Adaptive Message Routing with QoS Support in Vehicular Ad Hoc Networks

Hanan Saleet, Rami Langar[¶], Otman Basir, and Raouf Boutaba
University of Waterloo ; 200 University Ave. W., Waterloo, ON, Canada
(¶) UPMC - Paris Universitas ; LIP6, 104 av. du President Kennedy, 75016 Paris, France
E-Mail: hsaleet@uwaterloo.ca ; rami.langar@lip6.fr ; obasir@uwaterloo.ca ; rboutaba@uwaterloo.ca

***Abstract*— As progress in VANETs research continues, there is a persuasive need to support Quality of Service (QoS) routing in such networks. While greedy forwarding is used in many MANETs applications, it is found that it is not convenient for VANETs applications. In this paper, we investigate the important and difficult challenge of QoS routing in VANETs. First, we present an adaptive message routing protocol that uses up to date information about the local topology in order to find the route with minimum end-to-end delay while maintaining a threshold for the connectivity probability and hop count. Then, we propose a genetic algorithm to solve this. To do so, we formulate the QoS routing as a constrained optimization problem. We also derive analytical expressions for the delay as well as the connectivity probability of a route in a two-way street scenario. Numerical and simulation results show that our algorithm gives an optimal or near optimal solutions, which provides an interactive and effective design environment and enriches our protocol performance compared to GPCR.**

***Index Terms*— VANETS, adaptive message routing, QoS message routing, performance analysis.**

I. Introduction

Vehicular Ad-hoc Networks (VANETs) represent a rapidly emerging and challenging class of Mobile Ad Hoc Networks (MANETs), where vehicles (or Mobile Nodes, MNs) and road infrastructures are equipped with wireless devices [1] [2]. Accordingly, the vehicles are able to communicate with each other as well as interacting with the road infrastructure.

To route a message from a source to a destination, the source node should be able to locate the destination's current position after which the source begins to transmit messages to the destination. Therefore, the source node inquires a *location server* about current position of the destination node. This *location server* should have up to date information about the location of destinations in the network. Using this location service management, MNs are required to update their current location information to this *location server*.

Prior works [3] [4] have proved that adding comparatively inexpensive infrastructure at some locations in a city improve the VANET performance. As such, we propose in this study to use a local Road Side Unit (RSU) as a location server to store the current location information about the MNs in its vicinity. Given the fact that it is important to have up to date information about destinations' locations, MNs should be able to update their locations to the local RSU in a timely manner. This delay sensitive location updating process can be mitigated using QoS routing between the MNs and the RSU. Therefore for a given network, a QoS routing protocol that provides a high quality route under the mobility of vehicles is needed.

To this end, several routing protocols for VANETs have been presented. Specifically, GPSR [5], which is a position-based routing protocol, is proved to be well suited for highly dynamic environments such as inter-vehicle communication on highways. However, radio obstacles, as they are found in urban areas, will have a significant negative impact on the performance of this protocol [6]. GPCR [6] on the other hand, selects nodes at intersections as next hop to let messages be delivered along roads. However, this protocol does not consider characteristics of vehicles' movement and thus does not maintain stable routes (i.e., does not take into account the connectivity probability of the selected path). In addition, many information dissemination routing protocols have been proposed [7], [8], [9]. However, their target applications are mainly delay-tolerant data dissemination, which is inappropriate for routing the location updates messages.

To overcome these limitations, we first propose in this paper, an Adaptive Message Routing (AMR) protocol that uses up to date information about the local topology. Our aim is to find optimal or near-optimal routes that minimize the end-to-end delay while satisfying QoS constraints such as the connectivity probability and the hop count of the selected path. To achieve this, we formulate the QoS routing as a constrained optimization problem. We also derive analytical expressions for both the delay and connectivity probability in a two-way street scenario. Then we propose a genetic algorithm in order to solve our NP-complete optimization problem. Numerical and simulation results show that our algorithm gives an optimal or near optimal solutions, which provides an interactive and effective design environment and enriches our protocol performance compared to the well-known GPCR protocol [6].

