Situation-Aware QoS Routing Algorithm for Vehicular Ad Hoc Networks

Mahmoud Hashem Eiza, *Member, IEEE*, Thomas Owens, *Member, IEEE*, Qiang Ni, *Senior Member, IEEE*, and Qi Shi

Abstract—A wide range of services has been developed for vehicular ad hoc networks (VANETs), ranging from safety to infotainment applications. An essential requirement for such services is that they are offered with quality of service (QoS) guarantees in terms of service reliability and availability. Searching for feasible routes subject to multiple QoS constraints is, in general, an NP-hard problem. Moreover, routing reliability needs to be paid special attention as communication links frequently break in VANETs. In this paper, we propose employing the situational awareness (SA) concept and an ant colony system (ACS)-based algorithm to develop a situation-aware multiconstrained QoS (SAMQ) routing algorithm for VANETs. SAMQ aims to compute feasible routes between the communicating vehicles subject to multiple QoS constraints and pick the best computed route, if such a route exists. To mitigate the risks inherited from selecting the best computed route that may turn out to fail at any moment, SAMQ utilizes the SA levels and ACS mechanisms to prepare certain countermeasures with the aim of assuring a reliable data transmission. Simulation results demonstrate that SAMQ is capable of achieving a reliable data transmission, as compared with the existing QoS routing algorithms, even when the network topology is highly dynamic.

Index Terms—Ant colony system (ACS), quality of service (QoS) routing, situational awareness (SA), vehicular ad hoc networks (VANETs), vehicular networks.

I. INTRODUCTION

RECENTLY, the development of vehicular ad hoc networks (VANETs) has received more attention and research efforts from the industry and academic community [1]–[3]. VANETs are a key technology in intelligent transportation systems that are envisaged to play a significant role in the futuristic smart cities by improving road safety and providing innovative services relating to traffic management and infotainment applications. This has stimulated the creation of a wide range of services for future deployment in VANETs, ranging from safety and traffic management to commercial applications [4]. These services must be offered with quality of service (QoS) guarantees. Otherwise, they will not be successfully deployed

or used. The highly dynamic nature of VANETs makes the resource reservation for the services not applicable to providing QoS guarantees. Moreover, the communication link between two vehicles is vulnerable to disconnection due to the vehicular movements and the unpredictable behavior of drivers. These make the QoS metrics associated with the current established routes change rapidly, and the selected best computed route could quickly become inefficient or even infeasible. Therefore, picking the best computed route cannot guarantee reliable data transmission, even if it satisfies the defined QoS constraints when it is established. This problem is the subject of this paper.

Searching for feasible routes in a multihop vehicular network subject to multiple QoS constraints features a multiconstrained path (MCP) selection, which is proven to be an NP-hard problem [5] if the constraints are mutually independent [6]. Considerable work has been conducted to address QoS routing and MCP problems in stable networks (see [7]–[13]). Generally, there are two distinct approaches adopted to solve MCP problems, namely, exact QoS routing algorithms and heuristic and approximation routing algorithms. In these two approaches, different strategies have been followed, such as the nonlinear definition of the path length [14], lookahead feature [15], nondominated paths [16], Dijkstra-like path search [17], and k-shortest path [18]. Distributed heuristic solutions such as swarm-intelligence-based algorithms display several features that make them particularly suitable for solving MCP problems in VANETs. They are fully distributed; hence, there is no single point of failure; they are self-organizing and, thus, robust and fault tolerant; and they intrinsically adapt to traffic changes without requiring complex mechanisms [19]. The ant colony system (ACS) has been recognized as an effective technique for producing results for MCP problems that are very close to those of the best performing algorithms [20]. However, how and, in particular, to what degree the ACS mechanisms can contribute toward a situation-aware multiconstrained QoS (SAMQ) routing in VANETs is still unresolved.

In general, situational awareness (SA) is the ability to extract information from the environment, integrate that information with relevant internal knowledge, and use the resulting mental picture to anticipate future events [21]. It describes the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, the projection of their status in the near future, and the possible countermeasures that can be taken to manage the risks associated with decisions made based on the projection [22], [23]. In this context, a multiconstrained QoS routing process in VANETs can be considered from an SA perspective.

SA provides a different approach to the QoS routing process in VANETs. It considers QoS routing as a continuous process where the routing algorithm selects the best route that satisfies the QoS constraints; prepares certain countermeasures to be taken when the selected best route turns out to fail, i.e., enables a link or route recovery at or near the fail point; and continues to evaluate the current solutions based on the state of the vehicular network. To do that, vehicles' kinematic information, the mathematical distribution of their movements and velocities, and the current vehicular network conditions need to be perceived, comprehended, and analyzed. Based on this analysis, routing decisions are made to ensure that the best QoS route is utilized. Moreover, intermediate nodes are continuously engaged in the preparation of alternative QoS links/routes for immediate use should the current route fail. The information needed to apply this approach successfully in VANETs is already available through different communication channels, e.g., broadcast safety messages. This way, this information facilitates the monitoring activities of the network state conducted by the routing algorithm without being part of the routing control overhead. However, it contributes to the complexity of the routing process when updating the information within the nodes to convey the changes in the network conditions. The computational and communication complexity levels of this approach are further discussed in Section IV.

In this paper, we propose a SAMQ routing algorithm that addresses the QoS routing process in VANETs from an SA perspective. Focusing on the fundamental problem of guaranteeing a reliable data transmission in VANETs, the major contributions of this paper are as follows.

- The SA concept is extended to propose a novel SA model for multiconstrained QoS routing in VANETs and define the function of each SA level.
- An SAMQ routing algorithm is developed to apply the proposed SA model and ACS mechanisms to compute feasible routes subject to multiple QoS constraints among the communicating vehicles and select the best computed route between the source and the destination. The novelty of the SAMQ routing algorithm lies in its unique design that utilizes the SA levels and ACS mechanisms at intermediate nodes to respond locally and immediately, if possible, to any link breakage without interrupting the data transmission. This response not only replaces the broken link but considers the QoS requirements of the data being transmitted as well.
- A single-hop broadcasting technique is utilized at each node to minimize the routing control overhead in the SAMQ routing algorithm. In addition, SAMQ maintains the selected route as long as it is still feasible, i.e., satisfies the QoS constraints, and deals with link breakages locally to decrease the rate of transmitted routing error messages. This way, the routing control overhead is significantly reduced.

To further illustrate the effectiveness of the proposed SAMQ routing algorithm, we conduct extensive simulations in a highway scenario where vehicles move at varying velocities and are allowed to stop, turn, and leave the highway at intersections, as

in a real-world situation. Without loss of generality, highway scenarios are considered since their topologies are normally highly dynamic and, hence, challenging to be tackled. The SAMQ routing algorithm is targeted at the class of applications having only a single destination, i.e., unicast routing, in an infrastructureless VANET. Traffic-related inquiries, traffic accidents, video transmission between police and/or ambulance vehicles, and general information services such as Web surfing and e-mail through mobile Internet gateways are examples of such applications that can be operated in an ad hoc mode when the infrastructure is not available. Simulation results demonstrate the significant performance gains of the SAMQ routing algorithm in achieving a reliable data transmission, even when the network topology is highly dynamic.

We assume that the 5.9-GHz dedicated short-range communication (DSRC) standard is deployed. DSRC is a wireless technology that is designed to support collision prevention applications where each vehicle broadcasts a routine traffic message called a basic safety message (BSM) every 100 ms [24]. BSMs convey a vehicle's state information such as its location, velocity, heading, and acceleration. These BSMs are utilized in the SAMQ routing process but do not contribute to the routing control overhead since they are not part of the routing control messages. A dual-radio device can be introduced with a dedicated transceiver for the exchange of BSMs to avoid channel switching [25]. This way, the bandwidth available for unicast data transmission will not be affected by the periodic exchange of BSMs.

The rest of this paper is organized as follows: Section II overviews the state of the art in this field. Section III proposes the SA model for multiconstrained QoS routing in VANETs. Section IV proposes the SAMQ routing algorithm and discusses its properties in detail. Section V presents the implementation of the SAMQ-based routing protocol. Section VI evaluates the performance of the SAMQ routing algorithm and discusses the results obtained. Finally, Section VII concludes this paper.

II. STATE OF THE ART

Apart from military applications where SA is used in vehicle convoys [26], to the best of our knowledge, there are no previous studies on utilizing SA in multiconstrained QoS routing in VANETs. However, several studies have been conducted to address multipath routing [27]–[32], [34] and QoS routing using swarm intelligence mechanisms [36]–[44] in VANETs to achieve a reliable data transmission. Next, we give a brief review of some related work.

In the context of multipath routing, there are two main mechanisms for computing multipath routes: node-disjoint and link-disjoint. In the node-disjoint mechanism, e.g., ad hoc on-demand distance vector multipath (AODVM) [30], each node is allowed to participate in one route only, i.e., no common nodes are allowed between any two established routes other than the source and the destination nodes. Similarly, in the link-disjoint mechanism, e.g., ad hoc on-demand multipath distance vector (AOMDV) [29], each link is allowed to participate in only one route, i.e., no common links are allowed between any two established routes. According to [33], routes established

using the link-disjoint mechanism are only 15%–30% more stable than those established using the node-disjoint mechanism with a negligible difference in the average hop count.

Lee and Gerla [27] proposed an on-demand source routing scheme called split multipath routing (SMR). SMR establishes multiple routes of maximally disjoint paths between the source and the destination nodes. Data packets are then distributed among these multiple routes to avoid congestion and use the network resources efficiently. When a route disconnection occurs, the source node either initiates a new route discovery to replace the disconnected route or waits until all routes are broken before commencing a new route discovery process. It can be noticed that SMR produces high levels of routing control overhead because it starts a new route discovery when a link breakage occurs or when all routes are broken.

Yi *et al.* [34] proposed a multipath optimized link state routing (MP-OLSR) that combines proactive and reactive features. It sends out HELLO and topology control messages periodically to detect the network topology, similar to OLSR [35]. However, MP-OLSR only computes multipath routes when data packets need to be sent out, i.e., reactively. The entire route from the source to the destination is saved in the header of data packets. When an intermediate node receives a data packet, it checks the next-hop status in accordance with the source route before forwarding this packet. If the next hop is one of its neighbors, it forwards it; otherwise, the intermediate node recomputes the route and forwards the packet using the new route. It can be noticed that recomputing the route at intermediate nodes and changing it within the data packet header is not practical in a highly dynamic network such as the VANET. This method introduces high delays to the data-packet-forwarding process since at a node, a route that needs to be recomputed has to be recomputed for each data packet following that route.

In terms of QoS routing in VANETs, Rana *et al.* [38] utilized the vehicles' movements, density and velocities, and fading conditions to develop a hybrid multipath routing algorithm called mobility-aware zone-based ant colony optimization routing for VANET (MAZACORNET). The vehicular network is divided into multiple zones, where a proactive approach is used to find a route within a zone, and a reactive approach is used to find routes between zones. The link quality between the communicating vehicles is estimated using the link stability that is calculated using the velocity and position values of the vehicles, and the probability of successfully receiving the message, which depends on the distance between the vehicles lying within the same communication range, estimated using the Nakagami fading model [45]. MAZACORNET uses five different types of ants to perform the route discovery process. In addition, it uses two routing tables: intrazone and interzone routing tables. The intrazone routing table proactively updates the information within the zone using internal forward ants, which are transmitted every 20 s, whereas the interzone routing table updates the information between the zones on demand. MAZACORNET is suitable for dense network scenarios where a large number of vehicles exist within the zone. Due to the proactive approach used to update the intrazone routing table, MAZACORNET results in a high routing control overhead.

