UAV-Assisted Reactive Routing for Urban VANETs

Omar Sami Oubbati, Abderrahmane Lakas, Mesut Güneş, Fen Zhou, Mohamed Bachir Yagoubi

To cite this version:

Omar Sami Oubbati, Abderrahmane Lakas, Mesut Güneş, Fen Zhou, Mohamed Bachir Yagoubi. UAV-Assisted Reactive Routing for Urban VANETs. ACM Symposium on Applied Computing, Apr 2017, Marrackech, Morocco. 10.1145/3019612.3019904 . hal-02314136

HAL Id: hal-02314136
https://hal.archives-ouvertes.fr/hal-02314136

UAV-Assisted Reactive Routing for Urban VANETs

Omar Oubbati, Abderrahmane Lakas, Mesut Güneş, Fen Zhou, Mohamed Bachir Yagoubi

To cite this version:

Omar Oubbati, Abderrahmane Lakas, Mesut Güneş, Fen Zhou, Mohamed Bachir Yagoubi. UAV-Assisted Reactive Routing for Urban VANETs. ACM Symposium on Applied Computing, Apr 2017, Marrackech, Morocco. hal-02314136

HAL Id: hal-02314136
https://hal.archives-ouvertes.fr/hal-02314136

Submitted on 11 Oct 2019

UAV-Assisted Reactive Routing for Urban VANETs

Omar Sami Oubbati
Laboratory of Computer
Science and Mathematics
University of Laghouat
Algeria
s.oubbati@lagh-univ.dz

Abderrahmane Lakas
College of Information
Technology
United Arab Emirates
University
P.O. Box 17551, Al Ain, UAE
alakas@uaeu.ac.ae

Mesut Güneş
Institute of Computer Science
University of Münster
Germany
mesut.guenes@uni-muenster.de

Fen Zhou
Computer Science Laboratory
of Avignon (LIA)
University of Avignon
P.O. Box 1228, Avignon,
France
fen.zhou@univ-avignon.fr

Mohamed Bachir Yagoubi
Laboratory of Computer
Science and Mathematics
University of Laghouat
Algeria
m.yagoubi@lagh-univ.dz

ABSTRACT

Vehicular ad hoc networks (VANETs) are characterized by frequent path failures due to the high mobility caused by the sudden changes of vehicles direction. The routing paths between two different vehicles should be established with this challenge in mind. It must be stable and well connected in order to guarantee a reliable and safe delivery of packets. The aim of this work is to present a new reactive routing technique providing effective and well-regulated communication paths. These discovered paths are created based on a robust flooding discovery process involving UAVs (Unmanned Aerial Vehicles) to ensure the connectivity when the network is sparsely connected. The evaluation of this technique is performed using NS-2 simulator and its performances are compared with on-demand protocols dedicated for VANET. Simulation results show clearly that our approach gives interesting outcomes ensuring a high delivery ratio with a minimum delay. This hybrid communication between the vehicles and UAVs is attractive to initiate more smart connected nodes in the near future.

CCS Concepts

•**Computer systems organization** → **Embedded systems;** *Redundancy;* Robotics; •**Networks** → Network reliability;

Keywords

VANETs, Routing, Discovery phase, Urban Environment, Connectivity, UAVs, Traffic Density Estimation.

1. INTRODUCTION

The theme of vehicular networking has gained the attention of many scientists and researchers from both industry and academia. The establishment of such network provides a lot of possibilities to develop many applications that can supply a safe and comfortable driving experience. In addition, it offers useful services and entertainment applications to the drivers and passengers respectively. For example, vehicles can exchange information with each other about the real-time traffic congestion or incidents on the road in order to avoid traffic jam and enhance the road capacity. Furthermore, these communications can also include helpful infotainments like the weather, restaurant locations, gas station, parking places. Reliable data forwarding is considered as a foundation to put on the field these aforementioned applications.

Data packet routing plays a basic role to support the performance success of vehicular networks. Numerous routing challenges need to be addressed in order to adapt the proposed solutions to the unique characteristics of VANETs, especially the movements of vehicles (various speeds and directions). Most of the reactive routing protocols for VANETs [1, 4, 6, 8–12] only indicate presence or absence of routing paths between two vehicles; they also use a recovery process when a link-breakage takes place. When there is a path failure, a significant delay is diagnosed in the initializing of a new path. In addition, most of them do not take into account whether the discovered path is dense with vehicles or not in order to increase the chances of a successful delivering data packet between a source and destination. When the network is sparsely connected (no existing path), these protocols cannot forward the data packets because their route maintenance process (new route discovery) fails to find a new path, consequently, the data packets cannot reach their final destination.