The remainder of this paper is organized as follows. Section II describes our proposed AMR protocol. In section III, we present the analytical framework used to evaluate the QoS routing problem. Specifically, the delay and connectivity probability performance metrics are derived in a two-way street scenario. In section IV, we formulate the QoS routing problem as an optimization problem and present our proposed genetic algorithm to solve it. Numerical and simulation results are presented in section V. Finally, section VI contains our concluding remarks.

II. Adaptive Message Routing protocol

As stated earlier, we address in this paper the efficient QoS routing that ensures minimum end-to-end delay while maintaining a threshold for the connectivity probability and

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

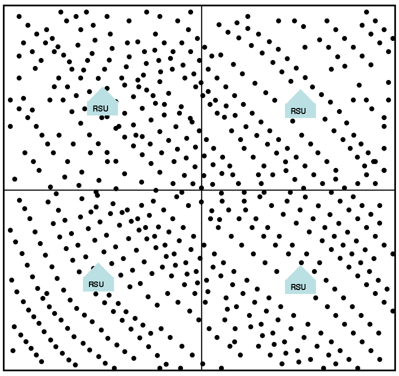


Fig. 1. Network structure

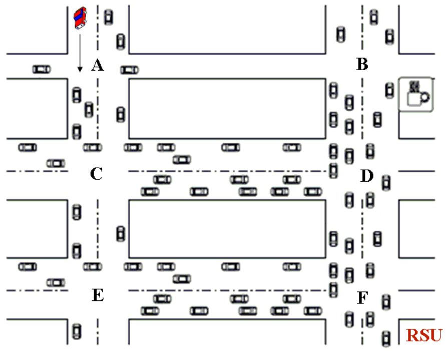


Fig. 2. Message routing in VANETs with AMR

the hop count through each selected path. To this end, we first propose an Adaptive Message Routing approach.

In this work, we model the network as a grid with fixed cell size across the area (see Fig. 1). Each MN is aware of the location of the grid origin (X_M, Y_M) (zero longitude and zero latitude). Also, each cell has an origin (X_c, Y_c) with respect to the grid origin. The origin of each cell gives the cell a unique identifier (ID) which identifies its location in regards to the grid origin. In the center of the cell there is a fixed infrastructure which is a Road Side Unit (RSU). This RSU is responsible of aggregating the location information about all MNs within the cell.

In AMR, the RSU located in the center of each cell acts as a *location server* (see Fig. 1). Therefore this RSU is responsible for saving current location information about all MNs that belong to that cell.Specifically, each RSU stores a detailed information about the MNs that it manages. This information contains the node ID, the transmission range T_r, $X - Y$ coordinates of the node location, time of the last update, and velocity and direction of the node's movement.

Each vehicle (i.e., MN) updates its location information to the RSU each time it moves one transmission range far from its previous location. This enables the local RSU to have a local view of the network composed by MNs that it manages. Doing so, a set of routes will be constructed between it and MNs. These routes are yet not stable due to nodes' mobility, but can use the same intermediate and adjacent street intersections towards the RSU (see Fig. 2). In this context, a road map can be modeled as a graph with a set of intersections and street segments. The successive intersections towards the RSU form the backbone routes. Thus, our aim is to determine the best intersections between the source node and RSU (i.e., the best backbone route connected by street segments) in order to forward location updating messages with minimum end-to-end delay while ensuring the above-mentioned QoS constraints.

It is worth noting that VANETs exhibits a bipolar behavior, where it may have a connected topology with high traffic volume or sparse topology where the traffic volume is low [10]. This implies a variation in the traffic patterns, which AMR tries to mitigate. As such, each RSU will construct the backbone routes every time there is a noticeable change in the statistical data received from the MNs in its vicinity (i.e., nodes' density, nodes' mobility, transmission range).

