



HAL
open science

Adaptation of Topology-Based Routing Protocols for Data Gathering Applications in VANETs

Diego Pacheco-Paramo, Luigi Iannone, Daniel Kofman, Aline Carneiro Viana

► **To cite this version:**

Diego Pacheco-Paramo, Luigi Iannone, Daniel Kofman, Aline Carneiro Viana. Adaptation of Topology-Based Routing Protocols for Data Gathering Applications in VANETs. IEEE Vehicular Networking Conference (VNC), Dec 2015, Kyoto, Japan. hal-01465623

HAL Id: hal-01465623

<https://inria.hal.science/hal-01465623v1>

Submitted on 12 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptation of Topology-Based Routing Protocols for Data Gathering Applications in VANETs

Diego Pacheco-Paramo^{*†}, Luigi Iannone^{*†}, Daniel Kofman^{*†}, Aline C. Viana^{*‡},

^{*} LINCS,[†] Telecom ParisTech,[‡] INRIA

Abstract—The use of VANETs for data gathering applications imply that vehicles are not only able to generate data but also to forward it from others towards an access point (AP). Given the mobility characteristics of vehicles and the difficult propagation conditions of urban scenarios, it is commonly stated that topology-based routing solutions are less efficient than geographic-based routing solutions. However, these statements refer to generic vehicular communications, ignoring specific characteristics of data gathering scenarios. In this work, two existing routing protocols that prior to forwarding data, set a path from each vehicle towards the AP are adapted for data gathering applications. While the existence of a path reduces the number of ineffective retransmissions and the delay, it also increases the overhead, and therefore the presented protocols were adapted to improve the overall performance. Aiming to correctly evaluate the routing protocols, simulations are launched using a realistic trace from the city of Cologne. The results obtained show that it is possible to limit the overhead associated to topology updates, outperforming geographic-based solutions.

I. INTRODUCTION

Data gathering applications in vehicular networks allows a central node to maintain recent information from several specific locations for storage and/or processing purposes. Typical applications that have received attention in the past are urban monitoring for safety or maintenance uses, or to support intelligent transport systems [1],[2],[3]. These applications have unique requirements in terms of data size, sampling frequency or maximum delay, and normally communicate to an Access Point (AP) through multi-hop communications using the free spectrum, especially since the standardization of 802.11p. Sending periodical data from several vehicles towards the same geographical point implies a different type of solution than those proposed for V2V communications applied to safety or infotainment.

In the same way that occurs with wireless sensor networks (WSN), data gathering applications suffer from high interference as a result of having several vehicles transmitting towards the same geographical point. However, the solutions found for WSNs cannot be directly applied to VANETs due to its specific mobility characteristics and urban conditions. In VANETs the most important constraint is the high intermittent connectivity which typically demands high overhead costs for topology-oriented routing protocols. This is the result of a high vehicular mobility, a non-homogeneous distribution that may create gaps, and also of losses caused by physical objects such as trees or buildings.

Corresponding author: diegofelipe.pacheco@gmail.com

Based on previous efforts in WSNs, a large portion of the routing protocols use geographic information such as the position and velocity in order to forward data towards the AP. Among them, The greedy perimeter stateless routing protocol (GPSR) [4] is characterized by forwarding data in the direction of the AP, that is, in a greedy way. The main advantage of this type of protocols is that vehicles do not need to maintain an structure of the network, and therefore no overhead is created to update its own state or that of the others. However, due to the unknown network conditions, it is very likely that packets run into a local optimum, especially when the vehicular density is not high or there are many obstacles, which is common in urban scenarios.

Yet, keeping a path towards the destination is a desirable feature since it avoids the local optimum problem and the delay growth associated to it. With this in mind, the connected aware routing protocol (CAR) [5] was proposed for VANETs. This hybrid protocol sets a path from origin to destination just like a topology oriented protocol, but forwards data greedily towards anchors defined by their position, like geographic oriented protocols. However, this protocol cannot be used as it is in data gathering applications due to the high overhead associated to setting end-to-end paths. Therefore, we have adapted a lighter version of the protocol. On the other side of the spectrum are topology-oriented protocols. In particular, our interest lies in RPL [6], which was originally designed for WSNs. As such, this protocol creates a tree that grows from the AP towards the vehicles, simplifying the path setting process. But as a WSNs protocol, some modifications have to be done in order to make it work in VANETs, in particular in its ability to identify and correct fast topology changes.