The aim of this work is to design a reactive routing protocol destined for city environments that is based on forwarding packets using the densest (connected) and stable path which includes the fastest path in term of delay. Improved techniques are used in the discovery process that aim to minimize the delay and the control traffic overhead. In the case where there are several discovered paths, a scoring technique is used to select the optimal path based on several criteria. The route maintenance process is ensured based on two steps: (i) another alternative path can be used in each path breakage without re-initiating the discovery process. When there is no routing path, (ii) Unmanned Aerial Vehicles (UAVs) are used to establish a connection between the two disconnected clusters to create an alternative path.

The rest of this paper is structured as follows. First of all, we present an overview on the works already done on reactive routing protocols for VANETs in Section II. In Section III, our proposed routing protocols will be described in details. The performance evaluation and the results analysis of our approach are presented in Section IV. Finally, Section V concludes the paper and summarizes some future perspectives.

2. RELATED WORK

During the last few years, different types of reactive routing contributions are proposed for VANETs. All these protocols are based on the broadcasting and the flooding of the entire network in order to discover always new routes. However, they all fail to deliver the data packets when there is no existing paths between a pair of source and destination. Another problem detected in this kind of routing protocols is that most of them do not consider the well-balanced density (*i.e.,* real distribution of vehicles) in the discovered paths which is an important parameter to ensure a reliable transmission of packets.

Road-Based using Vehicular Traffic (RBVT) [6] is based on two different routing protocols, the proactive protocol (RBVT-P) and the on-demand routing protocol well-known as reactive (RBVT-R). RBVT-R performs route discovery on-demand and reports back to the source based on the greedy forwarding using a route reply (RR) which includes in its header, the position of the destination and a list of traversed intersections. In the case when the destination received several route discovery (RD), this means that RBVT-R has to choose the path that has the smaller number of traversed junctions (shortest path to the source) among many discovered paths and then to send the RR through it. Once the source receives a RR, it starts sending data packet via the same path traversed by the RR. As drawback, RBVT-R selects the shortest path (minimum number of traversed intersections) back to the source without taking into account the vehicle density on the road segments which may cause a disconnection problem at any time.

MURU (Multi-hop Routing protocol for Urban VANET) [4] calculates a metric called Expected Disconnection Degree (EDD) which is a probability that a given path might be disconnected during a given time period. The lower of EDD, the better is the path. The EDD is estimated by combining the vehicle positions, velocities, and trajectories. Consequently, path along vehicles moving in similar speeds and directions are more stable and therefore more desirable. After calculating the shortest path to the destination, the source initiates the route discovery, at the same time the EDD is calculated permanently at each hop and stored in the route request (RREQ). When the destination receives a certain number of RREQ, it chooses the path with the smallest EDD. The principal drawback of this protocol is that it does not take into consideration the vehicles density which is an important factor to measure the connectivity and ensure an efficient data delivery.

The authors in [12] use a route discovery process for both finding the destination location and installing a robust data delivery path. At the end, several RREQ reach the destination indicating several routing paths. Then, the destination calculates a weight for each different path (a set of intersections) based on the vehicles density and the delay. The path which obtains the best weight will be selected to send the route reply (RREP) back to the source using a greedy forwarding technique. To deal with the mobility of the destination, an intermediate vehicle can use a trajectory prediction based on additional information (velocity and motion direction of the destination) included in data packets. The major disadvantage of this protocol is that does not calculate the real distribution of vehicles between two successive intersections on the selected path which may cause a path failure even if this path contains a large number of vehicles.

LCAD (Load Carry and Deliver Routing) [3], is among the first routing protocols involving unmanned aerial vehicles (UAVs). It is completely dedicated for Mobile Ad hoc Networks (MANets) allowing UAVs to assist nodes on the ground in the data delivery process enhancing the connectivity on sparsely connected network. LCAD uses the technique of Carry & Forward only with the UAVs in order to carry the packets to the destination by flying, so this type of protocol is called Disruption Tolerant Network (DTN) and it is used only in the sky. However, on the ground Ad hoc On Demand Distance Vector (AODV) [7] is used. The major drawback of this proposed protocol is that UAVs do not use GPS information and trajectory calculation during route discovery and data forwarding.

To deal with all the aforementioned drawbacks, we propose a new reactive routing approach for urban VANETs. It takes into account the real distribution of vehicles on the discovered paths based on scoring technique involving many criteria. Furthermore, UAVs may belong to the discovered paths, and it can also be used as forwarding node connecting disconnected clusters when the network is sparsely connected.