Specifically, when a vehicle (i.e., MN) enters for the first time into the cell managed by a RSU, it sends a signaling message to that RSU. In this case, the MN first inquires its neighboring vehicles about the optimal backbone route towards the RSU node. When the RSU receives the signaling message, it updates its location information base by adding the detailed information about the MN (i.e., transmission range, X-Y coordinates, time of last updates, and velocity and direction of the MN's movement). Then, it computes the optimal route among those available in the backbone. That is, the geographic position information of each selected intersection as well as the street ID to which the selected street segments belong are determined. Note that this is achieved while taking into consideration the QoS constraints such as end-to-end delay, connectivity probability and hop count. The result will be then forwarded to the MN as a response of the incoming signaling message, and will be used by the MN for its subsequent location updates.

To illustrate this, let us consider the example of Fig. 2. Assume that the red car, which enters the cell, moves southward. To send its location updates messages to the RSU, the vehicle can choose between three intersection paths: A-B-D-F, A-C-D-F, or A-C-E-F. The selected path will be given by the RSU since it is aware of all detailed information regarding the vehicles that it manages. Note that geographical routing is still used between two consecutive intersections (i.e., within one street segment). However, if there is a disconnectivity, the MN will use the carry and forward strategy. That is, the MN is allowed to store and carry messages while moving until it can forward the messages to the next MN. It is worth noting that AMR enhances the geographical routing, as we will see in section V, since this latter do not take into account the QoS constraints in the routing process (such as connectivity probability of the whole path, end-to-end delay, etc), and treats all segments in the network to be alike. This can often result in paths with high loss ratio and poor performance.

In the following, we present our analytical model used to derive the end-to-end delay and the connectivity probability.

III. Analytical Framework

In this section, we derive analytical expressions for the delay D_j as well as the connectivity probability $P_c^{(j)}$ of a street segment j in a two-way street scenario.

We model the road network as an undirected graph $G = (V, E)$ called street graph consisting of junctions (i.e., intersections of streets) $v \in V$ and street segments $e \in E$ connecting

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

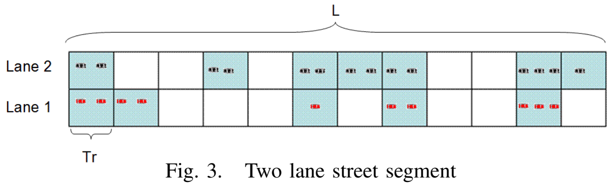


Fig. 3. Two lane street segment

these junctions. In order to implement AMR, we need to have a local view about the network topology (see Fig. 3). So that we can estimate up to date characteristics of each edge in the street graph G. These characteristics include: the length L of the street segment j, the average speed of nodes on the segment j (denoted by $\widehat{v}$) and the average spatial nodes' density (denoted by $\gamma_j^{(1)}$ and $\gamma_j^{(2)}$ for lane 1 and lane 2, respectively).

A. Delay D_j

The system model used in our analysis is shown in Fig. 3. We consider a two-way street scenario, where the vehicles are moving in two opposite directions. We divide the street segment into equal slots. Each slot corresponds to one transmission range T_r. That is, the two lane street is divided into slots according to the nodes' transmission range. In order to estimate the delay cost metric, we consider two cases.

Case 1

We may allow one vehicle to forward the message along the street segment. This case happens if the segment length L is less than one transmission range T_r. Let α be defined as $\alpha = \frac{L}{T_r}$. In this case, $\alpha \leq 1$. The delay of that segment will be t_p, where t_p is the time that the vehicle needs to recieve and transmit the message on that segment.

Case 2

This case occurs when the street segment length is larger than the transmission range (i.e., $\alpha \geq 1$), which is likely to be the case in real networks. In this context, we need more than one hop to forward the message along that edge.

Let $k1$ and $k2$ be random variables denoting the number of vehicles that is present in the interval of length T_r on lane 1 and lane 2, respectively (see Fig. 3). Assuming that the vehicles on both lanes are uniformly distributed with the node spatial density γ_1 for lane 1 and γ_2 for lane 2. It can be shown that $k1$ and $k2$ are Poisson distributed with the probability mass function (PMF) given as follows.