Correia *et al.* [39] proposed ACS mechanisms that utilize the information available in a VANET such as the vehicles' positions and velocities, to design an ant-based routing algorithm for VANETs. The proposed algorithm utilizes the route lifetime, which is predicted using the location and velocity information and the probability of reception of a message in a single-hop link, to indicate the level of pheromone to be deposited on that route. For multihop routes, the probability of reception is taken to be zero, and the predicted route lifetime will be the only indicator of quality. The evaporation mechanism is set such that the pheromone completely evaporates at the end of the route lifetime. Correia *et al.* adopted the dynamic MANET on-demand (DYMO) [46] routing protocol to propose their mobility-aware ant colony optimization routing DYMO (MAR-DYMO), where HELLO messages are modified by adding information on the vehicle's location and velocity to allow other vehicles to make predictions on its mobility. MAR-DYMO does not consider the unpredicted changes that may occur to the established routes or their quality. It extends the route lifetime when it is successfully used in the forwarding of data packets. When a link breakage occurs, MAR-DYMO initiates a new routing discovery at the source node to replace the invalid route, which results in high routing control overhead.

Li and Boukhatem [40] proposed a new adaptive multicriteria VANET routing protocol called the vehicular routing protocol based on ant colony optimization (VACO). VACO aims to find the best route from a source vehicle to a target roadside unit (RSU), which is at the closest intersection to the destination vehicle, in terms of latency, bandwidth, and delivery ratio. These metrics are combined to estimate the relaying quality of each road segment periodically using the on-going data flow. VACO combines both reactive and proactive approaches. At the beginning of the reactive route setup process, the source node generates several forward ants toward the target RSU to explore and set up the best route consisting of a list of intersections. Once the target RSU is reached, backward ants are generated and returned back to the source node. In terms of route maintenance, VACO implements a proactive approach by scheduling a periodic transmission of ants to explore and update routing paths by gathering the latest estimates of the relaying qualities of the road segments.

Finally, Li *et al.* [42] proposed IRQV, which is an intersection-based QoS routing protocol for VANETs in urban environments. Based on ACS mechanisms, IRQV computes the optimal route between two terminal intersections, which are the closest intersections to the source and the destination vehicles, respectively. The discovered route, which consists of a succession of intersections, is evaluated by the combination of two QoS metrics: connectivity probability and transmission delay. When forwarded between two neighboring intersections, data packets make use of a greedy carry-and-forward mechanism. At intersections, data packets are dynamically forwarded to the next intersection according to the maximum global pheromone from the current intersection to the terminal intersection of the destination. When the terminal intersections of the source and/or destination change, a new routing exploration is initiated at the source node to replace the invalid route. IRQV assumes that each vehicle is equipped with Global Positioning System

facility, a digital map, and a navigation system, and Wi-Fi access points are installed at each intersection to relay data packets and store routing information.

It can be noted that the current state of the art focuses on searching for feasible routes subject to a single or multiple QoS constraints and selects the best route with no consideration of the unpredicted disconnections that may occur due to the highly dynamic nature of VANETs. It mainly depends on estimating the link/route lifetime, which can be changed very quickly or suddenly. Moreover, there is no mechanism proposed for immediate response should the selected route fail. Most algorithms propose recalculating the best route again, which is not a practical option in VANETs. It implies a long delay in waiting for the new route to be established, and there is a possibility that by the time information on a discovered route is returned to the source node, the route could have failed, e.g., one of the intermediate node vehicles has left the route because it reached its destination or departed the route by turning off it. Moreover, this mechanism results in high routing control overhead.

To overcome these drawbacks, we propose the situation-aware routing concept to address the multiconstrained QoS routing problem in VANETs. Compared with multipath routing, where multipath routes are computed at the source node, and the entire route is replaced when a link breakage occurs, the situation-aware routing utilizes the SA concept to recover link failures by switching links or subroutes at or near the breakage point when it is possible. It strives to guarantee an immediate response at intermediate nodes rather than initiating a new route discovery process at the source node. This way, disruptions to data transmission are significantly reduced, as shown later via simulation results.

III. SITUATIONAL AWARENESS MODEL FOR MULTICONSTRAINED QUALITY-OF-SERVICE ROUTING IN VEHICULAR AD HOC NETWORKS

Here, we formulate the MCP problem in VANETs and propose the SA model for multiconstrained QoS routing in VANETs. For easy reference, Table I lists the notations used in this paper.

A. Problem Formulation

Let $G(V, E)$ be an undirected graph representing a vehicular communication network, where V is the set of vehicles, and E is the set of links connecting the vehicles. Let m denote the number of QoS constraints L_i, where $i = 1, 2, \ldots, m$. Each link between two vehicles C_1 and C_2, $l(C_1, C_2) \in E$ is associated with m weights corresponding to QoS constraints such that $w_i(C_1, C_2) \geq 0$. The MCP problem is to determine if there is a route P from a source s to a destination d such that all the QoS constraints are met, as described in the following equation:

$$w_i(P) \leq L_i, \quad i = 1, 2, \ldots, m. \tag{1}$$

If there is more than one route that satisfies the condition in (1), then the multiconstrained optimal path problem is to return the

TABLE I
NOTATIONS

s	The source node
d	The destination node
C_i, C_j, C_v	Intermediate nodes in the network
l_{ij}	Direct communication link between C_i and C_j
H	Wireless communication range
$P(s, d)$	The route connecting s and d
$r(l_{ij})$	The reliability value of link l_{ij}
$T_{ij}(t)$	The prediction interval for the continuous availability of l_{ij} at t
$G(V, E)$	A graph that represents the vehicular network
$M(s, d)$	The set of feasible routes between s and d
L	The QoS constraint, which can be an additive or multiplicative or concave constraint
Ψ_i	The tolerance factor of QoS constraint i
O_i	Weighting factor of QoS constraint i
$w_i(P)$	The weight value, *i.e.*, the QoS metric, of route P with respect to QoS constraint i
m	The number of QoS constraints
A_k	Ant k
$\tau_{ij}(t)$	The pheromone level associated with l_{ij} at time t
τ_0	The initial pheromone value on all established links
RT^i	The pheromone table at C_i
$p_{ij}^{A_k}$	The probability that A_k will choose C_j as next hop from the current node C_i
α, β	Constant values to control the relative importance of pheromone value versus expected link lifetime in the state transition rule
$\rho_{ij}(t)$	The evaporation rate of l_{ij} at time t
$F(P)$	The objective function of route P that satisfies the QoS requirements
U_0, U	Constant parameters that determine the relative importance of exploration versus exploitation in the state transition rule
$N(C_i^d)$	The set of neighbouring nodes of C_i over which a route to d is known and yet to be visited by A_k
$N(C_i)$	The set of all neighbouring nodes of C_i
t^{ex}	The pheromone evaporation interval
η	The number of evaporation process applications
τ_{ij}^{curr}	The current pheromone value after experiencing an evaporation
τ_{ij}^{new}	The new calculated pheromone value
$\mathcal{N}$	The number of processed routing replies at d

route that maximizes the objective function $F(P)$ as follows:

$$\arg \max_{P \in M(s,d)} F(P) \tag{2}$$

where $M(s, d)$ is the set of feasible routes between s and d, which meet the conditions in (1), and $F(P)$, which is the objective function, is defined in [47] as

$$F(P) = \sum_{i=1}^{m} O_i \left(\frac{L_i}{w_i(P)} \right) \tag{3}$$

where $0 < O_i \leq 1$ are weighting factors associated with each QoS constraint and depend on the transmitted traffic type. For instance, assume that voice data require two QoS constraints $L_1 = 100$ ms for end-to-end delay and $L_2 = 10$ hops for cost. The application could set $O_1 = 1$ and $O_2 = 0.5$ in (3) so that it favors the route that has the least end-to-end delay value since voice data are delay sensitive. For more details on the weighting factors, see [47].

Due to the current network state, computing feasible routes that satisfy all QoS constraints might not be always possible. In
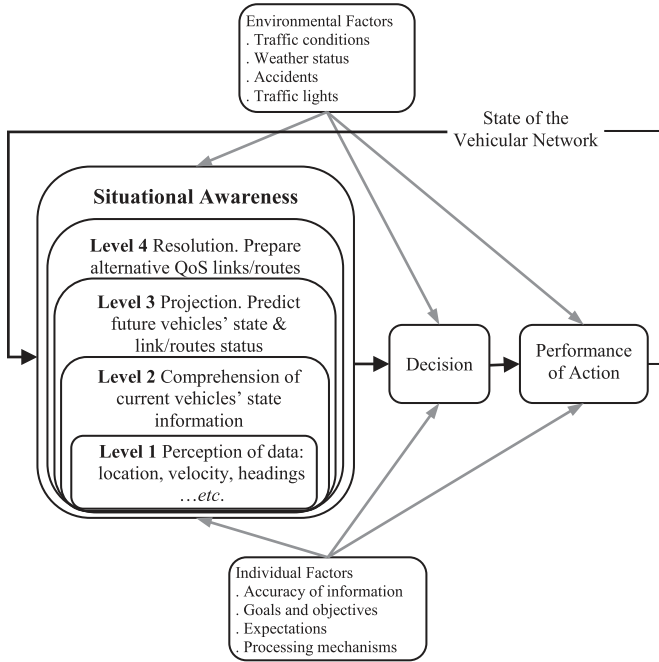
Fig. 1. SA model for QoS routing in VANETs.

this case, we propose a mechanism to allow applications to define tolerance factors that ease the QoS requirement restrictions. Such tolerance factors are only applied when the discovered route violates one or more QoS requirements according to (1). After applying tolerance factors, if a discovered route P still violates one or more of the QoS requirements, then it is discarded. Otherwise, it is added to $M(s, d)$ and follows the selection process in (2). Let Ψ_i be the tolerance factor for the constraint L_i, where $0 \leq \Psi_i < 1$. We extend (1) accordingly as follows:

$$w_i(P) \leq (1 + \Psi_i)L_i, \quad i = 1, 2, \ldots, m. \quad (4)$$

B. SA Model for Multiconstrained QoS Routing in VANETs

As we have previously mentioned, picking the best QoS route in the network does not guarantee a reliable data transmission. The routing algorithm needs to manage the risks inherent in making routing decisions due to unpredictable changes in the vehicular network. The SA concept provides this capability to the QoS routing algorithm. It considers the QoS routing as a continuous process where the routing algorithm selects the best route in the network according to (2), prepares certain countermeasures to be taken when the current route fails, and continues to evaluate the current solutions based on the state of the vehicular network. In the following, we discuss the four levels of the proposed SA model for multiconstrained QoS routing in VANETs, as shown in Fig. 1.

- Perception. Concerning QoS routing in VANETs, perception refers to knowledge of the vehicular network environment conditions such as vehicles' locations, headings, and velocities, which can be obtained from BSMs, vehicular velocity distribution, and traffic and weather conditions.