3. UAV-ASSISTED REACTIVE ROUTING FOR URBAN VANET

This section describes in details the different functionalities of the proposed reactive routing approach. The main idea behind this approach is to exploit the discovery phase to have an accurate vision about the traffic density in each discovered path. A multi-criteria score is given to each discovered path based on the real distribution of vehicles and the end-to-end delays.

In some situations, the selected path that was created between a source and destination cannot remain constant due to the high mobility of vehicles. This is the reason why we use an intelligent route maintenance process based on alternative paths discovered beforehand and stored in the source. UAVs can play a role of a routing assistant, both in the discovery process and recovery strategy.

3.1 Assumptions

Our approach consists of vehicles and UAVs equipped with GPS and map to obtain their location. Each node maintains and updates a table of neighbors. We assume that there is no energy constraint for both vehicles and UAVs because they can recharge their batteries power from their energy resources (e.g., vehicles energy resources, the solar energy). The UAVs are able to communicate with vehicles through wireless interfaces up to a large transmission range with each other, so they will not be affected by obstacles (buildings, etc.). In addition, we suppose that the network has a sufficient number of UAVs so that at each moment, at least one UAV hovers an area of four road segments. Every road segment is considered to be divided into small fixed zones. Figure 1 shows how the road segments are divided into fixed zone with a size $\approx$ 300m corresponding to the transmission of vehicles. A unique identifier (ID) is given to each zone.

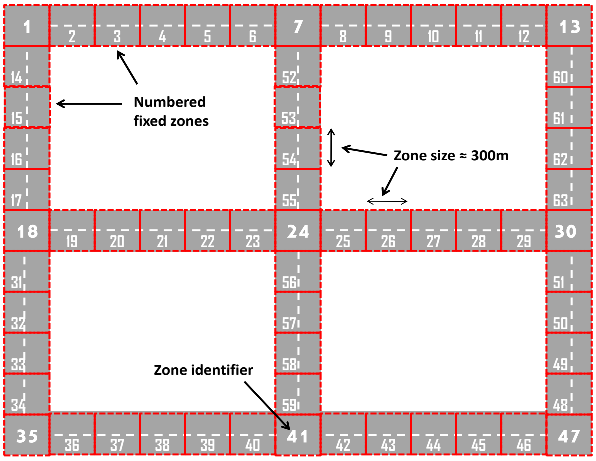

Figure 1: Network topology.

3.2 Path discovery

Our proposed approach establishes multiple paths on demand based essentially on "well-stable and connected" path. These paths, which are represented as a sequence of zone ID, are stored in the headers of data packets, are used by the intermediate nodes to send packets geographically between a source and destination. In addition, they are also used in the path maintenance process.

3.2.1 *The route request (RREQ) packet format*

Several fields compose the RREQ packet (*c.f.,* Figure 2). The $RREQ_{ID}$ field indentifies the initiated discovery path process to which the RREQ packet belongs. The Delay field defines the required time for the data packet to be delivered. $NB_{vehicles}$ field represents the exact number of vehicles that are in the discovered path until the target destination. The lifetime field determines the expiration time of the RREQ packet which is an important parameter limiting the flooding of the entire network. The fields of identifiers of the source vehicle and the target destination. Zones traversed field is a succession of transited zones by the RREQ packet until it reached the target destination.

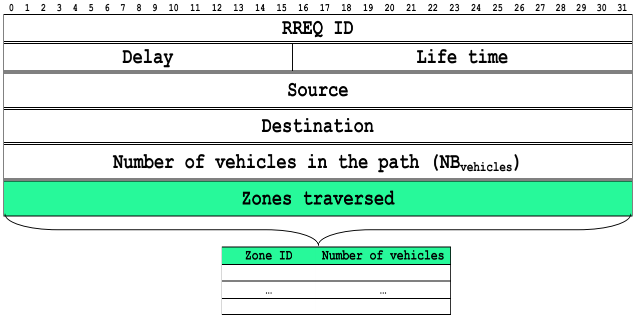

Figure 2: The route request (RREQ) format.

3.2.2 *The discovery process*

When a source vehicle wants to send a data packet to a target destination, it initiates a path discovery by flooding a RREQ packet to discover paths toward the destination. The flooding is necessary to get the location of the destination since the proposed protocol does not suppose using a location service. To minimize the impact of the broadcast storm, the $RREQ_{ID}$ field is checked when an intermediate node receives a RREQ packet. If a vehicle finds that the received RREQ has the same $RREQ_{ID}$ with a previously received one, it will be dropped. Otherwise, the $RREQ_{ID}$ of the received RREQ packet will be stored in the $List_{RREQ_{ID}}$ cached in this vehicle or UAV.

