P_{ji} will be updated according to

$$P_{ij} = P_{ji} = \frac{P_{ij} + \Delta P_{ij}}{1 + \Delta P_{ij}} \quad (1)$$

where ΔP_{ij} is

$$\Delta P_{ij} = A + \frac{2}{\pi} \arctan\left(\frac{mindelay_{ij}}{delay_{ij}}\right) \quad (2)$$

where

- $delay_{ij}$ is the time it takes the ant to traverse the street between junctions J_i and J_j ,
- $mindelay_{ij}$ is the minimum delay for ants that the vehicle has recorded for that street, and
- A is a constant.

Because $mindelay_{ij}$ is less than or equal to $delay_{ij}$, $(2/\pi) \arctan(mindelay_{ij}/delay_{ij})$ is between 0 and 0.5. Therefore, for A less than 0.5, ΔP_{ij} will be less than 1. The initial value for P_{ij} is selected to be less than 1 and therefore, P_{ij} will always be less than 1. Every time P_{ij} is renewed according to (1), it will be increased so the value L_{ij}/P_{ij} will be decreased. ΔP_{ij} is greater if the delay the ant encounters in a street is lower. As a result, the weight of that street would be decreased with decreasing delay. Similarly, if three junctions J_i , J_j , and J_k have been recorded in the ant packet as three consecutive junctions, P_{ij} , P_{ji} , P_{jk} , and P_{kj} will each be updated according to (1) and (2). If the vehicle that has received the ant is the next forwarder, it then selects its closest neighbor to the next junction as the next forwarder and rebroadcasts the ant. If it is not the next forwarder, it just renews its adjacency matrix according to the ant's information.

At constant intervals (i.e. t_{eva} seconds), each vehicle decreases the pheromone (P_{ij}) of all the streets, using the following formula:

$$P_{ij} = \max(0.1, \alpha \cdot P_{ij}), \quad \text{where } 0 < \alpha < 1 \quad (3)$$

This process, which is called pheromone evaporation, causes reduction of pheromone of street segments that are not good path anymore.

With the proposed mechanisms, pheromone increase and decrease, every vehicle regularly updates the weights of the edges of the graph representing the map of the surrounding area proportional to the connectivity of the streets. Thus, the route that every source vehicle computes for its data packets is adaptive to the traffic conditions on the streets.

Every node broadcasts its ID and position in beacons regularly. When a node broadcasts an ant, it also includes beacon information in the ant packet and resets its beacon timer so fewer beacons are required. The purpose of this is to reduce the congestion in the network; because sending two packets will contribute to congestion more than sending one packet with the sum of their sizes [45].

E. Ant Packets

An ant packet consists of the following fields:

- **Type:** Indicates the type of the packet.
- **Sender_Id:** ID (or address) of the first node that issued the ant.
- **Serial_Number:** Every node assigns numbers sequentially to the ants it creates.
- **Version:** First node that creates an ant sets this field to zero. Every time a node adds a junction ID to Sequence_Of_Junctions field, it increases this field by one.

- **Street_Id:** ID of the street that the ant is traversing.
- **Next_Junction_Position:** Position of the next junction that the ant should be sent toward.
- **Sequence_Of_Junctions:** Sequence of junctions that the ant has traversed.
- **S_Delays:** Time stamps showing the times that the ant has passed each junction.
- **Next_Forwarder:** ID (address) of the next node that should forward (broadcast) the ant.
- **LastSender_Id:** ID of the last node that has broadcast the ant.
- **LastSender_Position:** Position of the last node that has broadcast the ant.

Sender_Id and Serial_Number have the same functionality as in any other regular routing protocol, that is, to ignore repeated packets. Every node that receives an ant checks the Serial_Number, Sender_Id, and Version fields of the ant. If it has not already received one with the Serial_Number and Sender_Id, it uses the ant's information to update its adjacency matrix. If it has received the ant with the same Serial_Number and Sender_Id, but a lower Version, it just uses that part of the information of the packet that it has not already received, to update its adjacency matrix. For example, if a node has received an ant with version number 2, it means that the ant has passed three junctions. If it then receives an ant with the same Serial_Number and Sender_Id and version number 3, it just updates the P_{ij} related to the street between the last two junctions.