In addition, drivers' behavior with regard to their tendency toward acceleration/deceleration is also an important parameter, which the routing algorithm should be aware of. These parameters provide the information needed to determine the current state of the vehicular network and form the basis for the comprehension, projection, and resolution levels.

- Comprehension. Routing algorithms should understand and analyze the available information to provide an up-to-date picture of the current state of the vehicular network for the subsequent SA levels. For instance, to accurately estimate the link lifetime $T_{ij}(t)$ between two vehicles C_i and C_j in the projection level, the synthesis of information on location, heading, and velocity of each vehicle determines if they are moving toward each other or away from each other. This comprehension will be utilized in the projection level.

- Projection. With regard to QoS routing in VANETs, projection refers to the ability to forecast the future status of the vehicular network and predict the link lifetime and its QoS metrics based on the information synthesized from Level-2 SA. In this respect, projection tries to answer the following questions: How reliable is a link between two vehicles? When will this link fail? When will this established route cease to satisfy the QoS constraints? These projections are fed to the next SA level to enable appropriate decisions to be taken. Considering the example of accurately estimating the link lifetime $T_{ij}(t)$, let H denote the wireless transmission range and $v_i(t)$ and $v_j(t)$ the velocities of C_i and C_j at time t, respectively. $T_{ij}(t)$ can be accurately estimated using (5) with the appropriate parameters based on the feedback of the comprehension level rather than using a general calculation formula

$$T_{ij}(t) = \frac{H - \theta\sqrt{(y_i(t) - y_j(t))^2 + (x_i(t) - x_j(t))^2}}{|v_i(t) - \vartheta v_j(t)|} \quad (5)$$

where $\theta = -1$ and $\vartheta = 1$ when C_j overtakes C_i, $\theta = 1$ and $\vartheta = 1$ when C_i moves forward in front of C_j, $\theta = -1$ and $\vartheta = -1$ when C_i and C_j are moving toward each other, and $\theta = 1$ and $\vartheta = -1$ when C_i and C_j are moving away from each other.

- Resolution. This level refers to the actions required to recover a route between any two vehicles in case of a link failure. Based on the forecasts from the projection level, the routing algorithm prepares alternative QoS links/routes in case of failure of the current route. This task can be accomplished by determining a network of reliable QoS links and multipath routes among the communicating vehicles. The decision is then made to utilize the best available QoS link/route as primary, start data transmission and list its alternatives as backups according to their reliability and QoS metrics for use if required. After that, as the network topology changes rapidly, the routing algorithm continues to be aware of the state of the vehicular network to keep the information on the established links/routes up to date to ensure their validity.

The proposed model in Fig. 1 is specifically designed for VANETs to make QoS routing decisions more reliable. It helps the QoS routing algorithm establish reliable QoS routes and keep evaluating the current situation through the SA levels for an immediate response if the network topology changes. Sudden changes are considered while taking routing decisions to reduce their effects on data transmission. Location, velocity, heading, and acceleration/deceleration information is considered as low-level data when perceived by a vehicle, i.e., routing decisions cannot be made based on these data directly. Analyzing and synthesizing this information in the context of the vehicular network allows the QoS routing algorithm to consider the available options and be aware of how the current established links/routes evolve over time. At this point, the perception and comprehension levels are completed.

Since the movements of vehicles can be projected into the near future based on the comprehended information, the QoS routing algorithm can weigh its available options, i.e., links/routes, according to their QoS metrics and reliability. Thus, QoS routing decisions are taken based on cooperation between the first three levels of the proposed SA model. To mitigate the effects of unpredictable failures of the selected best route, a set of alternatives should be available to replace the failed link/route immediately without disrupting the current data transmission. This task can be accomplished by considering reliable multipath QoS routes at each node during the QoS routing process. The resolution level in Fig. 1 offers this capability and enables the QoS routing algorithm to respond to any link breakage near or at the node it occurs at. As a result, the QoS routing process in VANETs is significantly improved, as illustrated later through the simulation results.

It is worth noting that collecting and disseminating the information required to execute the proposed SA model successfully is performed before, during, and after completing the QoS routing process. This is accomplished through the exchange of BSMs and routing control messages, as explained later in Section V.

IV. SITUATION-AWARE MULTICONSTRAINED QUALITY-OF-SERVICE ROUTING ALGORITHM

Here, we develop the SAMQ routing algorithm that implements the proposed SA model in Fig. 1, but first, we briefly review the ACS rules for the SAMQ routing algorithm and the structure of the routing control messages and routing table in SAMQ.

A. ACS Rules

We recall that, in the ACS algorithm, a number of artificial ants build solutions to an optimization problem and exchange information on the quality of their solutions via a communication scheme that is reminiscent of that adopted by real ants [48]. The communication scheme comprises the following three conventional rules: the pheromone deposit rule, the state transition rule, and the pheromone evaporation rule. In the SAMQ routing algorithm, we utilize and extend the ACS rules defined in [47], which are adapted to the unique characteristics of vehicular networks and help reduce the probability of stagnation. Moreover, we devise a new rule called the QoS awareness rule.

1) Pheromone Deposit Rule: The level of pheromone τ_{ij} on a communication link/route between two vehicles C_i and C_j needs to reflect the quality of that link/route with respect to the considered QoS constraints. The reason pheromone levels are used is to quantify the link/route QoS metrics and normalize their values into a single quantity that can be used to compare the quality of a route with that of other routes. During the route discovery process, when ant A_k moves from C_i to C_j, it deposits a specific amount of pheromone τ_{ij} on the traversed link l_{ij}, which is calculated as follows:

$$\tau_{ij} = ar(l_{ij}) + b\sum_{x=1}^{m} \frac{L_x}{w_x(l_{ij})} \qquad (6)$$

where $a > 0$, $b > 0$, and $(a + b) = 1$. L_x denotes a QoS constraint, and $r(l_{ij})$ denotes the link reliability, which is defined as the probability that a direct communication link between two vehicles will stay continuously available over a specified time period [49]. It is assumed that τ_{ij} is based on the assumption of bidirectional communication links among vehicles so that this pheromone value applies when an ant moves from C_j to C_i. This way, we calculate the pheromone level of l_{ij} considering all its QoS metrics and its reliability value. Parameters a and b determine the relative importance of the link/route reliability versus its QoS metrics. Based on the nature of the transmitted data, a and b can be chosen. For instance, if the application wants to transmit video data, which requires highly reliable routes apart from the bandwidth and end-to-end delay constraints, the application can set $a = 0.65$ and $b = 0.35$. This way, among the routes that satisfy the QoS constraints, (6) favors the link/route with the highest reliability value. We worked out this function by experimentation, and its validity is illustrated by the simulation results presented in Section VI.

It is worth noting that the pheromone calculation in (6) and the objective function in (3) allow the SAMQ routing algorithm to take a set of different QoS constraints, including additive, multiplicative, and concave constraints. In the case of multiplicative constraints, e.g., a packet loss rate, we can take the logarithm of the multiplicative QoS metric on each link to transform it into an additive metric. However, in the case of concave QoS constraints, e.g., bandwidth, SAMQ can be set to return $P \in M(s, d)$ that maximizes or minimizes the concave QoS metric instead of applying (2) and selecting the route that maximizes the objective function in (3).

2) State Transition Rule: While searching for feasible routes, ants select their next hop when they arrive at intermediate nodes based on a stochastic mechanism called the state transition rule. When A_k arrives at an intermediate node C_i, if C_i's routing table RT^i does not contain routing information to d, A_k will be broadcast by C_i for reception by each one-hop away neighbor of C_i, as explained later in Section V. Otherwise, A_k selects C_j in RT^i as its next hop toward d according to (7) for exploitation or (8) for exploration, as shown below. Parameters U and U_0 determine the relative importance of exploitation versus exploration in the state transition rule. U is a randomly generated number uniformly distributed in [0, 1], and U_0 is a constant number selected between 0 and 1. If $U \leq U_0$, A_k prefers transition toward nodes that have larger

amounts of pheromone and longer link lifetimes, i.e., exploitation, as follows:

$$\arg \max_{C_j \in N(C_i^d)} \left\{ [\tau_{ij}(t)]^\alpha \, [T_{ij}(t)]^\beta \right\}. \qquad (7)$$

Here, $\tau_{ij}(t) \geq \tau_0$ is the pheromone level associated with link l_{ij}, where $\tau_0 > 0$ is the initial pheromone level for each link, and $T_{ij}(t) \geq 0$ is the predicted lifetime interval of l_{ij} at time t. α and β are parameters that control the relative importance of the pheromone level versus the predicted link lifetime. $N(C_i^d)$ is the set of neighboring nodes of C_i, each of which has a known route to d and is yet to be visited by A_k. Otherwise, if $U > U_0$, A_k will have the opportunity to explore new links, i.e., exploration, as follows:

$$p_{ij}^{A_k} = \begin{cases} \dfrac{[\tau_{ij}(t)]^\alpha [T_{ij}(t)]^\beta}{\sum\limits_{C_e \in N(C_i)} [\tau_{ie}(t)]^\alpha [T_{ie}(t)]^\beta}, & \text{if } C_j \in N(C_i) \\ 0, & \text{otherwise} \end{cases} \qquad (8)$$

where $p_{ij}^{A_k}$ is the probability that A_k selects C_j as its next hop from C_i toward d, and $N(C_i)$ is the set of all C_i neighbors.

Since the SAMQ routing algorithm is proposed to work in VANETs, parameters α and β in the transition rule in (7) and (8) reflect the importance of the predicted link lifetime $T_{ij}(t)$ versus the pheromone level $\tau_{ij}(t)$ when selecting the next node to traverse. According to (5), $T_{ij}(t)$ is calculated considering the current position, the relative velocity, and the direction of both vehicles. Therefore, it is vital for ants to traverse links that are expected to have longer lifetimes than others do particularly when the traffic density is low. This can be done by assigning a larger value to β to make ants avoid traversing vulnerable links that are prone to breakage and, consequently, avoid searching weak solutions.

With regard to parameter U_0, high values of U_0 mean that A_k prefers transition toward links that have a larger amount of pheromone and longer link lifetimes according to (7), i.e., exploitation. In this case, the probability of exploring new routes decreases, and the SAMQ routing algorithm could suffer from stagnation. On the contrary, small values of U_0 give A_k the opportunity to explore further links rather than just exploiting the pheromone level and following the trail, i.e., exploration. In the context of VANETs, selecting the constant value of U_0 depends on the vehicular network topology status, i.e., the extent of the environment dynamics and the performance gain of the SAMQ algorithm. For instance, if the network density is high, and the topology is stable, e.g., a highway during rush hour, it is preferable to choose a high value for U_0 since the communication links among vehicles are relatively stable. However, if the performance of the SAMQ routing algorithm decreases due to stagnation or the traffic density is low, then the U_0 value should be decreased to allow ants to explore new routes. We suggest letting the algorithm decide and adjust the value of U_0 depending on the performance gain and the vehicular network topology dynamics.