$$f(k_1) = \frac{(\gamma_1 T_r)^{k_1}}{k_1!} e^{-\gamma_1 T_r} \tag{1}$$

$$f(k_2) = \frac{(\gamma_2 T_r)^{k_2}}{k_2!} e^{-\gamma_2 T_r} \tag{2}$$

Let K be a random variable denoting the number of vehicles that is present in the interval of length T_r on both lanes, as shown in Fig. 3. Likewise, K follows a Poisson distribution with the following PMF:

$$f(K) = \frac{\left((\gamma_1 + \gamma_2) T_r\right)^K}{K!} e^{-(\gamma_1+\gamma_2) T_r} \tag{3}$$

In order to compute the delay on the street segment, we need to estimate the portion β of that segment that does not have any node to forward the message. In this case, the last node on that segment receiving the message is allowed to carry and forward the message along the street segment. The vehicle will not transmit the message until it comes within the transmission range of another vehicle. This portion (β) can be estimated as:

$$\beta = f(K=0) = e^{-(\gamma_1+\gamma_2) T_r} \tag{4}$$

In this case, the average delay can be computed using the average speed of nodes on the segment (i.e., $\widehat{v}$). Otherwise, the message will be forwarded hop by hop and the delay on each link will be equal to t_p as in the first case.

The average delay on the street segment j can be thus given as:

$$D_j = \begin{cases} t_p & \text{if } \alpha \leq 1 \\ \alpha(1-\beta) t_p + \beta \frac{L}{\widehat{v}} & \text{otherwise.} \end{cases} \tag{5}$$

Where $\widehat{v} = \dfrac{\sum_{k=1}^{N_j} v_k}{N_j}$ is the average speed of nodes on lane 1 of the street segment j, and N_j is the number of nodes on lane 1 of the street segment j.

B. Connectivity Probability $P_c^{(j)}$

In our work, location updates messages are relayed in the same direction of the vehicles' moving direction as opposed to [11]. To improve the connectivity probability, one may be able to take advantage of the opposing vehicles on a two-way street (see Fig. 3).

In this context, let us define a broken link between two consecutive vehicles V_i and V_{i+1} as the link with length $l = X_i > T_r$. This broken link will be fixable if there is vehicles on the opposite direction within the transmission range of each other and connecting V_i to V_{i+1}. This implies that the distance between any two consecutive vehicles of the new path on lane 2 must be smaller than the transmission range T_r.

Let's reuse k_2, which is a random variable denoting the number of vehicles on lane 2 that is present in the interval of length T_r. Under the assumption that the vehicles on this lane are uniformly distributed with the node spatial density γ_2 and using (2), the probability P_f that a broken link between two consecutive vehicles V_i and V_{i+1} is fixable can be thus given by:

$$\begin{aligned} P_f &= \prod_{k=1}^{\lfloor X_i/T_r \rfloor} (1 - f(k_2 = 0)) \\ &= (1 - e^{-\gamma_2 T_r})^{\lfloor X_i/T_r \rfloor} \end{aligned} \tag{6}$$

Note that the vehicles on lane 1 has a Poisson distribution and the distance X_i between V_i and V_{i+1} is exponentially distributed with parameter γ_1. To compute $P_c^{(j)}$, one should note that more than one broken link on lane 1 can occur. In this case, let Q be a random variable denoting the number of broken links on lane 1. The street segment will be considered as connected if all the Q links are fixable. Let $P_{c|Q}$ be the conditional connectivity probability given that there are Q broken links. $P_{c|Q}$ can be written as:

$$\begin{aligned} P_{c|Q}(q) &= \prod_{i=1}^{q} P_f \qquad \forall q = 0, 1, \ldots, N_j - 1 \\ &= (1 - e^{-\gamma_2 T_r})^{\sum_{i=1}^{q} \lfloor X_i/T_r \rfloor} \\ &= (1 - e^{-\gamma_2 T_r})^{(\alpha - \frac{(N_j - 1 - q)}{\gamma_1 T_r})} \end{aligned} \tag{7}$$

To obtain the connectivity probability of the segment j, we also need to know the PMF of Q (i.e., $P_Q(q), \forall q = 0, 1, \ldots, N_j - 1$). Recall that, a link is broken if the distance between any two consecutive vehicles is larger than T_r. Let P_b be the probability that a link q is broken. Since the distance between any two consecutive vehicles is exponentially distributed, it follows that

$$P_b = Pr\{X_i > T_r\} = e^{-\gamma_1 T_r} \quad (8)$$

Hence,

$$P_Q(q) = \binom{N_j - 1}{q} \times P_b^q \times (1 - P_b)^{(N_j - 1 - q)} \quad (9)$$