The paths are progressively built. Initially, the included path (zones traversed) is an empty list. When the RREQ packet is received by a node (vehicle or UAV) for the first time, it verifies if its zone ID location already exists in the zones traversed list or not. If so, only the $RREQ_{ID}$ is stored in this vehicle. If not, the $Zone_{ID}$ and the total number of vehicles in its location zone will be added to the enclosed zones traversed list in the RREQ packet and $NB_{v}ehicles$ will be updated. Then, the vehicle will re-broadcast the RREQ packet to all its neighbors.

We exemplify the discovery process based on Figure 3. The source vehicle generates a RREQ packet to find paths toward the destination. The source vehicle includes on it the $Zone_{ID}$ where it is located and the number of vehicles that exists within the zone based on its own table of neighbors. Then, the RREQ packet will be broadcasted. The same process is carried out by all intermediate nodes, except for UAVs in which they only add their own IDs in the zones traversed list.

Finally, when the first RREQ reaches the destination, it will start a timer to wait a certain time in order to have the knowledge about all the existing paths. The broadcast will be achieved by reaching all the RREQs the destination. The destination has three available paths to the source stored as a list of zones ID in each received RREQ.

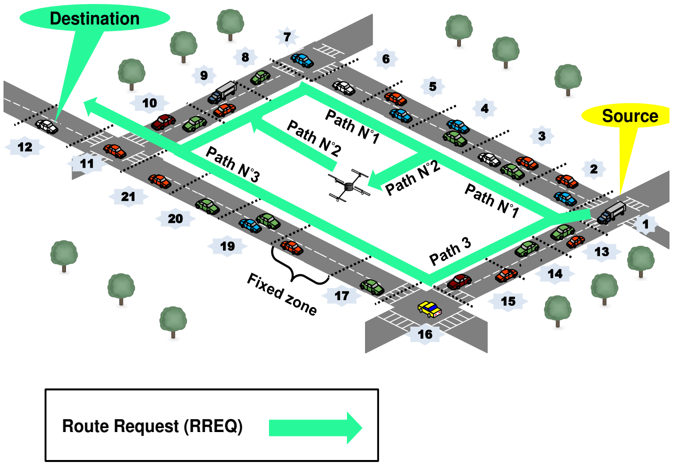

Figure 3: Path discovery process.

The RREQ packet will be handled by all nodes as shown in Algorithm 1:

Algorithm 1: RREQ handling

```
C ← The current vehicle;
S ← The source vehicle;
if (Source (RREQ) = C) OR
(RREQ_ID ∈ List_RREQ_ID (C)) then
    Drop (RREQ);
    // If I am the source or I already received
    this RREQ, I will drop it.
else
    if Destination (RREQ) = C then
        Goto path selection process;
        // If I am the final destination, I will
        make a routing decision.
    else
        // If I received this RREQ for the first
        time, I will record in it information about
        the density and geographic position, then I
        will rebroadcast it.
        Store ($RREQ_{ID}$, $List_{RREQ_{ID}}$);
        if $Zone_{ID}(C) \notin$ Zones traversed(RREQ) then
            Update ($Zone_{ID}$, Total number of vehicles on
            $Zone_{ID}$,$NB_{vehicles}$);
        Rebroadcast (RREQ);
```

3.3 Path selection

An appropriate path is selected from all discovered paths using the selection process. Indeed, several metrics are calculated for every discovered path based on the received information through the RREQs. As shown in Figure 3, the destination has the knowledge about three paths and accurate parameters about all of them (see TABLE 1). These information are exploited to know the suitable path.

Table 1: Discovered paths.

Path 1		Path 2		Path 3	
$NB_{vehicles}$=19		$NB_{vehicles}$=15		$NB_{vehicles}$=15	
Delay=1(s)		Delay=1.5(s)		Delay=4.5(s)	
Zone ID	Density	Zone ID	Density	Zone ID	Density
1	1	1	1	1	1
2	2	2	2	13	2
3	3	3	3	14	1
4	2	4	2	15	2
5	2	**UAV**		16	1
6	1	9	2	17	1
7	1	10	2	18	1
8	1	11	1	19	2
9	2	12	1	20	1
10	2			21	1
11	1			11	1
12	1			12	1

Based on the intercepted information as shown in Figure 1, two main metrics are calculated by the destination per each discovered $Path_i$. First, the average number of vehicles per each traversed zone in each path using the following equation:

$$Average = \frac{1}{N_z} \times NB_{vehicles} \tag{1}$$

Where N_z is the total number of traversed zones within a specific path. $NB_{vehicles}$ is the number of vehicles in the $Path_i$. Second, the standard deviation of zone densities based on the following equation:

$$S_{deviation} = \sqrt{\left(\frac{1}{N_z}\right) \times \left(\sum_{i=1}^{N_z} (Z_i - Average)^2\right)} \tag{2}$$

Z_i is the number of vehicles present at a specific $Zone_{ID}$. $S_{deviation}$ shows how the vehicles are balanced in a found path. Usually, a low $S_{deviation}$ indicates that the vehicles are not broadly dispersed around *Average*. However, a high $S_{deviation}$ denotes that the vehicles are more broadly dispersed.