3) Pheromone Evaporation Rule: The pheromone evaporation process is very important to avoid rapid convergence toward a suboptimal search space and to explore new routes. After building $M(s, d)$, for every t^{ex} seconds, each node

decreases the pheromone level of all its utilized links, either a primary or a backup, using the following formula:

$$\tau_{ij}(t + t^{ex}) = \begin{cases} (1 - \rho_{ij}(t)) \, \tau_{ij}(t), & 0 < \rho_{ij}(t) < 1 \\ \tau_0, & \text{otherwise} \end{cases} \qquad (9)$$

where $\tau_{ij}(t)$ is the pheromone level at time t, $\tau_{ij}(t + t^{ex})$ is the updated pheromone level at time $(t + t^{ex})$, and $\rho_{ij}(t)$ is the evaporation rate where $0 < \rho_{ij}(t) < 1$ and is calculated as follows:

$$\rho_{ij}(t) = 1 - \sqrt[n]{\frac{\tau_0}{\tau_{ij}(t)}} \qquad (10)$$

where η is the number of times (9) has been applied. It should be noticed that when the established link is no longer valid for use, i.e., it no longer satisfies the required QoS constraints, its pheromone value τ_{ij} is assigned to τ_0, and no more evaporations are applied.

4) QoS Awareness Rule: The QoS metrics associated with the current established routes can change rapidly, and routes can quickly become inefficient or even infeasible in VANETs. Therefore, we devise the QoS awareness rule to support the determining of the continuing feasibility of routes established by the SAMQ routing algorithm and the updating of their calculated QoS metrics, i.e., pheromone levels, to avoid stagnation.

Once s starts transmitting data packets, the QoS awareness rule takes effect. For each node that belongs to an established route $P \in M(s, d)$, the information received via BSMs is utilized to reevaluate its communication links and update their pheromone levels. We assume that each node is aware of the QoS constraints required for the current established links in its routing table. While data packets are transmitted from s to d, each node C_i reevaluates the QoS metrics of l_{ij} upon receipt of a BSM from C_j as follows. It calculates a new pheromone level τ_{ij}^{new} using (6). If $\tau_{ij}^{\text{new}} > \tau_{ij}^{\text{curr}}$, where τ_{ij}^{curr} is the current pheromone level, C_i increases the pheromone level on l_{ij} to enforce its presence as follows:

$$\tau_{ij}^{\text{curr}} = \tau_{ij}^{\text{curr}} + \left(\tau_{ij}^{\text{curr}} \left(1 - \frac{\tau_{ij}^{\text{curr}}}{\tau_{ij}^{\text{new}}} \right) \right). \qquad (11)$$

Otherwise, it decreases the pheromone level as follows:

$$\tau_{ij}^{\text{curr}} = \tau_{ij}^{\text{curr}} - \left(\tau_{ij}^{\text{curr}} \left(1 - \frac{\tau_{ij}^{\text{new}}}{\tau_{ij}^{\text{curr}}} \right) \right). \qquad (12)$$

Finally, the updated pheromone level is utilized to update the evaporation rate $\rho_{ij}(t)$ for this link according to (10).

This rule plays a fundamental role in realizing the implementation of the SA resolution level in Fig. 1. Along the current route from s to d, when C_i senses that the next link toward d is going to disconnect or its pheromone value has reached a certain pheromone limit, i.e., near complete evaporation, it switches to a backup link/route toward d, if such a backup exists. Then, it sends a notification message to inform s about the occurred switch so that it can decide whether to continue using the backup route or initiate a preemptive route discovery process. In all cases, the broken route will be removed from $M(s, d)$. It is worth noting that this process is performed

during the data transmission where vulnerable or infeasible links/routes are proactively replaced. However, there is still a slight possibility that the data transmission will be interrupted. This point is further illustrated in Section V.

B. Routing Control Messages and Routing Table in SAMQ

To fulfil the requirements of the proposed model in Fig. 1, the routing control messages, i.e., ants, in the SAMQ routing algorithm include the mechanism for processing multiple routing requests, routing replies, and routing error messages. Apart from these three conventional control messages, we add a notification message, which is designed to inform s about route switching at intermediate nodes.

1) SARQ: In addition to the default fields of a conventional routing request message such as the destination address and the originator address, the following fields are added to SAMQ routing request ant (SARQ).

- *SARQ_Gen* indicates the current ant generation. Different ant generations could be involved in the route discovery process of the same destination. This field plays a key role in decreasing the proliferation rate of ants, i.e., if C_i receives another ant from the same generation looking for the same destination, it may only be processed if it presents a better route than the existing route. Otherwise, it is discarded. This contributes to a low routing control overhead in SAMQ.
- *QoS_Metrics* contains the reliability and QoS metrics of the route that the SARQ has traveled so far.
- *QoS_Constraints* contains the QoS constraints that should be satisfied. These QoS constraints are necessary to calculate the pheromone value of the traversed link.
- *QoS_Tolerance_Factors* contains the tolerance factors if they are allowed. Otherwise, it is empty.
- *Kinematic information* contains the coordinates, current velocity, and heading of the vehicle that generates/ processes the SARQ.

2) SARP: The SAMQ routing reply ant (SARP) is designed to set up forward routes to d considering the quality of the links it has traversed. The SARP message includes the following fields in addition to the default fields of a conventional routing reply message.

- *SARP_Gen* indicates the current ant generation, which matches that given in the *SARQ_Gen* field of the SARQ, which generated it.
- *QoS_Metrics* contains the reliability value and QoS metrics of the corresponding computed forward route.
- *QoS_Constraints* and *QoS_Tolerance_Factors* fields contain the same information as that given in the corresponding SARQ.

3) SARE: The SAMQ routing error ant (SARE) message is designed to announce a link breakage when it occurs. It traverses back along the nodes of the route to the node that became unavailable due to a link breakage. The SARE contains a list of addresses of the destination node(s) that become unreachable due to the link breakage and a list of nodes that fail to recover the link breakage so that s avoids switching to routes

that go through these nodes. This is explained later as part of the discussion of the route maintenance process in Section V.

4) SANA: The SAMQ notification ant (SANA) message is designed to follow the trail of the selected backup route back to s. It contains information on the broken link/route and the quality of the established backup route that replaces the broken route. For that purpose, it contains the same information found in the SARQ message.

Routing tables also play an important role in the SAMQ routing algorithm to route data packets and routing control messages to their destinations efficiently. To fulfil the requirements of the SAMQ routing algorithm, routing table entries include the following information in addition to conventional fields, such as the destination address, the next-hop address, and cost.

- *QoS_Metrics* contains the reliability value and the weight value of each QoS constraint associated with this entry.
- *QoS_Constraints* contains the QoS constraints that are used to estimate the pheromone level associated with this entry.
- *rt_pherm* contains the pheromone level associated with this entry calculated according to the defined QoS constraints.
- *rt_evp* contains the evaporation rate of this entry. Each node uses this field to ensure that the routing table entry has evaporated at the end of its expected lifetime.
- *rt_state* indicates the state of the route entry, i.e., a primary route or a backup route. This indicator is updated upon discovering a better route in terms of its objective function value, which is defined in (3), or when the primary route fails and is replaced by a backup route.

C. SAMQ Routing Algorithm

The route discovery process in the SAMQ algorithm aims to determine the set of all feasible routes between s and d, $M(s, d) = \{P_1(s, d), P_2(s, d), \ldots, P_z(s, d)\}$. In addition, each intermediate node C_v also builds its routes to neighboring nodes in the same way, i.e., reliable multipath QoS routes. When s has data to send and no route to d is found, it starts by broadcasting a routing request message SARQ to its one-hop away neighboring nodes. The SARQ message includes the kinematic information of s together with the QoS constraints, which are required to be met by the data type s needs to send. These SARQs are forwarded in the vehicular network and mark the links of their routes with an amount of pheromone that reflects the quality of the traversed links/routes and their reliability. At each node, SARQ can only proceed if the route traveled so far satisfies the QoS constraints according to (1) or (4) if QoS tolerance is allowed. Once d is reached, a SARP message is generated to traverse back to s following the trail of the computed route. Later on, the next generation of routing requests is attracted by pheromone levels deposited along the computed feasible routes; hence, they continue the quest for good solutions. However, the relative importance of exploitation versus exploration, as determined by U and U_0, and the relative importance of $T_{ij}(t)$ versus $\tau_{ij}(t)$, as determined by α and β in (7) and (8), determine the next node a SARQ is going to traverse. Once all the control messages,

i.e., SARQs and SARPs, have finished their tours, s has built $M(s,d)$, which is the set of feasible routes that satisfy the defined QoS constraints. The SAMQ routing algorithm starts data transmission once a route that satisfies the QoS constraints is established to reduce the transmission delay and accelerate the convergence rate. Later on, for each new established route $P \in M(s,d)$, s calculates the corresponding $F(P)$ value according to (3). The best route among the established routes is then selected based on (2) or to minimize/maximize the concave QoS constraint if it is presented. The pseudocode in Algorithm 1 illustrates the design of the SAMQ routing algorithm when the best route is selected based on (2).

The "for" loop between steps 1–3 computes the route with the maximum pheromone value from s to every other network node C_v and then from C_v to d to compute the best route with respect to the defined QoS constraints. This way, $O(|V|)$ concatenated routes, i.e., in the form of $P(s, C_v) + P(C_v, d)$, are returned for checking against the required QoS constraints. This check is performed in steps 6–9 using the QoS-Satisfied function. Each route obtained from the concatenation of routes with the maximum pheromone values at node C_v, which connects s to d via C_v, is inspected to see if it satisfies all QoS constraints. If so, it is also checked to see if it gives the highest current corresponding value of $F(P)$ calculated according to (3). The *MaxPherm* variable, which is defined in step 5, is utilized to help accomplish this task. In step 10, the best route according to (2) is returned. Note that if the application allows QoS constraints tolerance, the function QoS-Satisfied is modified to check the QoS constraints according to (4).

Algorithm 1 SAMQ Routing Algorithm

/* $P(s,s) = $ Null, $P(d,d) = $ Null, $M(s,d)$ is empty */
1. **For** each node $C_v \in V/\{s, d\}$ **do**
2.   Compute the route with the maximum pheromone value from s to C_v;
3.   Compute the route with the maximum pheromone value from C_v to d;
4. $P(s,d) = $ Null; $F(P(s,d)) = 0$;
5. MaxPherm $= 0$; // Used for selecting the best route according to (2)
6. **For** each node $C_v \in V/\{s, d\}$ **do**
7.   if (QoS-Satisfied$(s, d, L_1, L_2, \ldots, L_m, C_v)$) and $(F(P(s, C_v) + P(C_v, d))) \geq $ MaxPherm) **then**
8. $P(s,d) = P(s, C_v) + P(C_v, d)$;
9.   MaxPherm $= F(P(s,d))$;
10. **return** $P(s,d)$;
/* Return true if QoS satisfied check is passed */
Function QoS-Satisfied$(s, d, L_1, L_2, \ldots, L_m, C_v)$
11. bool result $=$ true;
12. **if** $w_1(P(s, C_v) + P(C_v, d)) > L_1$ **then**
13.   **return** result $=$ false;
14. **else if** $w_2(P(s, C_v) + P(C_v, d)) > L_2$ **then**
15.   **return** result $=$ false;
16. **else if** $w_m(P(s, C_v) + P(C_v, d)) > L_m$ **then**
17.   **return** result $=$ false;
18. **return** result;

D. Properties of SAMQ Routing Algorithm

Property 1: The SAMQ routing algorithm always terminates. If there is no route that satisfies the defined QoS constraints, SAMQ terminates and returns null in step 10. Otherwise, it returns the best feasible route according to (2) computed in steps 6–9.