If the node that receives the ant is the next forwarder, it selects the subsequent next forwarder and then broadcasts the ant, whether or not it has received it before. For example, in Fig. 4, vehicle $V1$ creates and sends an ant toward junction J_2 . $V1$ sets the Version field to zero, Sequence_Of_Junctions to J_1 , S_Delays to current time, Next_Forwarder to $V2$, Next_Junction_Position to the coordinates of J_2 , Street_Id to $S1$, and LastSender_Id and LastSender_Position to its ID and position, subsequently broadcasting the ant. All nodes $V11$, $V12$, $V13$, and $V2$ receive the ant and update their adjacency matrices if required, which is not necessary in this situation because the ant has only one junction in its Sequence_Of_Junctions field. The ant is then broadcast by $V2$ and $V3$ in turn. Assuming that $V3$ has selected $V4$ as the next forwarder, $V4$ checks the Street_Id of the ant and selects one of its neighbors that is not in the street $S1$, for example $V5$. Subsequently, $V4$ changes the Street_Id field to $S2$, Next_Junction_Position to coordinates of J_3 , Next_Forwarder to $V5$, and LastSender_Id and LastSender_Position to its ID and position, respectively. It also records J_2 to Sequence_Of_Junctions, current time to S_Delays, and increases the Version number by one. Consequently, vehicles receiving this packet, can compute the time this ant took to travel from J_1 to J_2 .

The requirement of the Version field can be explained as follows. When vehicle $V3$ broadcasts the ant, both $V5$ and $V6$ are within its communication range and receive it. When $V4$ updates the ant and broadcasts it, if the Version field does not exist, $V5$ and $V6$ ignore the new information that has been added by $V3$. In addition, by comparing the new and

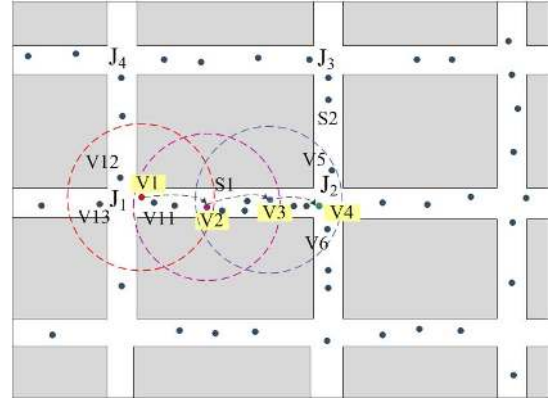


Fig. 4. Ant launching and forwarding by vehicles. Dashed circles show communication ranges of vehicles $V1$, $V2$, and $V3$.

Algorithm 2 Forwarding ants and updating pheromones process

```

// Vehicle  $V_i$  received ant  $A_i$  with Street_Id =  $S_n$ 
// Street segment  $S_m$  is between junctions  $J_i$  and  $J_j$ 
1: if ( $A_i$  not received before) then
2:   if ( $V_i == Next\_Forwarder$ ) then
3:     if ( $V_i$  is in the anchor area of  $J_i$ ) then
4:       if ( $J_i$  has not been recorded in  $A_i$ ) then
5:         Record  $J_i$  in Sequence_Of_Junctions
6:         if (size of Sequence_Of_Junctions < max
           size of Sequence_Of_Junctions) then
7:           Record current time in S_Delay
8:           Select next forwarder  $V_j$  in  $S_m$  ( $n \neq m$ )
9:            $Version \leftarrow Version + 1$ 
10:           $Street\_Id \leftarrow S_m$ 
11:           $Next\_Junction\_Position \leftarrow position(J_j)$ 
12:        end if
13:      end if
14:    else
15:      Select the next forwarder  $V_j$  using greedy mechanism
16:    end if
17:    Update pheromones using (1) and (2)
18:     $LastSender\_Id \leftarrow V_i$ 
19:     $LastSender\_Position \leftarrow position(V_i)$ 
20:     $Next\_Forwarder \leftarrow V_j$ 
21:    Transmit  $A_i$ 
22:    Reset beacon timer
23:  else
24:    Update pheromones using (1) and (2)
25:  end if
26: end if
27: update neighbor table

```

previous versions of the ant, they use just that part of the information they have not used before (newly added junctions). In the protocol, there is a limit on the number of junctions ants can travel and after that limit, the nodes kill them. _Id and LastSender_Position are required because the ant can have the same functionality as a beacon, with the benefit being that fewer beacons are required. Algorithm 2 presents the ant