A score is calculated for every discovered path by combining the intercepted parameters and the metrics calculated above based on the following equation:

$$Score = \left(\frac{NB_{vehicles}}{Delay}\right) \times \left(\frac{1}{(1 + S_{deviation} + HOPs_{UAV})}\right) \tag{3}$$

As we can observe, the score has a proportional relationship with $NB_{vehicles}$ within a specific path. However, it has an inverse relationship with $\left(\frac{1}{1+S_{deviation}+HOPs_{UAV}}\right)$ and the *Delay* where we penalize paths with a large *Delay* and

$S_{deviation}$ and paths which consist of UAVs. Consequently, paths with a low score are undiserable because they can be quickly broken due to the high mobility of vehicles and more particularly UAVs. Once a path that obtains the best score will be selected ($Path_1$ in Figure 3), a RREP packet will be generated and sent unicasting back to the source using the greedy forwarding technique along the zones succession of the selected path until it reaches the source (*c.f.,* Figure 4). It is important to mention that all discovered paths will be copied in the RREP packet in order to be used later in the path maintenance process.

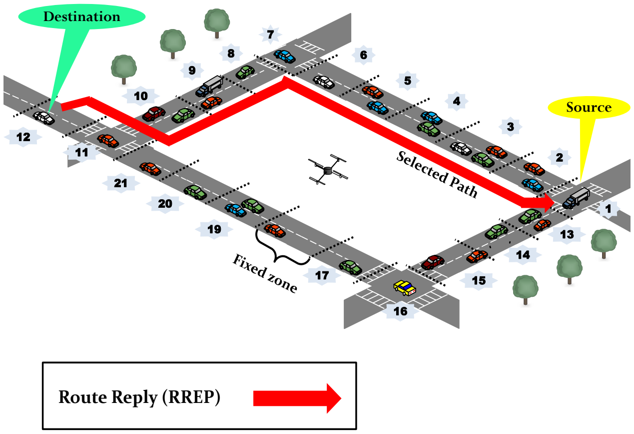

Figure 4: Path selection process.

The RREP packet will be handled by all nodes as shown in Algorithm 2:

Algorithm 2: RREP handling

```
C ← The current vehicle;
D ← The destination vehicle;
if Source (RREP) = C then
    Start Data packet delivering;
    // If I am the source of this RREP, I start sending the data packet.
else
    if D = C then
        // If I am the destination vehicle, all beforehand discovered paths will recorded in the RREP packet.
        Store (Discovered paths, RREP);
    Greedy Forwarding (RREP, $Selected_{path}$,Source);
    // If I am not the source of this RREP, I will use the greedy forwarding along the selected path toward the source.
```

3.3.1 The route reply (RREP) packet format

Two information are added by the destination to the RREP packet (*c.f.,* Figure 5): its geographic location and discovered paths are copied which will be used in the path maintenance.

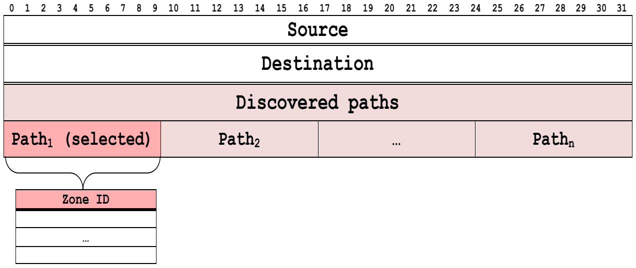

Figure 5: The route reply (RREP) format.

3.4 Path maintenance

Once the source receives the RREP packet, it will copy the discovered paths field into the header of the data packet and starts to send the data packet through the selected path. When this path disconnects, the first vehicle or UAV that detects this disconnection (*i.e.,* no neighbouring nodes in the next zone by checking the table of neighbours) checks available paths already recorded in the header of the data packet, and tries to find alternative path in all discovered paths. If there is an alternative path, it will be selected and the data packet is sent through the new link. Otherwise, a Route Error (RERR) message is sent back to the source which contains the details of the disconnection. Then, the source will reinitiate a new path discovery process.