Property 2: For a node $C_v \in V$, if the following inequality holds for any $i \in \{1, 2, \ldots, m\}$:

$$w_i (P(s, C_v) + P(C_v, d)) > L_i \qquad (13)$$

C_v cannot be part of any feasible route between s and d at this time. Due to the highly dynamic nature of vehicle movements, C_v might be eligible to be part of a feasible route computed later.

Property 3: The SAMQ routing algorithm computes a feasible route $P(s,d)$ that satisfies the QoS constraints and maximizes the value of $F(P)$ if and only if such a route exists during the route establishment process. It is worth noting that there is a possibility that by the time information on a discovered route is returned to s, the route could have failed, e.g., one of the intermediate node vehicles has left the route because it reached its destination or departed the route by turning off it. Therefore, s processes and keeps all the discovered routes to deal with such a situation. *Property 1* proves the "*only if*" part because if no feasible route is found, SAMQ terminates. Steps 6–9 check repeatedly for each node $C_v \in V/\{s, d\}$ that its primary concatenated route satisfies the QoS constraints and that its objective function value, which is calculated in (3), is at least as large as those of the other available routes. If such a route exists, it is returned as the best route according to the considered QoS constraints.

E. Complexity of SAMQ Routing Algorithm

We discuss the complexity of the SAMQ routing algorithm in terms of communication and computational complexity levels. In the context of communication complexity, the worst-case complexity is $O(|V| + |E|)$ messages, i.e., ants, where $|V|$ is the number of network nodes, and $|E|$ is the number of communication links. The number of ants needed to compute a feasible route is proportionate to the number of links that compose the route. Each node requires, at most, one ant, either an SARQ or an SARP, to add one link to its pheromone table. Computing the longest route in the network, which contains, at most, $|V|$ nodes and $|V - 1|$ links, requires $O(V)$ ants in the worst case. Therefore, it requires $O(|V| + |E|)$ messages to build $M(s,d)$. It should be noticed that in the SAMQ-based routing protocol, we limit the listed links/routes to the same destination at each intermediate node to two routes for reasons given later in Section V-A. This way, SAMQ scales well for large vehicular networks in terms of control message overhead. This point is further supported via simulation results presented later in this paper. Here, the BSMs are not considered as part of the routing algorithm communication overhead.

In terms of the computational complexity of SAMQ, upon receiving a control message, i.e., SARQ, SARP, SARE, or BSM, each node performs the same amount of calculations to

update the corresponding pheromone table entries. Thus, the computational complexity depends on the size of the routing table RT^i at vehicle C_i. Let us assume that we have a mesh vehicular network topology where the size of RT^i will be $|RT^i| = (|V|(|V|-1)/2)$. For each received new message, C_i performs $O(|RT^i|)$ calculations to update the pheromone table entries, i.e., the computational complexity is $O(|V|^2)$. It should be noted that the number of vehicles that can form the network is controlled by the environmental conditions, e.g., the highway capacity and the communication range of vehicles. Moreover, once the pheromone level on a link/route entry has completely evaporated, it is no longer included in RT^i. Therefore, the computational complexity of SAMQ does not increase much because $|V|$ cannot be too big, and RT^i does not grow quickly. Nonetheless, vehicles are expected to be equipped with high-speed processors to cope with the requirements of vehicular communication.

V. SITUATION-AWARE MULTICONSTRAINED QUALITY-OF-SERVICE-BASED ROUTING PROTOCOL FOR VEHICULAR AD HOC NETWORKS

Here, we develop the SAMQ-based routing protocol that implements the SAMQ routing algorithm proposed in the previous section. In the following, we describe the route discovery and the route maintenance processes of the SAMQ-based routing protocol.

A. Route Discovery in SAMQ-Based Routing Protocol

The route discovery process in the SAMQ-based routing protocol aims to compute feasible routes that satisfy the QoS constraints from s to d and select the best route, if such a route exists. When s has data to send to d and $M(s,d)$ is empty, it issues a new route discovery process by broadcasting a single-hop SARQ message to its neighbors. As we have previously mentioned, the exchange of BSMs ensures that each node is aware of its neighbors' states, i.e., their locations, velocities, and headings, in its vicinity. These BSMs are not considered as a part of the routing control overhead since they are periodically exchanged in the network to comply with the safety application requirements. Upon receipt of a BSM, each node records the corresponding node it heard from and creates a new link to the node with τ_0 assigned in its routing table. During the route discovery process, if this link is utilized, then it is evaluated against the required QoS constraints, and its pheromone value is estimated. There is no need to broadcast the SARQ any further as the one-hop away neighbor node is responsible for taking the following routing decision. When a neighboring node C_v receives the SARQ, there are two different paths the algorithm will follow depending on the answer to the following question: Has this SARQ been processed before or not?

If not, the QoS metrics and reliability values are calculated, and the link $l(C_v, s)$ is evaluated against the QoS constraints in SARQ. If $l(C_v, s)$ violates any of the QoS constraints determined by the data traffic type, even after applying the given QoS tolerance criteria, this SARQ is not registered as processed and discarded. This will allow C_v to process other routing requests because they may report acceptable QoS properties. If $l(C_v, s)$
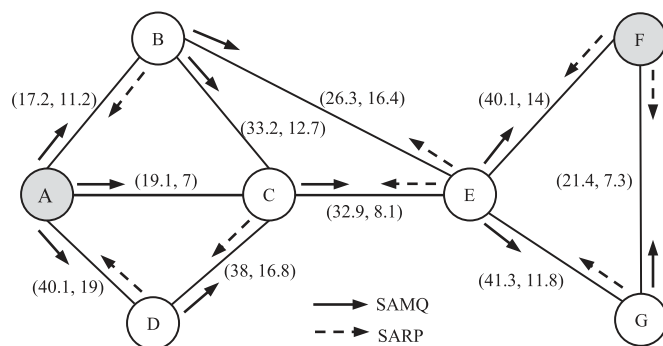

Fig. 2. Route discovery process in SAMQ-based routing protocol at time t.

satisfies the QoS constraints, C_v inserts this link as a primary in its routing table, registers the SARQ as processed, and updates it to be forwarded.

If it has been previously processed, the computed QoS metrics and reliability values are compared with those held in the existing primary route entry to the same destination with the same set of QoS constraints. If the route reported by this duplicate SARQ is better than the existing route, i.e., its $F(P)$ value is higher, the existing route entry state will be updated to a backup, the discovered route will be inserted as a primary, and the SARQ will be updated and forwarded. Otherwise, the discovered route information will be either inserted into the backup route entry or used to update it or discarded as appropriate, and the SARQ will be dropped. The reason for dropping the SARQ here is that we limit the listed links/routes to the same destination at each intermediate node to two routes only, i.e., primary and backup, to avoid the complexity of listing every route in the network and to reduce the routing control overhead. Finally, if the SARQ is to be forwarded, C_v checks its routing table RT^v for route entries to d. If an entry is found, the SARQ is forwarded based on the transition rule defined in (7) and (8). Otherwise, it is single-hop broadcasted to the neighbors of C_v.

Once d is reached, a SARP is generated and sent back to s with the corresponding QoS information. The destination node is allowed to process up to $\mathcal{N}$ routing requests and send a SARP for each if the discovered route satisfies the QoS requirements. SARPs follow the trail of their corresponding SARQs to arrive back at s. Once a SARP is received at s, the application starts data transmission. If SAMQ becomes aware of another route, then s selects the route that maximizes the $F(P)$ value defined in (3), and so on. In both cases, all feasible routes, i.e., $M(s,d)$, are kept at s for further use if needed as to be explained later in the route maintenance process. This is done to avoid the transmission delay that would occur if s waited until two or more replies had arrived before transmitting data and to accelerate the convergence rate of the SAMQ routing algorithm.

Fig. 2 shows an example of a route discovery process from source vehicle A to destination vehicle F using the SAMQ routing algorithm. Each link is associated with the following 2-tuple $(T_{ij}(t), \tau_{ij}(t))$, i.e., the link lifetime and its pheromone value. In Fig. 2, the SARQ is broadcasted at vehicle B because it does not have a route to F. Two SARQs arrive at E. One of them exploits the pheromone trail and disseminates through the link $l(E, F)$, whereas the other explores a new link $l(E, G)$ and

arrives at F from G. Two SARPs are then generated at F and traverse back to A along the feasible routes P_1 (A, D, C, E, F) and P_2 (A, B, E, G, F) discovered by the two SARQs, respectively. Assuming that the SARP via P_1 arrives at A first, A starts data transmission over P_1. When receiving the other SARP through P_2, A saves P_2 as a backup since, for the objective function in (3) in this example, P_1 gives a higher value than P_2.

Unlike conventional ACS algorithms, SARPs are not allowed to enforce pheromone levels on the route they traverse back to s. Since the pheromone value is an estimation of the link QoS metrics and its reliability, pheromone updates should not be performed using constant values or constant evaporation parameters. Instead, the reevaluation of the current established links is a task that we suggest the QoS awareness rule performs, as previously explained in Section IV-A.

B. Route Maintenance in SAMQ-Based Routing Protocol

In the SAMQ-based routing protocol, the route maintenance process follows two approaches, namely, reactive and proactive. This process represents the implementation of the SA resolution level in Fig. 1. It mainly aims to resolve a link/route disconnection locally and immediately rather than sending SARE messages to s and interrupting the current data transmission. Using the QoS awareness rule, SAMQ will be able to proactively respond to any possible link/route disconnection or infeasibility in terms of QoS constraint satisfaction. It first tries to replace the broken link, which is a primary one, with a backup link/route to the same destination if it is available, valid to use, and still satisfies the QoS requirements. In case of no feasible backup links/routes to switch to, the route maintenance process sends a SARE message to the precursor node assigned with the broken route. The precursor node invalidates the routing table entry that uses the broken link and tries to switch to an available valid backup route that satisfies the QoS requirements. If no feasible backup route is found, the SARE message is forwarded to the next precursor node, and so on, until s is reached.

If $M(s, d)$ is empty, s starts a new route discovery process. Otherwise, switching to a backup route is commenced. The selected backup route should not include any of the nodes that fail to recover the occurred link breakage. This can be guaranteed using the information within the received SARE message. The mechanism of processing link breakages locally at the intermediate node level saves available bandwidth and, as long as valid backup routes exist, provides a reliable uninterrupted data transmission service due to the application of the SA model in Fig. 1.

As we have previously hinted, there is a possibility that data transmission could be interrupted. This situation could happen when the reception rate of BSMs is low. In this case, when a link breakage occurs, the intermediate node does not switch to an outdated link/route but issues a SARE message to the precursor node to switch to another backup route. If no updated backup route exists, the process continues until s is reached. Within this time, the data transmission will be interrupted because the destination is unreachable, and it has to wait until a new route becomes available. To alleviate this problem, SAMQ could allow intermediate nodes to exchange update messages, which contain the same information as BSMs, for a specific period of time when it is sensed that the reception rate of BSMs is low. This period can be varied and depends on the current network conditions. However, it is highly likely that the reception rate of these update messages could be low as well. Therefore, SAMQ does not adopt this approach and prefers to initiate a new route discovery process in this situation.