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Scenario Area	2000 m × 2000 m
Communication Range	300 m
MAC Protocol	IEEE 802.11p
Simulation Time	800 seconds (s)
Vehicle Velocity	45, 50, 55, 60, 65, 70 km/h
Carrier Frequency	5.89 GHz
Number of Concurrent Connections	10
Bit Rate	18 Mbps
Beacon Frequency	2 Hz
Data Packet Size	512 Bytes unless specified
EGSR Parameters	$t_{ant} = 1.5$ s, $t_{evt} = 5$ s $\alpha = 0.92$

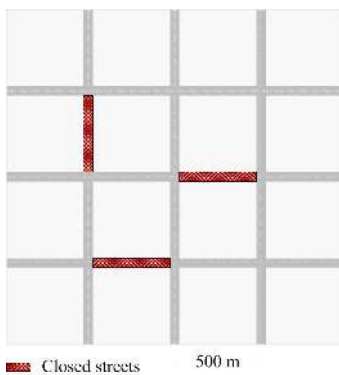


Fig. 5. Simulation map.

forwarding and the pheromone updating process.

IV. PERFORMANCE RESULTS

The performance of EGSR has been compared with GSR and VACO. OMNeT++ [46] and SUMO [47] have been used to simulate the network and generate vehicular traffic mobility. OMNeT++ has a framework called Veins (vehicles in network simulation), that specifically designed for simulation of VANETs. Veins has standard models for physical and MAC layers of VANET devices and couples OMNeT++ and SUMO simulators together. It also supports various radio propagation models from simple free space, to more complicated models designed for vehicle to vehicle communication [48]. For the mobility model and urban map topology, the Manhattan model has been employed. The propagation model is log-normal shadowing. The simulation area covers a 2000 × 2000 meter grid in which the distance between two adjacent junctions is set to 500 meters. The streets are two-way, with two lanes in each direction. Three of them have no traffic, as shown in Fig. 5. The source and destination of data packets are selected randomly. Every experiment has been repeated 20 times and the similar conditions have been considered with regards to performance analysis of the three protocols. Simulation time is 800 s and the data have been collected over 700 s. All the vehicles have capability of wireless communications. The size of beacon and ant packets, respectively, are considered to be

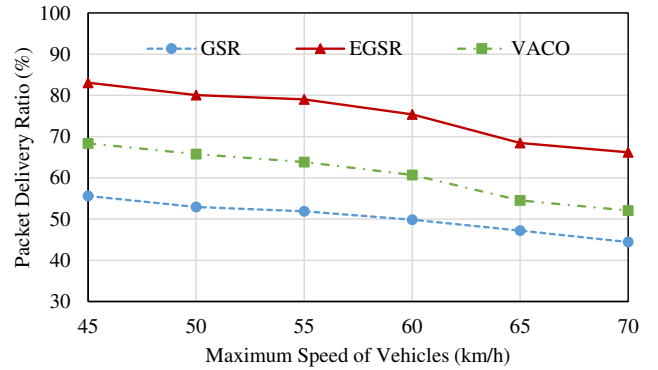


Fig. 6. Data packet delivery ratio for different vehicle speeds.

300 and 200 bytes. Thus, in EGSR that includes the beacon's information in the ant packets, beacons are 300 bytes and ants are 500 bytes. The simulation parameters are summarized in Table I.

A. Packet Delivery Ratio of Protocols

Fig. 6 shows the packet delivery ratio of the protocols for a packet rate of 5 Pkt/s for different vehicle speeds. EGSR performs better than the other protocols by at least 10% up to a speed of 70 km/h. VACO needs more control packets for route set up and maintenance, because the ants deposit pheromone on the backward journey, while in EGSR the forward ants deposit pheromone, so a backward journey is not required. Generally, longer journeys for ants increase both overhead and packet loss due to collision. In EGSR, the control packets take a probabilistic path and the data packets have a deterministic path because the source node determines the junction IDs of the path for the data packets. In VACO, both kinds of packets have a probabilistic path, which might result in sub-optimal choices for data packets. While the pheromones deposited by the ants increase the probability of selecting the optimal path for data packets, there is a possibility a sub-optimal path will be selected. These are reasons for the lower delivery ratio of VACO compared to EGSR. It should be noted that GSR, EGSR, and VACO use the same greedy mechanism to relay packets but work differently when selecting junctions. GSR selects the shortest path without considering the packet relaying quality of the path, which leads to the lowest packet delivery ratio among them.