Therefore, the main objective of this work is to adapt these two protocols (one hybrid and one topology-oriented) for data gathering applications in VANETs, hence exploiting their ability to have ready paths towards the AP while aiming to reduce their associated overhead, and evaluate their relevance in such scenarios.

A necessary condition for an appropriate evaluation of data gathering applications in VANETs is a correct mobility model. Most simulations use mobility models that underestimate the impact that vehicular mobility and urban scenarios have on the performance. Instead, we use the available dataset of the city of Cologne, Germany [7], which represents 2 hours of traffic (6 am-8 am). This dataset has a series of characteristics that set it apart from other traces, and add value to our

analysis. First, it is not limited to a specific type of vehicles, such as taxis or buses. Second, it is designed to cover an area of approximately 400 km² in a period of 24 hours, and therefore although our scope is limited to an area of 4 km², the behaviour of the vehicles is not. Third, the dataset has been modified to respect microscopic characteristics such as the road layer and the traffic signs. Finally, the trace is based in daily activity reports that considered 1.2 million trips in a 24 hours period, and some algorithms were applied to make it consistent from a macroscopic point of view. This last characteristic is important since it shows that previous efforts may have been too optimistic in terms of connectivity, and by extension in crucial parameters such as delay, path length or path duration[8].

The rest of the document is organized as follows. In section II we present related work. Two routing protocols that maintain an optimal path to the destination are described with their modifications in order to be used in data gathering scenarios in Section III. In section IV, we describe the experimental scenario and the tools used to evaluate the different protocols. In section V, we evaluate the routing protocols with varying conditions in terms of overhead, delay and PDR. Finally, the conclusions of the work are given in Section VI.

II. RELATED WORK

In this section we review some of the relevant work that exists in the area of data gathering and related routing protocols.

In [1], a data gathering system that aims to provide urban monitoring services is presented. Although it defines the similarities to traditional WSNs problems, it also brings into account the fact that traditional broadcasting is not possible due to mobility and paths conditions, and also that data filtering is not feasible since a vehicle cannot know which data will be useful for the application.

Most of the geographic routing protocols are based in GPSR [4]. As a result of the well known limitations of the perimeter mode in urban scenarios, different protocols have been proposed. The protocol GPCR [9] forwards data into junctions, where routing decision are made. If the decision in the junction leads to a local optimum, then the packet is sent back and a new decision is made. In GPSRJ+[10], another GPSR-based protocol, when a local optimum is found the nodes that have a neighbor that lies inside a junction have the ability to decide which route to follow, reducing the traffic in the junctions, and also reducing the number of hops.

In [11] the A-STAR protocol is proposed, where the path is set as a group of anchors and data is forwarded greedily between them. These anchors are street aware, that is, traffic information is considered for their definition. The same anchor principle is followed by CAR [5] as mentioned earlier. An improvement of this protocol, ACAR [12], considers the vehicular density in order to define the preferred path. With this tool, the protocol can react to sudden changes in traffic conditions. ECAR [13] defines secondary paths in order to avoid launching correction procedures that increase the overhead.

The MURU [14], is a topology-oriented protocol for VANETs that considers the disconnection probability is introduced, and this probability is calculated using information about the position and velocity of the nodes. However, unlike RPL, it uses reply messages to set a path, increasing the overhead.

None of the presented protocols fulfil the requirements of data gathering applications. In [15], a broadcasting method that searches simultaneously through several paths is proposed for an urban sensing scenario. This solution may avoid the delay associated to finding a local optimum and launching a perimeter mode of GPSR, but increases the overhead, reducing the PDR. In [16], a routing protocol that uses velocity vectors in order to decide where to forward data is presented. However, it considers that the density of vehicles is very high (high connectivity) while at the same time only a fraction of them can send data towards the AP. In [17] a protocol for delay-tolerant networks is proposed. This solution assumes that the solution allows data aggregation for a better use of the resources, and also its delay can be sometimes prohibitive (>1 min).

III. ADAPTATION OF A HYBRID AND A TOPOLOGY-ORIENTED ROUTING PROTOCOL.

In this section we describe the modifications and implementation of the two path setting routing protocols (RPL and CAR) evaluated in this paper.

A. RPL Implementation

The Routing Protocol for Low Power and Lossy networks (RPL), was originally designed for wireless sensor networks, aiming to be effective with limited resources and high link unreliability. Accordingly, the protocol keeps secondary routes when possible, identifies degrading links and corrects paths, all while controls the associated overhead. Also, many important characteristics are implementation dependant.