For example, if the selected $path_1$ is disconnected, the intermediate vehicle located at the $Zone_{ID} = 4$ (*c.f.,* Figure 6) finds that the path is disconnected in its level and it cannot continue delivering the data packet via this path.

Table 2: Alternative path selection.

Path 1		Path 2	
$NB_{vehicles}$=19		$NB_{vehicles}$=8	
Delay=1(s)		Delay=1.5(s)	
Zone ID	Density	Zone ID	Density
1	1	1	1
2	2	2	2
3	3	3	3
4	2	4	2
sparse		**UAV**	
sparse		9	2
sparse		10	2
sparse		11	1
sparse		12	1

Therefore, it checks available paths in the header of data packet (see TABLE 2). After this verification, an alternative path is found through the UAV which will be selected to send the data packet.

According to the simulation, the UAVs can play the role of connection link between two disconnected clusters. In most cases, alternative paths used to deliver the data packets involve UAVs.

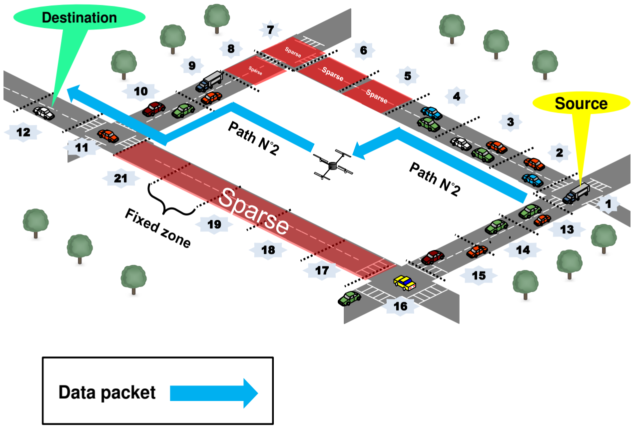

Figure 6: Path maintenance process.

4. PERFORMANCE EVALUATIONS

In this section, we present the evaluation of the proposed approach. NS-2 (Network Simulator 2) is used to perform the simulations in order to conduct series of experiments. Our approach is compared with reactive routing protocols for urban vehicular environment such as, RBVT-R [6] and AGP [12].

4.1 Simulation setup

The simulations are carried out in a city map size of 4×4 km^2 which consists of 9 intersections (*c.f.*, Figure 7). We generate the mobility of vehicles based on the created map using the VanetMobiSim mobility generator [2]. In addition, the MobiSim mobility generator [5] is used to generate a Random Walk mobility model for 16 UAVs hovering the network.

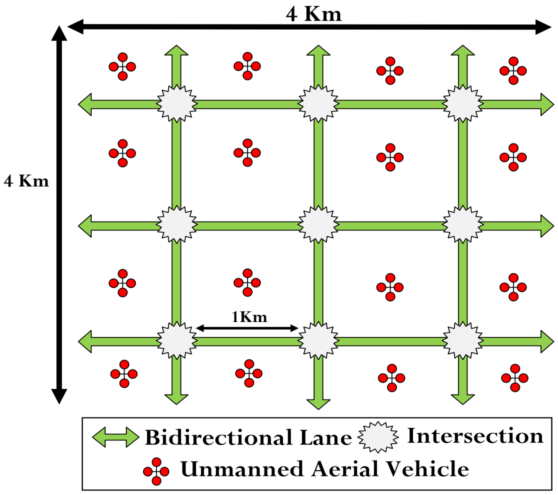

Figure 7: The simulation map.

The Table 3 summarizes the rest of parameters:

Table 3: Simulation parameters

Parameter	Value
Simulation area	4000m × 4000m
Number of UAVs	16
Communication range	250m
MAC Protocol	802.11 DCF
Frequency Band	5.15 GHz
Number of packets senders	35
Data packet size	1 KB
Number of vehicles	80-200
Vehicle speed	0-60 km/h
UAV speed	50-120 km/h
Simulation repeat times	15 times/scenario

4.2 Evaluation metric

Four performance metrics are studied in the evaluation:

1. **Packet Delivery Ratio:** The ratio of delivered data packets at destinations to the total number of packets sent by sources.
2. **Average Delay:** The average time taken by a successfully delivered data packets.
3. **The average number of hops:** The number of a successfully delivered data packets divided by the total number of hops.
4. **Overhead:** is the number of extra routing packets divided by the successfully delivered data packets at the destination.

4.3 Results Analysis

Figure 8 shows that for most cases, especially in high densities, our approach outperforms the other protocols. In general, we notice an increase in the PDR as the density of vehicles increases.