Finally, the connectivity probability of the street segment j can be expressed as:

$$P_c^{(j)} = \sum_{q=0}^{N_j - 1} P_{c|Q}(q) \times P_Q(q) \quad (10)$$

IV. Problem Formulation

In this section, we address the problem of finding the optimal route by minimizing the end-to-end delay while satisfying the connectivity probability and the hop count of each selected path. To achieve this, the performance metrics derived above will be used. The presented approach is first formulated as an optimization problem and then makes use of a genetic algorithm to solve it.

The objective function is as follows:

$$Min_{\ s_s, \ldots, s_d} \left(\sum_{j=1}^{L_{sd}} D_j \right) \quad (11)$$

subject to $P_c = \prod_{j=1}^{L_{sd}} P_c^{(j)} \geq P_{cth}$ and $H = \sum_{j=1}^{L_{sd}} H_j \leq H_{th}$, where L_{sd} is the number of street segments of the selected route, H_j denotes the hop count used to forward a message along the street segment j, P_{cth} and H_{th} are thresholds on the connectivity probability and hop count, respectively, s_s is the first intersection in the backbone route that is connected to the source node and s_d is the last intersection in the route that is linked to the RSU.

Consequently, we find that a metaheuristic technique, such as genetic algorithms [12] [13], is eminently appropriate in addressing such type of combinatorial nonlinear optimization problem. As such, we propose a genetic algorithm to solve this routing optimization problem. Fig. 4 displays the flow chart of our proposed algorithm which includes the following components: solution representation, initialization, evaluation, selection, crossover, mutation and termination.

A. Solution Representation and Initialization

Choosing an appropriate representation to encode the feasible solutions is the first step in applying genetic algorithms. This representation should be suitable for the fitness function and the genetic operations. In our approach, a natural encoding scheme would be to define each intersection in the backbone route as a gene. Then, the successive intersections in one route can be represented as a chromosome. Each feasible solution consists of one chromosome, denoted as $[S_s, S_1, S_2, ..., S_d]$. For example, routes A-B-D-F, A-C-D-F, or A-C-E-F in Fig. 2 are chromosomes. Therefore, an individual (or chromosome) is a vector containing the ordered intersections.

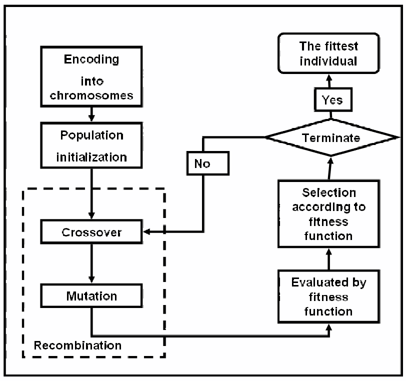


Fig. 4. Flow chart of the proposed genetic algorithm

Our search is conducted from a population of solutions. The initial population is generated by randomly selecting feasible solutions. For each MN in the network, solution or chromosome in the population is first constructed from a randomly selected intermediate intersection. Then, the process randomly chooses the next intersection in the backbone route. The process stops when the next intersection corresponds to the one adjacent to the RSU. It is important to ensure that the solution is feasible, i.e., each two consecutive intersections in the route are connected by a backbone link. A population of individuals can be constructed by continuing this process until generating a certain number of chromosomes independently.

B. Evaluation

A value for the fitness function $f(x)$ is assigned to each chromosome x depending on how it is close to solving the problem, and then the best individuals are selected depending on their fitness function. Since our objective is to minimize the end-to-end delay when delivering the update messages along the backbone route, we define $g(x) = \sum_{j=1}^{L_{sd}} D_j$. Consequently, in order to minimize $g(x)$, we define the fitness function $f(x)$ as the inverse of the delay value, i.e., $f(x) = 1/g(x)$.