VI. PERFORMANCE EVALUATION OF SITUATION-AWARE MULTICONSTRAINED QUALITY-OF-SERVICE ROUTING ALGORITHM

We conducted our performance evaluation using the OMNet++ 4.6 network simulator [50]. We performed 20 runs of each simulation and obtained the average of the runs' results.

A. Simulation Setup

The simulations were run on a six-lane traffic simulation scenario of a 10-km highway with two independent driving directions in which vehicles move. The average velocity of vehicles in each lane is 40, 60, and 80 km/h, respectively. We use the highway mobility model developed in [51], which is built based on traffic theory rules and considers the drivers' behavior. The highway scenario used in this evaluation is of a straight stretch of highway, which does not have hard bends or roundabouts or curves. However, when curves are present in the highway, the link reliability estimation will, in general, not be significantly affected since the only new variable will be the left or right velocity component of vehicle movement. Moreover, the left or right velocity component generally does not affect the link lifetime estimation because it is usually small in value in comparison to the forward or backward velocity component and communication range.

The source/destination pairs of vehicles are randomly selected for each simulation run, where we assume that each destination is reachable through multihop routing from the source. However, when the traffic density is extremely low and no communications links are available, i.e., the destination is not directly reachable via multihop routing, the SAMQ routing algorithm can be extended to use a store–carry–forward mechanism if the application can tolerate the delay constraint to some extent, i.e., $\Psi_i > 1$ for the delay constraint in (4). Knowledge of the destination location is required to use a store–carry–forward mechanism efficiently. In this case, the pheromone deposit rule in (6) can be extended to give higher pheromone values to links/routes that go through vehicles that are expected to reach the destination vehicle quickly based on its current location, velocity, and direction. In addition, parameter U_0 will be given a high value in the state transition rule to ensure quick data delivery to the destination by exploiting the communication links that have high pheromone values, which now reflects that these links/routes could lead to the destination vehicle quickly. It can be noted that the SAMQ routing algorithm can adapt to this situation due to the flexible design of its ACS rules that can incorporate more parameters. The assistant of road infrastructure, e.g., RSU, might be needed to acquire the location information of the destination node. Extending the SAMQ routing algorithm for vehicular delay-tolerant networks needs more investigation and is left for our future work.
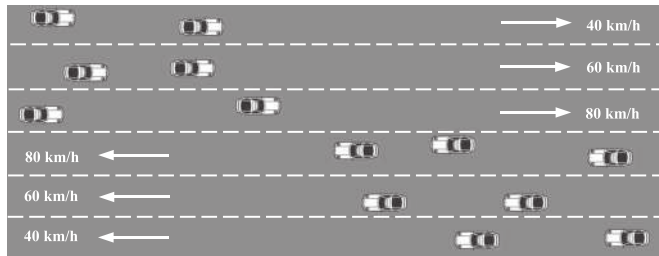
Fig. 3. Simulation scenario.

The MAR-DYMO [39], VACO [40], and SAMQ routing algorithms were evaluated in the simulations. Although the VACO routing protocol is proposed for urban environments, it fits the purpose of our performance evaluation in a highway scenario. The reason is that VACO follows a reactive approach of setting up a route by forwarding several ants toward the destination and a proactive approach of updating the quality of the established routes by periodically estimating the quality of road segments using the on-going data flow. These bring VACO some features related to our SAMQ routing algorithm; hence, the performance comparison between them can clearly demonstrate which algorithm is more effective in the highway scenario. Fig. 3 shows the simulation scenario.

The following performance metrics are considered in the simulations.

- Average packet delivery ratio (PDR): This metric represents the average ratio of the number of successfully received data packets at d to the number of data packets sent.
- Routing control overhead: This metric expresses the ratio of the total number of routing control messages generated, including SARQs, SARPs, SAREs, and SANAs, to the total number of data messages sent. BSMs are not counted in this metric.
- Average transmission delay: This metric represents the time needed to perform the route discovery process and compute the first feasible route that satisfies the QoS constraints, i.e., the time interval between sending a routing request from s and receiving the first routing reply from d.
- Average dropped data packets ratio: This metric shows the average ratio of the number of dropped data packets to the number of successfully received data packets. Data packets are dropped at d because they violate the defined end-to-end delay constraint. This metric demonstrates how effectively each concerned routing algorithm identifies a feasible route. Ideally, the routing algorithm should provide routes that have zero-dropped-data-packet ratio.

The simulation parameters are summarized in Table II.

B. Simulation Settings

A 10-MB video file is generated to be streamed from three video stream servers, i.e., senders, along the road to many video stream clients, i.e., receivers, which request this file. The number of clients is randomly chosen between five to ten clients for each simulation run. Both servers and clients are

TABLE II
SUMMARY OF THE SIMULATION PARAMETERS

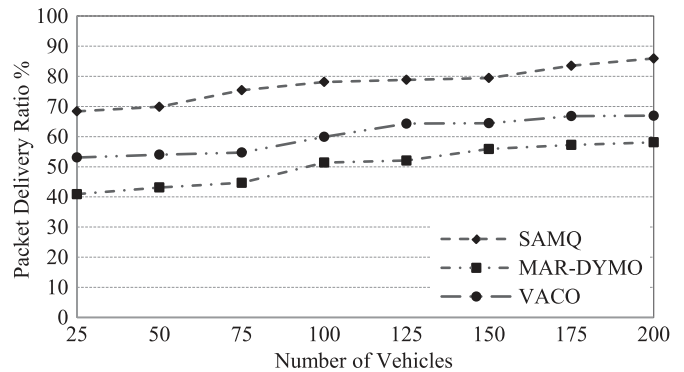| Simulation Area	1 *km* x 10 *km*
Mobility Model	Highway
Communication Range	450*m*
MAC Layer	IEEE 802.11p
Vehicles' velocities	Normally distributed
Vehicles' distances	Exponentially distributed
Number of runs	20
Simulation duration	300 seconds
SAMQ parameters	$U_0 = 0.6, \alpha = 0.45, \beta = 0.55, \tau_0 = 5, \mathcal{N} = 5$

Fig. 4. Simulation A—PDR.

vehicles moving on the highway but not necessarily in the same direction. The transmission data rate is 0.5 Mb/s, which is the recommended data rate at which to watch a YouTube video [52]. Such a video transmission could take place when a customer requests to download a video file from a mobile Internet gateway or when a video clip about an incident is streamed to police vehicles in the area to enable assessment of the situation before approaching the incident. The QoS constraints are $L_1 = 300$ ms for end-to-end delay and $L_2 = 10$ hops, whereas the tolerance factors are $\Psi_1 = 0.5$ and $\Psi_2 = 0.6$. The application sets the weighting factors $O_1 = 0.6$ and $O_2 = 0.8$ for the video data transmission. Finally, since route reliability is an essential requirement for video transmission, we set $a = 0.65$ and $b = 0.35$ in (6) for the pheromone value estimation. The following simulations are performed.

- Simulation A—We change the number of vehicles on the highway from 25 to 200 vehicles. The average velocity of vehicles for each lane is 40, 60, and 80 km/h, respectively.
- Simulation B—We change the average velocity of the vehicles in the third lane only from 60 to 130 km/h. The number of vehicles on the highway is 120 vehicles.

C. Simulation Results

1) Simulation A—Effects of Network Density: Fig. 4 shows the simulation results for the routing algorithms examined in this simulation. Generally, higher network density contributes to a better PDR of the routing algorithms because more vehicles imply more potential links; hence, there are more options from which to compute feasible routes to the destination. Moreover,
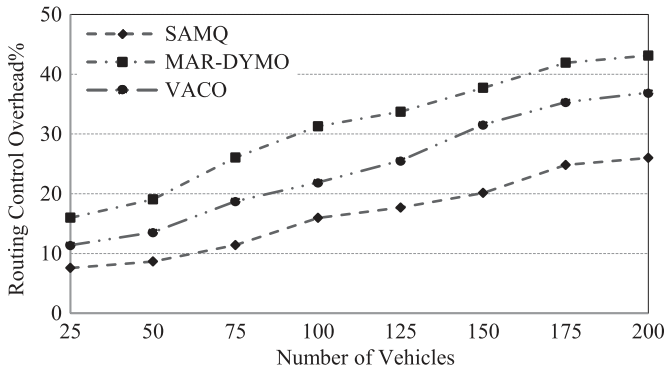
Fig. 5. Simulation A—Routing control overhead.

Fig. 6. Simulation A—Transmission delay.

Fig. 7. Simulation A—Dropped data packets.

higher network density improves the connection duration in the opposite direction for vehicles, which helps increase the PDR as well.

It can be observed in Fig. 4 that the PDR of the SAMQ routing algorithm is higher than those of VACO and MAR-DYMO over different densities for video data transmission. The video stream application requires a large data packet size, which is set to 6250 bytes in this simulation to accommodate the high data transmission rate. Moreover, video data transmission requires the routing algorithm to maintain the most reliable routes in the network. When the network density increases, the established routes may become longer and, consequently, less reliable. If one of the data packet's fragments is not delivered, the entire data packet is dropped. It can be seen from the high PDR achieved by SAMQ in Fig. 4 that it avoids this scenario through its dynamic pheromone-based evaluation and applying the SA model in Fig. 1 to stay aware of the quality of the established routes. Applying the SA model in Fig. 1 allows SAMQ to build a network of reliable QoS links/routes among the communicating vehicles. This way, different communicating vehicles, i.e., sender/receiver pairs, can benefit from the already established links/routes to forward their data.

Fig. 5 shows the routing control overhead ratio generated by each routing algorithm examined in this simulation. It is anticipated that the routing control overhead would increase when the network density increases because more nodes are available for routing requests to traverse. The SAMQ routing algorithm maintains the least routing control overhead in comparison to VACO and MAR-DYMO.

As previously mentioned, SAMQ allows nodes to broadcast SARQs only to nodes a single-hop away. Each single-hop away neighbor that receives this SARQ does not necessarily rebroadcast it unless there is no link/route to the destination node. This method of disseminating routing requests complies with the strategy of distributing routing decisions among intermediate nodes rather than at the source node. Therefore, unlike MAR-DYMO and VACO, the probability of broadcasting routing requests is thus kept down. Moreover, SAMQ utilizes the selected route as long as it is still feasible using the QoS awareness rule and deals with link/route breakages locally with no need to transmit SARE messages or initiate a new routing discovery process. These enable SAMQ to achieve the least routing control overhead as shown in Fig. 5, resulting in more
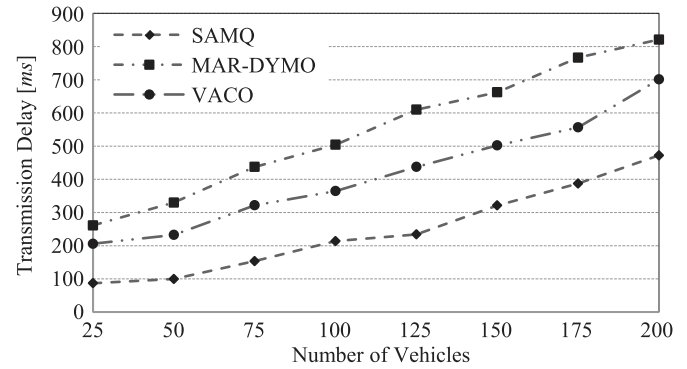
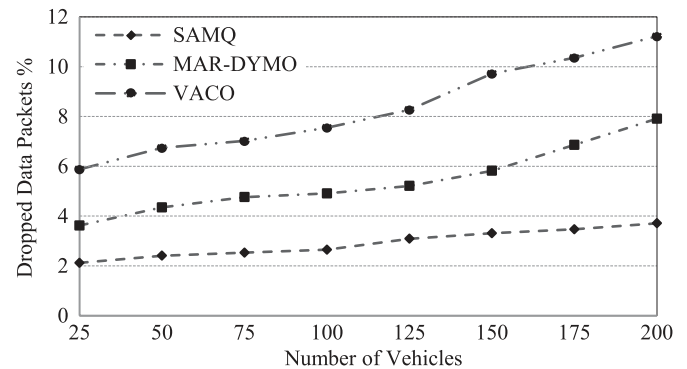bandwidth for data packet transmission and, hence, the higher PDR shown in Fig. 4.