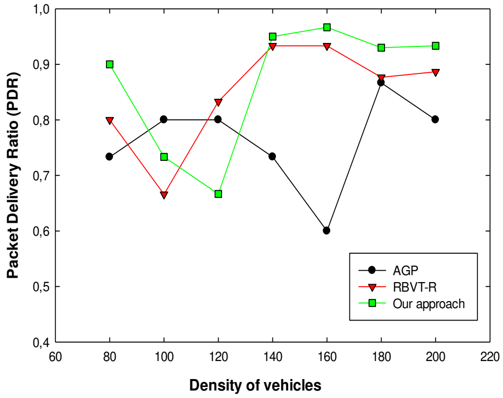

Figure 8: PDR vs. Density

The main cause is that our approach takes routing decisions based on the real distributions of vehicles in each discovered path which allow the senders to forward data packets efficiently with a minimum packet losses increasing the average delivery ratio. In addition, when there are path failures, we distinguish clearly the role of UAVs to enhance the connectivity in the network contributing also to this result. RBVT-R performs well than AGP in PDR. As RBVT-R uses a reliable recovery strategy based on a dynamic route updating technique when paths break due to the high mobility of vehicles which is not the case of AGP. RBVT-R demonstrates certain advantages in term of PDR than AGP.

In Figure 9 we can observe the performance in terms of average delay achieved by the evaluated protocols for different densities of vehicles. As we can see, in overall, our approach provides the lowest delay compared with RBVT-R and AGP, particularly when the network is weakly connected (fewer than 140 vehicles) The use of UAVs gurantees the connectivity and ensures the shortest path towards the target destination in certain situation which leads to the small delay comparing with the other protocols. However, in the high densities (more than 140 vehicles), RBVT-R outperforms our approach and AGP which results in the routes which remain active for longer periods of time thanks to the reliable route maintenance used by the source vehicle. Unlike RBVT-R, the routes composed of UAVs in our approach are constantly not stable due to high mobility thus causing the triggering of the path discovery process in each path failure. AGP achieves the high delay because it needs more time to discover routing paths and does not have a recovery strategy.

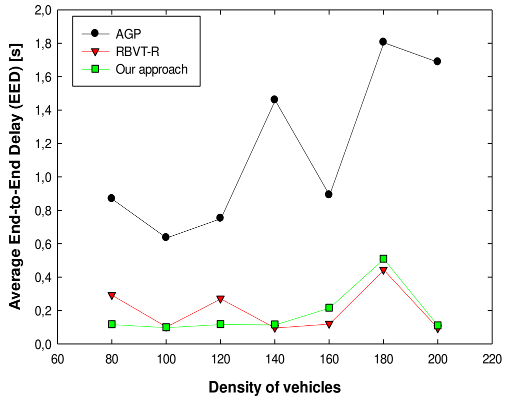

Figure 9: EED vs. Density

The average number of hops of data packets delivered along the selected paths from their source to their target destinations is depicted in Figure 10. We can clearly see that data packets in our approach need fewer hops to reach target destinations than RBVT-R and AGP. The main reason is that paths in our approach are often composed of UAVs in both delivering process and when there are path failures thus limiting the transited distance and consequently minimizing the average number of hops. In addition, our approach uses the greedy forwarding technique towards destination since it does not assume a location service which also decreases considerably the number of hops. However, RBVT-R achieves better average of hops compared with AGP, this is because data packets are geographically forwarded along the paths that have a smaller number of intersections towards target destinations which is not the case of AGP. AGP selects paths with high density of vehicles independently of the number of intersections.

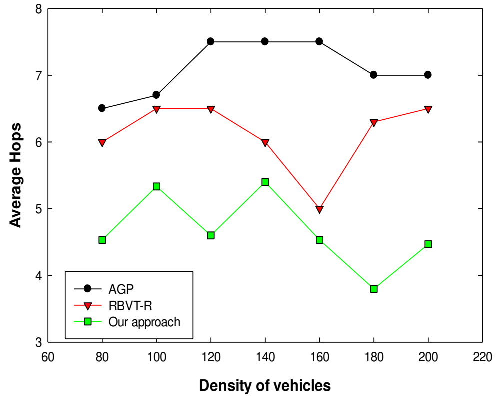

Figure 10: Avg(Hops) vs. Density

As shown in Figure 11, overall, RBVT-R generates less overhead packets in high density because it does not generate frequent route error (RERR) packets since discovered paths have long lifetime. In low density, AGP performs better than RBVT-R and our approach, because it uses a mobility prediction technique to deal with the frequent topology changes of the network. In addition, AGP does not use RERR packets when there are disconnections in paths and using maps and the real time traffic of vehicles which lead to lower overhead. However, for high density, AGP generates high number of overhead packets caused by the Hello packets.