C. Selection

During the selection operation, the quality of the population is improved by giving the high-quality solutions a better chance to produce offsprings that will be part of the next generation. In our implementation we use the roulette wheel selection strategy. Doing so, the chromosomes are selected based on a probability in proportion to its normalized fitness value, i.e., probability of choosing a chromosome corresponds to $\frac{f(x)}{\sum_{i=1}^{p_z} \left(f(x_i)/p_z \right)}$, where p_z is the population size.

D. Crossover

Crossover is usually executed with a probability θ. One possible crossover operation is the one point crossover where we select two chromosomes from the current population and

TABLE I

PARAMETER SETTINGS

Parameter	Value	Parameter	Value
t_s	1000 sec	T_r	250m
t_p	3 msec [14]	μ	0.3
P_{cth}	$0.5 \sim 0.8$ (default 0.5)	θ	0.8
H_{th}	$20 \sim 30$ (default 24)	N_g	20
v_k	50km/h	p_z	10

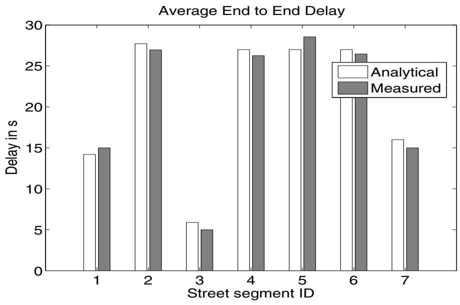


Fig. 5. Analytical vs. Simulation results of the delay for links on the backbone route

then find a common intermediate gene. That is, we find an intermediate intersection that is common to the two selected routes and then swap their second part, beyond the point of crossover to form two new offsprings. It is important that we check the new individuals to be sure that they are feasible.

E. Mutation

Mutation is an operator which causes random changes in the genes inside one chromosome. Therefore, mutation causes diversion in the genes of the current population, which prevents the solution from being trapped in a local optimum. Mutation is performed on the current population with a rate μ. In our implementation, we use uniform mutation operator, i.e after choosing any individual from the population with equal probabilities, we randomly pick an intermediate gene (intersection) and then choose randomly the adjacent intersection. It is important to make sure that the new individual is a feasible solution too.

F. Termination

The termination criteria in Fig. 4 can be based on the total number of generations, maximum computing time, or an acceptable threshold of the standard deviation between solutions in one population, or an hybrid termination criteria among them. In our implementation, we use the maximum number of generations.

V. NUMERICAL AND SIMULATION RESULTS

In this section, we compare our proposal with respect to a benchmark routing protocol, GPCR [6], through both simulations and analytical approaches. To this end, we developed our own discrete-event simulator using Matlab.

The simulation environment consists of a number of vehicles N that are uniformly distributed on each street segment j according to the statistical values of $\gamma_j^{(1)}$ and $\gamma_j^{(2)}$ for lane 1 and lane 2 of the street segment, respectively. In our experiments, we considered different vehicles' density representing morning rush hours which have high density (i.e., dense network), the noon time which has intermediate density and the night

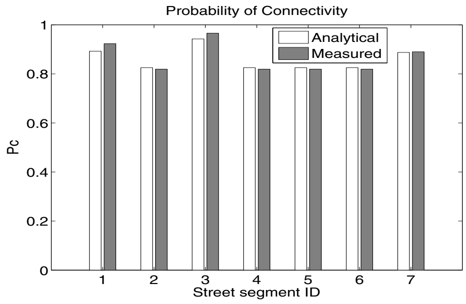


Fig. 6. Analytical vs. Simulation results of the connectivity probability for links on the backbone route

time which has low density (sparse network). To do so, we used different number of vehicles given that the area of the simulated network is fixed. Indeed, N is varied between 100 and 900 nodes. In addition, the mobility of nodes is modeled based on a given streets map. Specifically, if a vehicle reaches an intersection, it chooses the next street segment according to a probability which is proportional to that street segment density. Then, the vehicle adjusts its velocity to the mean speed allowed on the new street segment. The parameter settings in our experiments are listed in Table I, where t_s denotes the simulation time and N_g is the number of generations for our genetic algorithm.