Fig. 6 shows the average transmission delay for each of the examined routing algorithms. This figure shows how fast each routing algorithm can converge and starts data transmission. In general, higher network densities contribute to higher transmission delays since the routes between the communicating vehicles might be longer, depending on their locations. SAMQ has the advantage of lower transmission delays because it does not wait to find the global optimal route but starts data transmission once a feasible route becomes available. If a better route in terms of its QoS metrics and reliability emerges, SAMQ switches to use this route. The SAMQ routing algorithm's awareness of the current network situation and response to link/route breakages at or near the breakage point has a noticeable impact on decreasing the rate of initiating new route discovery processes and, therefore, decreasing the transmission delays. Furthermore, the computed feasible links/routes at intermediate nodes can be utilized by different source/destination pairs to decrease the time needed to perform the route discovery process and quickly start data transmission.

Finally, Fig. 7 clearly shows the advantage of SAMQ in avoiding dropped data packets in comparison to MAR-DYMO and VACO. To avoid higher rates of dropped data packets at the destination node, the selected feasible route should be able to deliver data packets according to their QoS requirements. The performance of SAMQ in this figure shows the efficiency of its ACS rules in identifying feasible routes that satisfy the
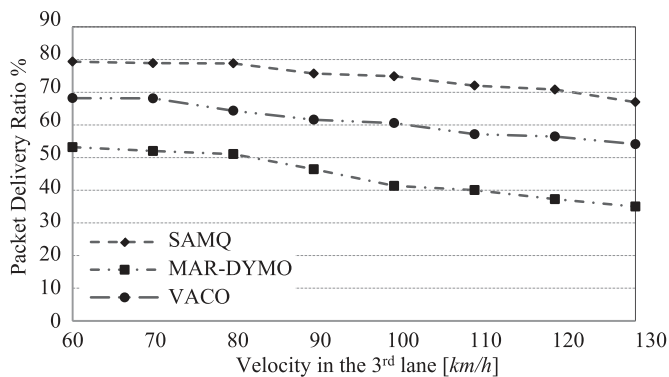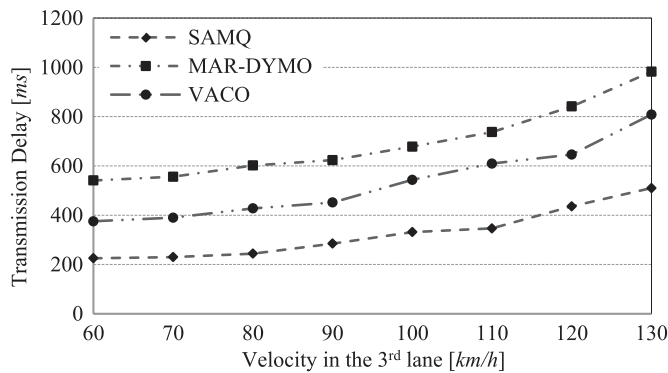
Fig. 8.   Simulation B—PDR.

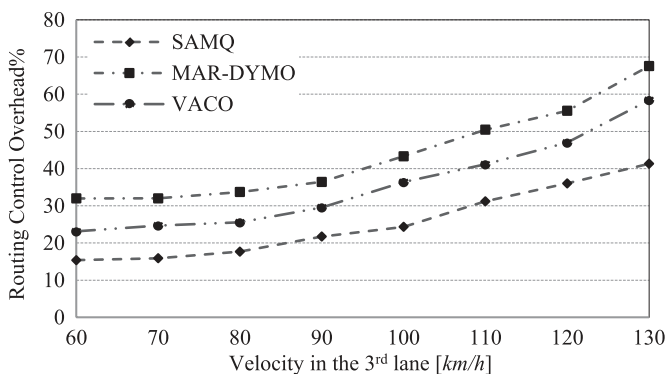Fig. 10.   Simulation B—Transmission delay.

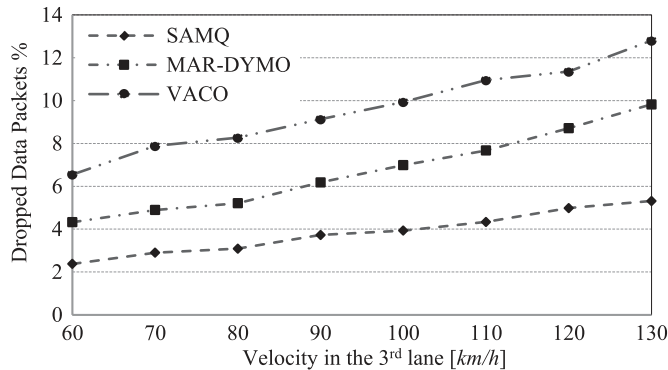Fig. 9.   Simulation B—Routing control overhead.

Fig. 11.   Simulation B—Dropped data packets.

defined QoS requirements. Apart from dropping data packets when the defined delay constraint is violated, losing one of a data packet's fragments results in the entire data packet being dropped. Thus, it is essential to ensure that all fragments are delivered within the QoS constraints, and SAMQ does this best among the simulated algorithms.

2) Simulation B—Effects of Dynamic Changes in Network Topology: The aim of this simulation is to investigate the impact of increased dynamics of the vehicular network on the routing algorithm performance when higher velocities are introduced.

It is shown in Fig. 8 that the average PDR reduces noticeably for all routing algorithms when the average velocity in the third lane starts to exceed 80 km/h. This reduction comes from the fact that the network topology becomes more dynamic, and thus, links/routes are more vulnerable to disconnection. In this particular case, it is important to build reliable multipath QoS routes among the communicating vehicles and immediately respond to possible disruptions. The SAMQ routing algorithm achieves this objective by utilizing the SA resolution level in Fig. 1 to deal with link/route vulnerabilities immediately and proactively, if possible. This is why SAMQ achieves higher PDRs than VACO and MAR-DYMO with the increasing velocity of vehicles.

In Fig. 9, it can be observed that all the routing algorithms are affected by the dynamic changes in the network topology. However, SAMQ maintains the least routing control overhead among the examined routing algorithms in this figure. Higher

vehicular velocities increase the dynamics of the network topology and increase the probability of disconnection of established links/routes. Therefore, it is essential for a routing algorithm to be prepared to respond to such dynamic changes in the network and choose the most reliable routes between the communicating vehicles. Moreover, countermeasures should be prepared to respond as soon as possible to disconnections without soliciting the source node to issue a new route discovery process that consequently increases the routing overhead. The employment of the SA levels shown in Fig. 1 helps SAMQ endure these challenges and maintain a reasonable routing control overhead.

In Fig. 10, it is confirmed that SAMQ is faster in computing feasible routes than VACO and MAR-DYMO. In this simulation, it becomes of great importance to start data transmission as soon as the first feasible route is computed. As other routing algorithms wait until they have established the global optimal route, the possibility of this route becoming quickly infeasible increases, particularly when the network dynamics increase. Therefore, SAMQ excels in this situation with the shortest transmission delay shown in Fig. 10 and accomplishes the best performance evidenced by the PDRs in Fig. 8. Moreover, different senders benefit from the established QoS links/routes and implement data forwarding faster. This is possible since SAMQ stays aware of the status of the vehicular network and updates the established links/routes accordingly.

Finally, Fig. 11 shows the average ratio of dropped data packets for each routing algorithm examined in this simulation. It can be noticed that SAMQ has the lowest average number of

dropped data packets. The route discovery process in SAMQ considers the dynamics of the network topology by forwarding ants toward routes that are expected to have longer lifetimes, as explained in the state transition rule in (7) and (8). Therefore, the established links/routes will be more reliable and expected to deliver data packets on time according to the QoS constraints. This feature of SAMQ, which is supported by the highest PDR shown in Fig. 8, gives SAMQ the lowest dropped data packet ratio in Fig. 11.

VII. Conclusion

In this paper, we have investigated the multiconstrained QoS routing problem in VANETs. More specifically, we proposed the new SAMQ routing algorithm for VANETs based on the ACS algorithm mechanisms. SAMQ aims to select the best route over multiple computed feasible routes between the communicating vehicles subject to multiple QoS constraints, if such a route exists. Moreover, SAMQ benefits from a novel SA model that provides a framework for improving QoS routing reliability in VANETs. SAMQ stays aware of the current vehicular network situation and responds locally and proactively to link/route disconnections, whenever possible. The performance of the SAMQ routing algorithm is evaluated through extensive simulations of video data transmission on a highway and comparison of its performance with that of the VACO and MAR-DYMO routing algorithms. SAMQ shows promising results in terms of achieving a higher PDR and, consequently, guaranteeing more reliable routing of data packets. It has been evident that utilizing both the SA concept and ACS mechanisms in QoS routing algorithms for VANETs is very promising to achieve a continuous and stable data transmission.

References

[1] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *IEEE Netw. Mag.*, vol. 27, no. 5, pp. 48–54, Sep./Oct. 2013.

[2] K. Yang, S. Ou, H. Chen, and J. He, "A multihop peer-communication protocol with fairness guarantee for IEEE 802.16-based vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3358–3370, Nov. 2007.

[3] A. Vinel, "Performance aspects of vehicular ad-hoc networks: Current research and possible trends," presented at the GI/ITG-Workshop MMBnet, Hamburg, Germany, Sep. 2009.

[4] F. Bai, H. Krishnan, V. Sadekar, G. Holland, and T. ElBatt, "Towards characterizing and classifying communication-based automotive applications from a wireless networking perspective," in *Proc. IEEE AutoNet*, San Francisco, CA, USA, 2006, pp. 1–25.

[5] Z. Wang and J. Crowcroft, "Quality-of-service routing for supporting multimedia applications," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 7, pp. 1228–1234, Sep. 1996.

[6] D. S. Reeves and H. F. Salama, "Distributed algorithm for delay-constrained unicast routing," *IEEE/ACM Trans. Netw.*, vol. 8, no. 2, pp. 239–250, Apr. 2000.

[7] M. Curado and E. Monteiro, "A survey of QoS routing algorithms," in *Proc. ICIT*, Istanbul, Turkey, 2004, pp. 43–46.

[8] Y. Bejerano, Y. Breitbart, A. Orda, R. Rastogi, and A. Sprintson, "Algorithms for computing QoS paths with restoration," *IEEE/ACM Trans. Netw.*, vol. 13, no. 3, pp. 648–661, Jun. 2005.

[9] B. Zhang, J. Hao, and H. Mouftah, "Bidirectional multi-constrained routing algorithms," *IEEE Trans. Comput.*, vol. 63, no. 9, pp. 2174–2186, Sep. 2014.