The reason for the increase in dropped packets at higher speeds is that the position of vehicles changes more rapidly. By using the greedy mechanism to select the next hop, the node that is closest to the destination is selected. Such nodes are usually close to the border of the communication range and can leave it in a shorter time when the speed is higher. Fig. 7 shows packet delivery ratios for different packet rates. Fig. 8 shows the packet delivery ratios for different packet sizes from 256 bytes to 2304 bytes. Larger packets are more susceptible to loss due to higher probability of collision and fading. By increasing the packet size, the packet delivery ratio of EGSR drops less than that of VACO because it uses fewer control packets and thus the protocol suffers fewer collisions.

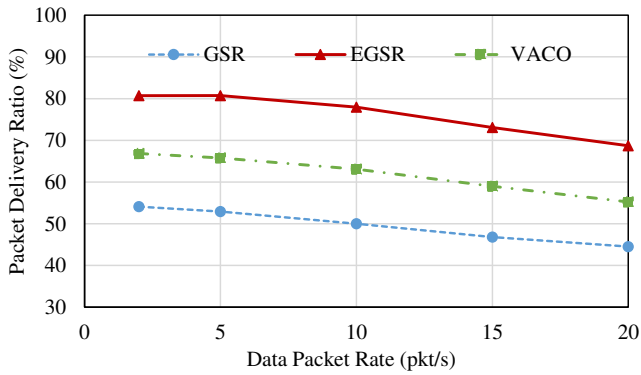


Fig. 7. Data packet delivery ratio for different data rates; maximum vehicle speed of 50 km/h.

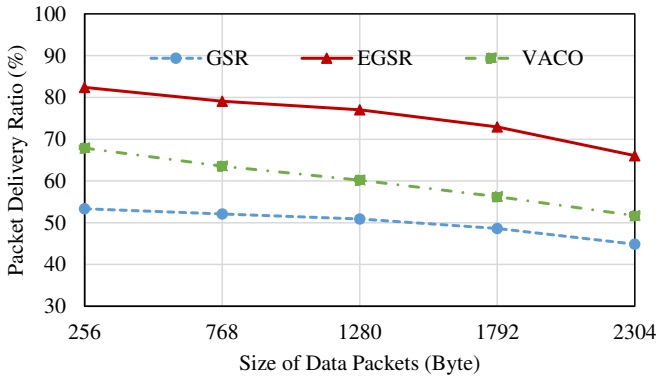


Fig. 8. Data packet delivery ratio with different packet sizes; maximum vehicle speed of 50 km/h.

As it was explained in Section III-E, ants can travel up to a number of junctions; i.e. in our proposed protocol up to seven junctions. Therefore, vehicles obtain connectivity information for the surrounding streets and can compute the most connected path up to seven junctions away. For longer paths, the last vehicle using the same mechanism computes the remaining path up to the next seven junctions and thus, the number of reachable junctions is extended. This approach recursively provides connectivity to any reachable destinations. In relations to the overall data packet delivery ratio for longer distances, we can say that packet loss ratio is multiplicative, i.e.

$$\text{End-to-end data packet loss} = \left(1 - \prod_{i=1}^n (1 - p_i) \right) \quad (4)$$

where p_i is packet loss ratio (probability) measured for street segment i and n is the number of street segments along the route. It is generally assumed that the losses are independent and are not correlated with respect to end-to-end packet loss calculation [49].

B. Routing Protocols Overhead

In this section, the overhead of routing protocols, i.e. beacons and ants, have been compared in GSR, EGSR, and VACO. Fig. 9 demonstrates that in EGSR, at different speeds overhead is almost constant. This is due to the fact that EGSR

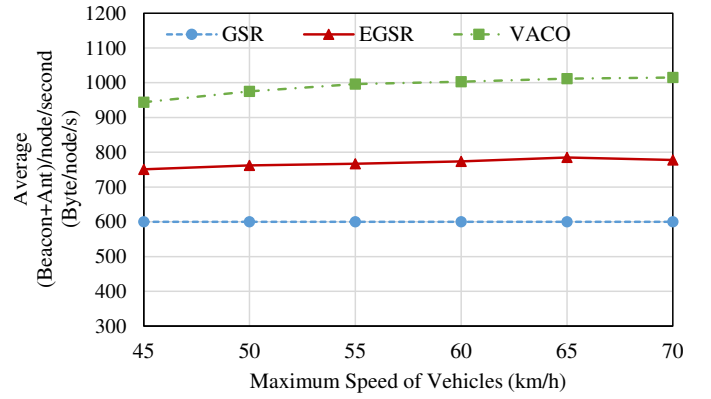


Fig. 9. Control packet rate for different vehicle speeds.

is road-based, not node-based, and it evaluates connectivity between junctions rather than between nodes. Even when there are many vehicles in an anchor area, which might occur at junctions, the network is not flooded with ants, because the vehicles cooperate on launching ants. A vehicle issues an ant if it does not receive a new one in a specified time (t_{ant}). VACO is a road-based protocol as well, but it has increasing overhead with increasing velocity. In VACO a communication session is established between source and destination vehicles through RSUs. The source vehicle sends its packet to the first RSU, and the packet is delivered to the destination vehicles by the last RSU. Because of the movement of source and destination vehicles, the first and the last RSU changes during communication. This requires a new reactive route set up. At higher velocities, this happens more frequently, which contributes to more overhead.