To get an insight into our analytical expressions regarding the delay and the connectivity probability, let us consider Figs. 5 and 6. It can be observed that the analytical and simulation results are in good agreement, which demonstrates the accuracy of our analytical model.

Let us now focus on the performance comparison of AMR with that of GPCR. Fig. 7 depicts the end-to-end delay for both protocols as a function of N. As expected, AMR reduces the end-to-end delay especially when the network is sparse, i.e., when the nodes' density is low. Indeed, GPCR uses the shortest path between the vehicle and the RSU, without taking into consideration the connectivity degree of the selected path. As such, when the nodes' density is low, more vehicles are allowed to store and carry the packets, and then forward them to the next hop only when two vehicles become close to each other. In contrast, AMR uses routes with high connectivity degree. As such, the end-to-end delay is reduced. However, this may happen at the cost of increasing the number of hops as shown in Fig. 9. We can also observe from Fig. 7 that, as the number of vehicles increases, the performance of GPCR improves and becomes close to AMR. This is because vehicles tend more and more to forward the messages hop by hop without relying on the carry and forward strategy.

Fig. 8 depicts the connectivity probability of selected paths for the underlying protocols as a function of N. We can see that AMR builds routes with higher connectivity degree than that in GPCR. The performance gap between these two schemes is more significant in sparse networks, where longer paths (in terms of number of hops) are preferred. This is achieved in order to attain minimum end-to-end delay instead of using the carry and forward strategy, as explained above. In addition, we can observe that the connectivity probability increases with the number of vehicles since the MNs become close to each other and can thus use hop by hop forwarding. However, this happens

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

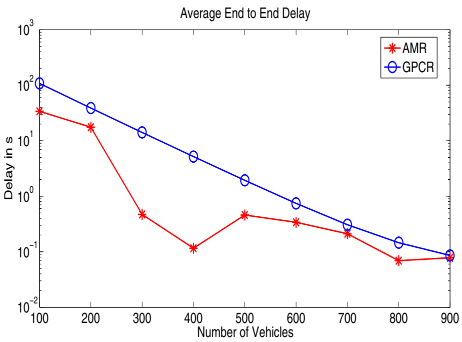


Fig. 7. End-to-end delay

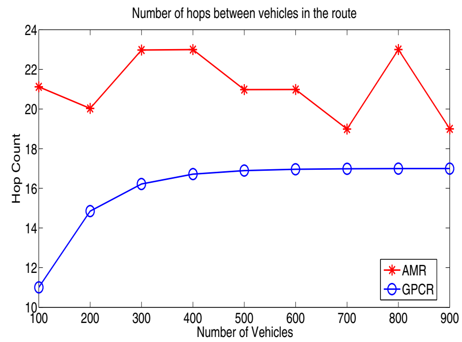


Fig. 9. Hop count

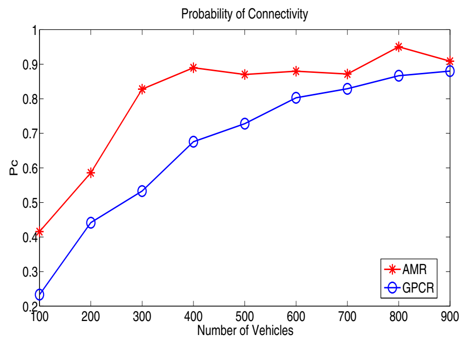
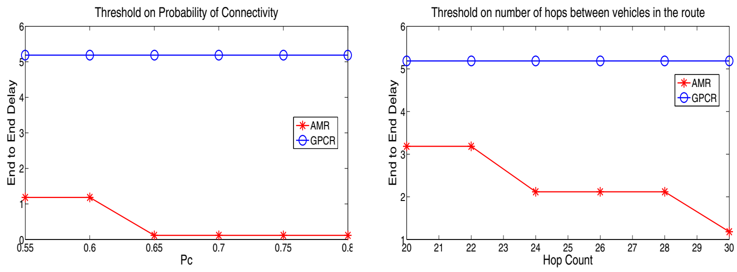


Fig. 8. Connectivity probability



(a) (b)

Fig. 10. Effect of P_{cth} and H_{th} on the end-to-end delay

at the cost of increasing the number of hops in the route as shown in Fig. 9. This latter figure shows that the hop count of routes generated by AMR is higher than that of GPCR, which supports the previous results.