[10] F. Kuipers, P. Van Mieghem, T. Korkmaz, and M. Krunz, "An overview of constraint-based path selection algorithms for QoS routing," *IEEE Commun. Mag.*, vol. 40, no. 12, pp. 50–55, Dec. 2000.

[11] Q. Ma and P. Steenkiste, "Routing traffic with quality-of-service guarantees in integrated services networks," presented at the 8th Int. Workshop Network Operating Systems Support Digital Audio Video, Cambridge, U.K., 1998.

[12] R. Hassin, "Approximation schemes for the restricted shortest path problem," *Math. Oper. Res.*, vol. 17, no. 1, pp. 36–42, Feb. 1992.

[13] A. Juttner, B. Szviatovski, I. Mecs, and Z. Rajko, "Lagrange relaxation based method for the QoS routing problem," in *Proc. IEEE INFOCOM*, Anchorage, AK, USA, 2001, vol. 2, pp. 859–868.

[14] P. Van Mieghem, H. D. Neve, and F. A. Kuipers, "Hop-by-hop quality of service routing," *Comput. Netw.*, vol. 37, no. 3/4, pp. 407–423, Nov. 2001.

[15] P. Van Mieghem and F. A. Kuipers, "Concepts of exact QoS routing algorithms," *IEEE/ACM Trans. Netw.*, vol. 12, no. 5, pp. 851–864, Oct. 2004.

[16] H. D. Neve and P. Van Mieghem, "TAMCRA: A tunable accuracy multiple constraints routing algorithm," *Comput. Commun.*, vol. 23, no. 7, pp. 667–679, Mar. 2000.

[17] J. M. Jaffe, "Algorithms for finding paths with multiple constraints," *Networks*, vol. 14, no. 1, pp. 95–116, Spring 1984.

[18] T. Korkmaz and M. Krunz, "Multi-constrained optimal path selection," in *Proc. IEEE INFOCOM*, Anchorage, AK, USA, 2001, vol. 2, pp. 834–843.

[19] L. Rosati, M. Berioli, and G. Reali, "On ant routing algorithms in ad hoc networks with critical connectivity," *Ad Hoc Netw.*, vol. 6, no. 6, pp. 827–859, Aug. 2008.

[20] M. Dorigo and G. D. Caro, "AntNet: Distributed stigmergetic control for communications networks," *Artif. Intell. Res.*, vol. 9, no. 1, pp. 317–365, Aug. 1998.

[21] C. Dominguez, M. Vidulich, E. Vogel, and G. McMillan, "Situation AWARENESS: Papers and annotated bibliography," Armstrong Lab., Hum. Syst. Centre, Wright Patterson AFB, OH, USA, AL/CF-TR-1994-0085, 1994, (Accessed: 04/01 2015). [Online]. Available: http://www.dtic.mil/dtic/tr/fulltext/u2/a284752.pdf

[22] C. Onwubiko and T. Owens, "Review of situational awareness for computer network defence," in *Situational Awareness in Computer Network Defense: Principles, Methods and Applications*. Hershey, PA, USA: IGI Global, 2012, pp. 1–9.

[23] B. Mcguinness and L. Foy, "A subjective measure of SA: The Crew Awareness Rating Scale (CARS)," in *Proc. 1st Hum. Perform., Situation Awareness Autom. Conf., User-Centered Des. New Millennium*, Savannah, GA, USA, 2000, pp. 286–291.

[24] J. B. Kenney, "Dedicated Short-Range Communications (DSRC) standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.

[25] A. Vinel, "3GPP LTE versus IEEE 802.11p/WAVE: Which technology is able to support cooperative vehicular safety applications?" *IEEE Wireless Commun. Lett.*, vol. 1, no. 2, pp. 125–128, Apr. 2012.

[26] Fortress ES820 Vehicle Mesh Point, General Dynamics C4 Systems, Scottsdale, AZ, USA (Accessed Jan. 20, 2014). [Online]. Available: http://www.gdc4s.com/es820

[27] S. J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *Proc. IEEE ICC*, Helsinki, Finland, 2001, pp. 3201–3205.

[28] A. Nasipuri, R. Castañeda, and S. R. Das, "Performance of multipath routing for on-demand protocols in mobile ad hoc networks," *ACM Mobile Netw. Appl.*, vol. 6, no. 4, pp. 339–349, Aug. 2001.

[29] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," in *Proc. ICNP*, Riverside, CA, USA, 2001, pp. 14–23.

[30] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi, "A framework for reliable routing in mobile ad hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, 2003, pp. 270–280.

[31] G. Yang, X. Zheng, C. Chen, and L. Huang, "Robust multi-path routing for VANET based on mobility prediction," *Inf. Technol.*, vol. 13, no. 11, pp. 1916–1919, Apr. 2014.

[32] Y. Ge, H. Li, D. Huang, and L. Wan, "Node-disjoint multipath routing using segment-by-segment way in VANET," in *Proc. ISCID*, Hangzhou, China, 2013, vol. 1, pp. 418–421.

[33] N. Meghanathan, "Stability and hop count of node-disjoint and link-disjoint multi-path routes in ad hoc networks," in *Proc. WiMOB*, White Plains, NY, USA, 2007, pp. 1–8.

[34] J. Yi, A. Adnane, S. David, and B. Parrein, "Multipath optimized link state routing for mobile ad hoc networks," *Ad Hoc Netw.*, vol. 9, no. 1, pp. 28–47, Jan. 2011.

[35] P. Jacquet *et al.*, "Optimized link state routing protocol for ad hoc networks," in *Proc. INMIC*, Lahore, Pakistan, 2001, pp. 62–68.

[36] S. Balaji, S. Sureshkumar, and G. Saravanan, "Cluster based ant colony optimization routing for vehicular ad hoc networks," *Int. J. Sci. Eng. Res.*, vol. 4, no. 6, pp. 26–30, Jun. 2013.

[37] H. Dong, X. Zhao, L. Qu, X. Chi, and X. Cui, "Multi-hop routing optimization method based on improved ant algorithm for vehicle to roadside network," *Bionic Eng.*, vol. 11, no. 3, pp. 490–496, Jul. 2014.

[38] H. Rana, P. Thulasiraman, and R. K. Thulasiram, "MAZACORNET: Mobility aware zone based ant colony optimization routing for VANET," in *Proc. CEC*, Cancun, Mexico, 2013, pp. 2948–2955.

[39] S. L. O. B. Correia, J. Celestino, and O. Cherkaoui, "Mobility-aware ant colony optimization routing for vehicular ad hoc networks," in *Proc. IEEE WCNC*, Cancun, Quintana Roo, 2011, pp. 1125–1130.

[40] G. Li and L. Boukhatem, "Adaptive vehicular routing protocol based on ant colony optimization," in *Proc. 10th ACM Int. Workshop Veh. Inter-Netw., Syst., Appl.*, Taipei, Taiwan, 2013, pp. 95–98.

[41] O. A. Wahab, H. Otrok, and A. Mourad, "VANET QoS-OLSR: QoS-based clustering protocol for vehicular ad hoc networks," *Comput. Commun.*, vol. 36, no. 13, pp. 1422–1435, Jul. 2013.

[42] G. Li, L. Boukhatem, and S. Martin, "An intersection-based QoS routing in vehicular ad hoc networks," *Mobile Netw. Appl.*, vol. 20, no. 2, pp. 268–284, Feb. 2015.

[43] G. Li and L. Boukhatem, "An intersection-based delay sensitive routing for VANETs using ACO algorithm," in *Proc. ICCCN*, Shanghai, China, 2014, pp. 1–8.

[44] S. Bitam and A. Mellouk, "QoS swarm bee routing protocol for vehicular ad hoc networks," in *Proc. IEEE ICC*, Kyoto, Jun. 2011, pp. 1–5.

[45] M. Killat and H. Hartenstein, "An empirical model for probability of packet reception in vehicular ad hoc networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2009, no. 1, Mar. 2009, Art. ID. 721 301.

[46] I. Chakeres and C. Perkins, Dynamic MANET On-demand (DYMO) Routing, (Accessed: 11/24 2013). [Online]. Available: http://tools.ietf.org/search/draft-ietf-manet-dymo-21

[47] M. H. Eiza, T. Owens, and Q. Ni, "Secure and robust multi-constrained QoS aware routing algorithm for VANETs," *IEEE Trans. Dependable Secure Comput.*, DOI: 10.1109/TDSC.2014.2382602.

[48] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization: Artificial ants as a computational intelligence technique," *Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.

[49] 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.

[50] OMNeT++ Community, OMNeT++ Network Simulator, (Accessed: 05/20 2011). [Online]. Available: http://www.omnetpp.org/

[51] M. H. Eiza, Q. Ni, T. Owens, and G. Min, "Investigation of routing reliability of vehicular ad hoc networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, p. 179, Jul. 2013.

[52] YouTube Help, System Requirements, 2014, (Accessed: 02/20 2014). [Online]. Available: https://support.google.com/youtube/answer/78358?hl=en

Thomas Owens (M'83) received the Ph.D. degree in electrical and electronic engineering from the University of Strathclyde, Glasgow, U.K., in 1986.

In 1987, he was appointed Lecturer with the Department of Electronic and Electrical Engineering, Brunel University London, which was eventually absorbed into the College of Engineering, Design and Physical Sciences, where he is currently is a Senior Lecturer of communications. He was the project coordinator of the IST FP5 project CONFLUENT, the IST FP6 Integrated Project INSTINCT, and the FP6 Specific Support Action PARTAKE. He is the project coordinator of FP7 Support Action CHOICE. He has published more than 100 papers.

Qiang Ni (SM'08) received the B.Sc., M.Sc., and Ph.D. degrees from Huazhong University of Science and Technology, Wuhan, China, all in engineering, in 1993, 1996, and 1999, respectively.

He is a Professor of communications and networking with the School of Computing and Communications, Lancaster University, Lancaster, U.K. Prior to that, he led the Intelligent Wireless Communication Networking Group at Brunel University London, London, U.K. He has published more than 100 papers in his areas of interest. His main research interests include wireless communications and networking.

Dr. Ni was an IEEE 802.11 Wireless Standard Working Group Voting Member and a contributor to the IEEE wireless standards.

Qi Shi received the Ph.D. degree in computing from Dalian University of Technology, Dalian, China, in 1989.

He is a Professor of computer security and the Director of the PROTECT Research Center with the Department of Computer Science, Faculty of Engineering and Technology, Liverpool John Moores University, Liverpool, U.K. He has published nearly 200 papers in his areas of interest and managed a number of research projects with funding from various sources, such as the Engineering and Physical Sciences Research Council and the European Union. His research is centered on network security and privacy, including system-of-systems security, intrusion and denial-of-service detection, cryptography, and sensor network and cloud security.

Mahmoud Hashem Eiza (M'15) received the M.Sc. and Ph.D. degrees in electronic and computer engineering from Brunel University London, London, U.K., in 2010 and 2015, respectively.

He is a Research Assistant with the Department of Computer Science, Faculty of Engineering and Technology, Liverpool John Moores University, Liverpool, U.K. His research mainly focuses on computer, communication, and network security, with specific interests in quality of service and wireless network security and privacy in vehicular networks, smart grids, cloud computing, and Internet of things.