Fig. 10 shows routing control overhead created by the protocols during the simulation time for different numbers of flows (source-destination pairs). VACO creates both reactive and proactive ants and they should travel a route two times (forward and backward) to deposit pheromone. Every flow of data requires route set up and maintenance thus the overhead increases by increasing the number of flows. Fig. 11 shows the total number of routing control packets created by the protocols during the simulation time for different numbers of vehicles, while the data rate is 5 Pkt/s. As every vehicle creates beacons at 2 Hz frequency, by increasing the vehicle density the number of control packets grows for all the protocols. The figure shows that the excess overhead in EGSR with respect to GSR due to ants does not grow with increasing vehicle density, so it is not sensitive to the number of vehicles. Therefore, EGSR is scalable and can work well for different vehicle densities. It is worth noting that overhead due to beaconing can be alleviated by using adaptive beaconing approaches [50], [51].

C. End-to-End Packet Delay

Fig. 12 shows the average end-to-end delay against Euclidean distance between source and destination for data packets. GSR selects road segments to forward packets based only on physical shortest paths between source and destination.

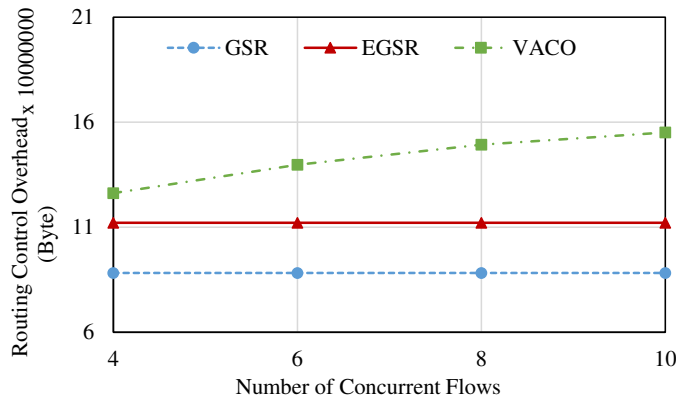


Fig. 10. Overhead for different numbers of flows.

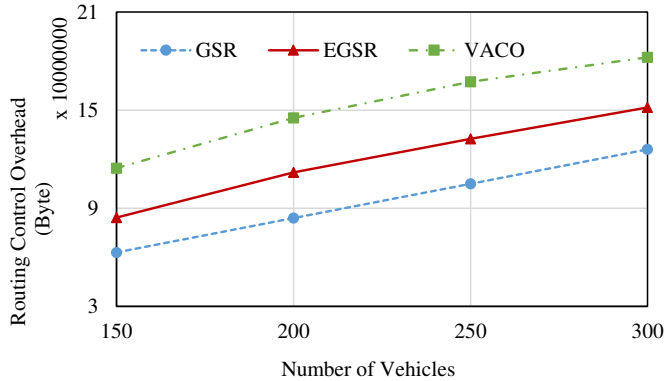


Fig. 11. Overhead for different numbers of vehicles; maximum vehicle speed is 50 km/h and data packet rate is 5 Pkt/s.

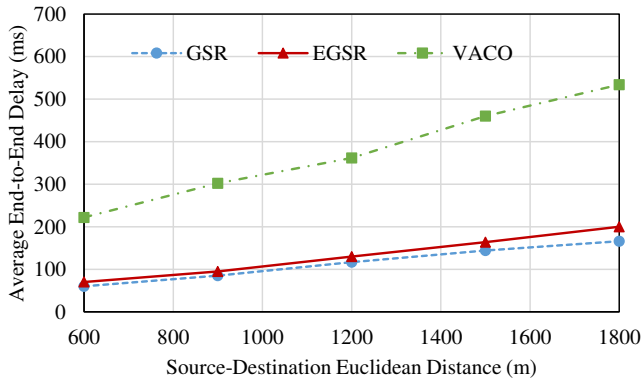


Fig. 12. Average end-to-end delay for data packets.