Figs. 10(a) and 10(b) show the effect of the connectivity threshold (i.e., P_{cth}) and the hop count threshold (i.e., H_{th}) on the end-to-end delay, respectively. We can notice that the delay metric decreases as the threshold level increases. This is related to the fact that routes with more and more vehicles are allowed to be selected. This enforces the hop by hop forwarding and may result in a lower delay. Note that the variation of the threshold level (i.e., P_{cth} or H_{th}) does not affect the performance of GPCR, since it does not consider these parameters in the routing process.

VI. Conclusion

In this paper, we proposed a new approach for routing delay-sensitive location updates messages in vehicular ad hoc networks. Our proposal tends to minimize the end-to-end delay by using paths with high connectivity degree while maintaining a threshold on the hop count along each street segment. To achieve this, we formulated the QoS routing as a constrained optimization problem. We also derived analytical expressions for both the delay and connectivity probability in a two-way street scenario. Then, we proposed a genetic algorithm to solve our optimization problem. Using both analytical and simulation approaches, we compared our proposal with GPCR. We found that our protocol achieves substantial end-to-end delay reduction, especially in sparse networks. As such, our solution stands out as a promising candidate for large scale ad hoc networks such as VANETs.

Acknowledgements

The authors would like to thank Dr. Atef Abdrabou for his invaluable discussions.

References

[1] Y. Toor, P. Muhlethaler, A. Laouiti and A. Fortelle, “Vehicular Ad Hoc Networks: applications and related technical issues”, IEEE Communications Surveys and Tutorials, 3rd Quarter 2008.

[2] J. Zhao and G. Cao, “VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks”, IEEE Trans. Vehicular Technology, Vol. 57, No. 3, May 2008.

[3] H. Saleet, R. Langar, O. Basir, and R. Boutaba, “A distributed Approach for location lookup in Vehicular Ad Hoc Networks, In Proc. IEEE ICC 2009.

[4] C. Lochert, B. Scheuermann, et al., “Data Aggregation and Roadside Unit Placement for a VANET Traffic Information System”, In Proc. ACM VANET 2008.

[5] B. N. Karp and H. T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks”, In Proc. ACM MOBICOM 2000.

[6] C. Lochert, M. Mauve, H. Fusler, and H. Hartenstein. ”Geographic routing in city scenarios”, ACM SIGMOBILE Mobile Computing and Communications Review, 2005.

[7] L. Briesemeister, G. Hommel. ”Role-based multicast in highly mobile but sparsely connected ad hoc networks”, In Proc. ACM MOBIHOC 2000.

[8] H. Hartenstein, B. Bochow, A. Ebner, M. Lott, M. Radimirsch, and D. Vollmer. ”Position-aware ad hoc wireless networks for inter-vehicle communications: the Fleetnet project”, In Proc. ACM MOBICOM 2001.

[9] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter. ”MDDV: a mobility-centric data dissemination algorithm for vehicular networks”, In Proc. of ACM VANET 2004.

[10] N. Wisitpongphan, F. Bai, P. Mudalige and O. Tonguz, ” On the Routing Problem in Disconnected Vehicular Ad Hoc Networks”, In Proc. IEEE INFOCOM 2007.

[11] S. Panichpapiboon and W. Pattara-atikom, ”Connectivity Requirements for Self-Organizing Traffic Information Systems”, Vehicular Technology, IEEE Transactions, 2008.

[12] D. Goldberg, ”Genetic algorithms in search, optimization, and machine learning”, Addison-Wesley, 1989.

[13] J. M. G. Lopez, M. Imine and O. B. Madsen, “Network Planning Using GA For Regular Topologies”, In Proc. IEEE ICC 2008.

[14] S. Haykin and M. Moher, ”Modern wireless communication”, Prentice Hall, 2005.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.