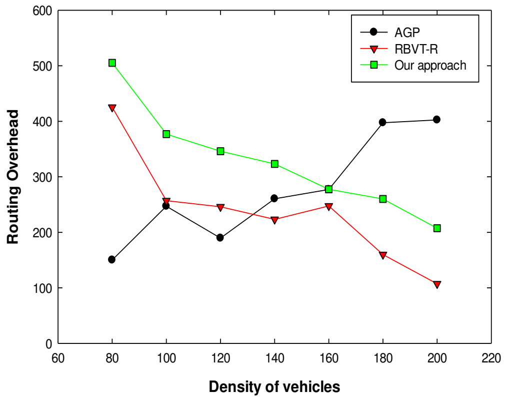

Figure 11: Overhead vs. Density

Our approach generates more overhead packets in low densities but reduced progressively with the increase in density of vehicles. This is because our approach generates more RERR packets in low densities because there are no alternative paths. As the number of vehicles increases, our approach always finds alternative paths to recover the selected paths decreasing the number of packets overhead.

5. CONCLUSION

This paper has introduced a reactive routing technique dedicated for urban VANET that takes into account the stability and the real distribution of vehicles in the path selection process. In addition, UAVs can be involved to both ensure better connectivity in sparsely connected networks and maintain the paths when the link failures occur. Simulation results show that our approach outperforms existing routing in terms of delivery ratio and average number of hops for high densities. It is believed that our approach should be able to provide good performances in terms of delivery ratio and average delay in both highway and rural environments. As future work we plan to deal with the mobility of UAVs in order to improve an efficient cooperation with vehicles on the ground. In addition, we plan to enhance this routing protocol by dividing it into two heterogeneous routing components. The first one is executed in the sky exclusively with UAVs and the second one is executed on the ground with vehicles. Furthermore, our recovery strategy will be improved to support the high mobility of UAVs.

Acknowledgment

This research was partially supported by grant # 31R012 from the UAEU Roadway, Transportation and Traffic Safety Research Center.

6. REFERENCES

[1] M. H. Eiza and Q. Ni. An evolving graph-based reliable routing scheme for VANETs. *IEEE Transactions on Vehicular Technology*, 62(4):1493–1504, 2013.

[2] J. Härri, F. Filali, C. Bonnet, and M. Fiore. VanetMobiSim: generating realistic mobility patterns for vanets. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, pages 96–97. ACM, 2006.

[3] M. Le, J.-S. Park, and M. Gerla. UAV assisted disruption tolerant routing. In *Proceedings of IEEE Military Communications Conference (MILCOM)*, pages 1–5, 2006.

[4] Z. Mo, H. Zhu, K. Makki, and N. Pissinou. MURU: A multi-hop routing protocol for urban vehicular ad hoc networks. In *Proceedings of the 3rd Annual IEEE International Conference on Mobile and Ubiquitous Systems: Networking & Services*, pages 1–8, 2006.

[5] S. M. Mousavi, H. R. Rabiee, M. Moshref, and A. Dabirmoghaddam. Mobisim: A framework for simulation of mobility models in mobile ad-hoc networks. In *Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMOB)*, pages 82–82, 2007.

[6] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea. VANET routing on city roads using real-time vehicular traffic information. *IEEE Transactions on Vehicular Technology,*, 58(7):3609–3626, 2009.

[7] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.

[8] C. Sommer and F. Dressler. The DYMO routing protocol in vanet scenarios. In *Proceedings of the 66th IEEE Vehicular Technology Conference (VTC-2007 Fall)*, pages 16–20, 2007.

[9] W. Sun, H. Yamaguchi, K. Yukimasa, and S. Kusumoto. Gvgrid: A qos routing protocol for vehicular ad hoc networks. In *Proceedings of the 14th IEEE International Workshop on Quality of Service (IWQoS)*, pages 130–139, 2006.

[10] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto. A stable routing protocol to support its services in vanet networks. *IEEE Transactions on Vehicular Technology*, 56(6):3337–3347, 2007.

[11] C. Tee and A. Lee. Adaptive reactive routing for VANET in city environments. In *Proceedings of the 10th IEEE International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN)*, pages 610–614, 2009.

[12] S. Yan, X.-y. JIN, and S.-z. CHEN. AGP: an anchor-geography based routing protocol with mobility prediction for vanet in city scenarios. *The Journal of China Universities of Posts and Telecommunications*, 18:112–117, 2011.