Therefore, a road segment might be congested and the packets face delay. VACO and EGSR consider delay to select road segments. In VACO, due to the required time to set up a route by reactive forward and backward ants, end-to-end delay is much higher than in EGSR and GSR. The higher delay of EGSR compared to GSR is due to successful delivery of packets with EGSR for which the shortest path between their source and destination does not have connectivity. Therefore these packets are dropped by GSR. While EGSR can find a connected path for delivering them through a longer path, leading to a higher average end-to-end delay.

In situations where the next forwarder vehicle equipped with VANETs device is not found (is quite common in early deployment of VANETs), the conventional carry-and-forward approach should be used; i.e. a car carries a packet until it comes across another VANET enabled vehicle that can take and pass the packet to other vehicles. Using this approach will certainly increase end-to-end delay and packet drop.

V. CONCLUSIONS

A Traffic-aware routing protocol based on ACO called EGSR was proposed. In the EGSR protocol, using small control packets, called ants, vehicles regularly evaluate the connectivity of the streets in their vicinity. Ants are broadcast by an efficient broadcasting mechanism to control broadcast storm. The protocol is road-based, traffic-aware, and not sensitive to the movement of nodes. Moreover, it does not need additional hardware like traffic sensors or RSUs at every junction. By defining an area around every junction, called an anchor area, vehicles in this area cooperate to launch ants. Simulation results showed that the proposed routing protocol (EGSR) gives better performance than GSR and VACO. That is its packet delivery ratio is at least 10% higher than that of the other compared protocols up to a speed of 70 km/h. In addition routing control overhead of EGSR is less than VACO. In summary, we showed that the presented EGSR provides a good performance and contributes towards an operationally optimized routing protocol in VANETs.

REFERENCES

- [1] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surveys Tut.*, vol. 13, no. 4, pp. 584–616, 2011.
- [2] A. M. Vegni, M. Biagi, and R. Cusani, "Smart vehicles, technologies and main applications in vehicular ad hoc networks," in *Vehicular Technologies-Deployment and Applications*. InTech, 2013.
- [3] P. Papadimitratos, A. De La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Commun. Mag.*, vol. 47, no. 11, 2009.
- [4] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC 3561, Jul. 2003.
- [5] P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Proc. IEEE INMIC*, 2001, pp. 62–68.
- [6] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Springer, 1996, pp. 153–181.
- [7] 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 Intell. Veh. Symp.*, 2003, pp. 156–161.
- [8] A. Husain, R. S. Raw, B. Kumar, and A. Doegar, "Performance comparison of topology and position based routing protocols in vehicular network environment," *Int. J. Wireless & Mobile Netw. (IJWMN)*, vol. 3, no. 4, pp. 289–303, Aug. 2011.
- [9] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. and Netw.*, 2000, pp. 243–254.
- [10] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE Mobile Comput. and Commun. Rev.*, vol. 9, no. 1, pp. 69–72, 2005.
- [11] 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 *Networking 2004*. Springer, 2004, pp. 989–999.
- [12] F. Giudici and E. Pagani, "Spatial and traffic-aware routing (STAR) for vehicular systems," in *High Performance Computing and Communications*. Springer, 2005, pp. 77–86.

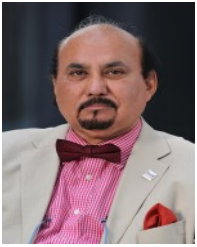
- [13] M. Jerbi, S.-M. Senouci, R. Meraihi, and Y. Ghamri-Doudane, "An improved vehicular ad hoc routing protocol for city environments," in *IEEE Int. Conf. Commun.*, 2007, pp. 3972–3979.
- [14] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea, "VANET routing on city roads using real-time vehicular traffic information," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3609–3626, Sep. 2009.
- [15] G. Li and L. Boukhatem, "Adaptive vehicular routing protocol based on ant colony optimization," in *Proc. 10th ACM Int. Workshop Veh. Inter-Netw., Syst. and Appl.* ACM, 2013, pp. 95–98.
- [16] P. K. Sahu, E. H.-K. Wu, J. Sahoo, and M. Gerla, "BAHG: back-bone-assisted hop greedy routing for VANET's city environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 199–213, Mar. 2013.
- [17] H. Saleet, R. Langar, K. Naik, R. Boutaba, A. Nayak, and N. Goel, "Intersection-based geographical routing protocol for VANETs: a proposal and analysis," *IEEE Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4560–4574, Nov. 2011.
- [18] V. Naumov and T. R. Gross, "Connectivity-aware routing (CAR) in vehicular ad-hoc networks," in *26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 1919–1927.
- [19] F. Dressler and O. B. Akan, "Bio-inspired networking: from theory to practice," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 176–183, Nov. 2010.
- [20] M. Wang and T. Suda, "The bio-networking architecture: A biologically inspired approach to the design of scalable, adaptive, and survivable/available network applications," in *Appl. the Internet, 2001. Proc. 2001 Symp. on.* IEEE, 2001, pp. 43–53.
- [21] C. Prehofer and C. Bettstetter, "Self-organization in communication networks: principles and design paradigms," *IEEE Commun. Mag.*, vol. 43, no. 7, pp. 78–85, Jul. 2005.
- [22] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Comput. Netw.*, vol. 54, no. 6, pp. 881–900, Jan. 2010.
- [23] S. Bitam, A. Mellouk, and S. Zeadally, "Bio-inspired routing algorithms survey for vehicular ad-hoc networks," *IEEE Commun. Surveys Tut.*, vol. 17, no. 2, pp. 843–867, May 2015.
- [24] M. Farooq and G. A. Di Caro, "Routing protocols for next-generation networks inspired by collective behaviors of insect societies: An overview," in *Swarm Intelligence*. Springer, 2008, pp. 101–160.
- [25] R. Schoonderwoerd, O. E. Holland, J. L. Bruten, and L. J. Rothkrantz, "Ant-based load balancing in telecommunications networks," *Adaptive Behavior*, vol. 5, no. 2, pp. 169–207, 1997.
- [26] G. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," *J. Artificial Intell. Res.*, vol. 9, pp. 317–365, Dec. 1998.
- [27] G. Di Caro, F. Ducatelle, L. M. Gambardella, and M. Dorigo, "AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," *Eur. Trans. Telecommun.*, vol. 16, no. 5, pp. 443–455, Sep. 2005.
- [28] S. Kamali and J. Opatrny, "A position based ant colony routing algorithm for mobile ad-hoc networks," *J. Netw.*, vol. 3, no. 4, pp. 31–41, Apr. 2008.
- [29] S. L. Correia, J. Celestino, and O. Cherkaoui, "Mobility-aware ant colony optimization routing for vehicular ad hoc networks," in *2011 IEEE Wireless Commun. Netw. Conf.* IEEE, 2011, pp. 1125–1130.
- [30] H. Rana, P. Thulasiraman, and R. K. Thulasiram, "Mazacornet: Mobility aware zone based ant colony optimization routing for VANET," in *2013 IEEE Congr. Evol. Computation.* IEEE, 2013, pp. 2948–2955.
- [31] R. R. Sahoo, R. Panda, D. K. Behera, and M. K. Naskar, "A trust based clustering with ant colony routing in VANET," in *Comput. Commun. & Netw. Technol. (ICCCNT), 2012 3rd Int. Conf.* IEEE, 2012, pp. 1–8.
- [32] G. Li, L. Boukhatem, and J. Wu, "Adaptive quality-of-service-based routing for vehicular ad hoc networks with ant colony optimization," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3249–3264, 2017.
- [33] G. Li and L. Boukhatem, "A delay-sensitive vehicular routing protocol using ant colony optimization," in *Ad Hoc Networking Workshop (MED-HOC-NET), 2013 12th Annual Mediterranean.* IEEE, 2013, pp. 49–54.
- [34] J. Toutouh, J. García-Nieto, and E. Alba, "Intelligent OLSR routing protocol optimization for VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 1884–1894, May 2012.
- [35] C. Wu, S. Ohzahata, and T. Kato, "Flexible, portable, and practicable solution for routing in vanets: a fuzzy constraint Q-learning approach," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, pp. 4251–4263, Nov. 2013.
- [36] J. F. Bravo-Torres, M. Lopez-Nores, Y. Blanco-Fernandez, J. J. Pazos-Arias, M. Ramos-Cabrer, and A. Gil-Solla, "Optimising reactive routing over virtual nodes in VANETs," *IEEE Trans. Veh. Technol.*, vol. PP, no. 99, pp. 1–22, Jun. 2015.
- [37] M. Al-Rabayah and R. Malaney, "A new scalable hybrid routing protocol for VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2625–2635, Jul. 2012.
- [38] T. Darwish and K. A. Bakar, "Traffic density estimation in vehicular ad hoc networks: A review," *Ad Hoc Netw.*, vol. 24, pp. 337–351, Jan. 2015.
- [39] I. Chakeres and C. Perkins, "Dynamic MANET on-Demand (DYMO)," *draft-ietf-manet-dymo-26*, 2013.
- [40] "IEEE Guide for Wireless Access in Vehicular Environments (WAVE)-Architecture," *IEEE Std. 1609.0-2013*, 2013.
- [41] "IEEE Standard for Wireless Access in Vehicular Environments (WAVE)- Multi-channel Operation," *IEEE Std. 1609.4-2016*, 2016.
- [42] 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, 2014.
- [43] K. N. Nakom and K. Rojviboonchai, "POCA: position-aware reliable broadcasting in VANET," in *Proc. Asia-Pacific Conf. Inform. Process.*, 2010, pp. 420–428.
- [44] F. Ducatelle, G. A. Di Caro, and L. M. Gambardella, "An analysis of the different components of the AntHocNet routing algorithm," in *Int. Workshop Ant Colony Optimization and Swarm Intell.* Springer, 2006, pp. 37–48.
- [45] S. Yousefi, S. Bastani, and M. Fathy, "On the performance of safety message dissemination in vehicular ad hoc networks," in *4th Eur. Conf. Universal Multiservice Netw.*, 2007, pp. 377–390.
- [46] OMNeT++ Discrete Event Simulator. (Accessed: 09/05/2016). [Online]. Available: <http://www.omnetpp.org/>.
- [47] Simulation of Urban MObility (SUMO). (Accessed: 09/05/2016). [Online]. Available: <http://sumo.sourceforge.net/>.
- [48] F. Goudarzi, "Non-cooperative beaconing control in vehicular ad hoc networks," Ph.D. dissertation, Brunel University London Department of Electronic and Computer Engineering, Feb. 2017.
- [49] A. Asgari, R. Egan, P. Trimintzios, and G. Pavlou, "Scalable monitoring support for resource management and service assurance," *IEEE Net.*, vol. 18, no. 6, pp. 6–18, 2004.
- [50] F. Goudarzi and H. Asgari, "Non-cooperative beacon rate and awareness control for VANETs," *IEEE Access*, vol. 5, Aug. 2017.
- [51] F. Goudarzi and H. Al-Rawashidy, "Distributed transmit power control for beacons in VANET," in *Veh. Technol. Intell. Transp. Syst. (VEHITS 2017), 3rd Int. Conf. on.* Scitepress, 2017, pp. 181–187.



Forough Goudarzi received the M.Sc. and PhD degree in Electrical Engineering from Sharif University of Technology, Tehran, Iran and Brunel University London, UK respectively. She is currently Senior Research Assistant in the Transport Research Group, University of Southampton, UK. Her research interests include VANETs, bio-inspired networks and applications of game theory in wireless networks.



Hamid Asgari (SM'03) received the Ph.D. degree in Electrical and Electronics Engineering from the University of Wales, Swansea, UK in 1997. He has been with Thales UK Research, Technology and Innovation since 1996 and is now a Chief Technical Consultant there. He is also a Visiting Professor at King's College London. Professor Asgari is a highly experienced and skilled professional in leading large R&D teams, a technical expert in data communication networks, and security subjects. He has been leading R&D teams and participating in collaborative European Commission (EC) projects since the year 2000. He is the Network Technical Lead working in Advanced and Future Networking and Security Concepts including Information and Cyber Security, Multi-Level Security, Wired/Wireless Networks Architectures and Technologies, Network, Service and Quality of Service (QoS) Management. He has a proven track record and published 60 book chapters and papers in the most respected scientific journals and peer-reviewed conferences. He is an IET Fellow and Senior member of both the IEEE and ACM.



Hamed S. Al-Raweshidy (SM'03) received the B.Eng. and M.Sc. degrees from the University of Technology, Baghdad, Iraq in 1977 and 1980, respectively, the Post Graduate Diploma degree from Glasgow University, Glasgow, U.K. in 1987, and the Ph.D. degree from Strathclyde University, Glasgow in 1991. He has worked with the Space and Astronomy Research Center, Baghdad, Perkin Elmer, Waltham, British Telecom, Oxford University, Manchester Metropolitan University, and Kent University, Canterbury, U.K. Currently, he is the Director of Wireless Network and Communications Center at Brunel University, London, U.K.