In RPL, the root node builds a hierarchical topology, defined as a Destination Oriented Directed Acyclic Graph (DODAG), that specifies a route from each node to itself using one type of message, the DODAG Information Object (DIO). Once a node receives a DIO, it can first, calculate its own rank, then, choose a set of parent nodes and finally, send a new DIO to inform other neighbors. The rank is an integer that grows linearly from the root node, and identifies the position of the node in relation to the root and the rest of the nodes. The parents are a set of candidate nodes where data can be forwarded towards the root node. Accordingly, the parents must have a rank equal or lower than the node. Each node has a preferred parent, that is, a default path, but maintains a list of parents for resilience purposes. Hence, immediately after the reception of a DIO, a node has an optimal path towards the root node.

The optimal path is set according to an optimization function (OF) which indicates the criteria that the nodes must follow in order to choose their preferred parent, and thus affects their rank in the DODAG. In our implementation, the optimization function depends on two parameters: The

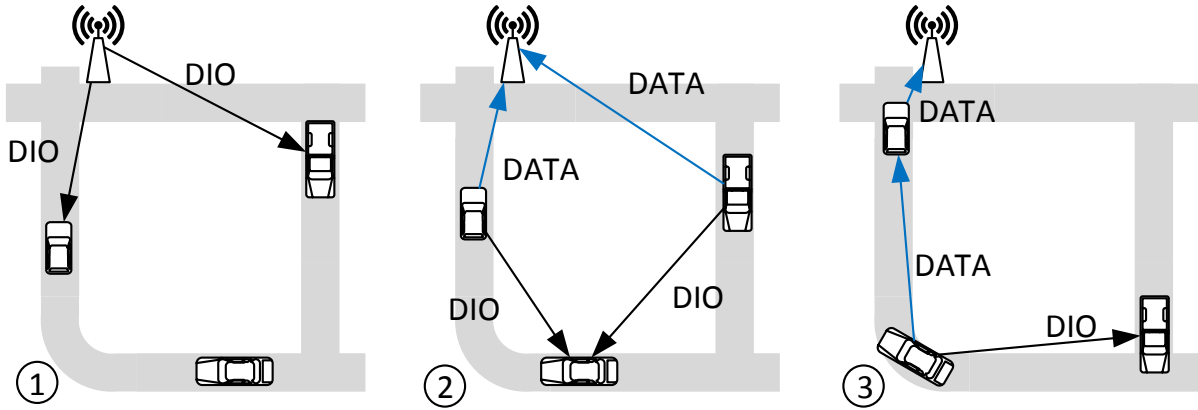


Fig. 1. Path setting and data forwarding to an AP using RPL. DAO messages are not used.

expected transmission count (ETX) [18], which is the number of necessary retransmissions to successfully perform a one-hop delivery, and the hop count as follows:

$$OF(Parent) = ETX_{Parent} + \frac{Hops_{Parent}}{max(Hops)_{Parents}}, \quad (1)$$

where ETX_{Parent} is calculated through the interaction of hello beacons with its neighbors, $Hops_{Parent}$ is the number of hops from the root to the Parent candidate and $max(Hops)_{Parents}$ is the maximum number of hops from the root to the candidate parents. Hence, by choosing the parent with the lowest OF, the protocol penalizes unreliable links and benefits shorter paths. If the link between a node and its preferred parent degrades, the node can easily choose another parent of its parent list, adding resilience to the network without increasing the overhead. The introduction of the link state in the OF is crucial for the correct performance of RPL in a VANET scenario.

To control the generation of DIO messages, the trickle algorithm [19] is used. With this algorithm, the node decides to forward DIO messages if it has received less than R_{DIO} redundant messages about the status of its neighbours in a given initial period of time T_{DIO} . If enough consistent DIO messages are received, then the node doubles the original period to account for consistent DIO messages. If the conditions are met, this process is repeated until a maximum period is reached. However if an inconsistency is detected (e.g. a neighbor changes its rank), a DIO message is sent, and the period is reset to T_{DIO} . In this work we keep this algorithm since it performs better than sending DIO messages with a fixed frequency [20]. However, due to the highly dynamic nature of VANETs, we have identified three inconsistencies on data forwarding: i) A node may receive data when it has no active parents, ii) A node may receive data from a node with lower rank, and iii) A node may receive data originated by itself. These inconsistencies are the result of an outdated

DODAG (i,ii) or connectivity issues (iii), and cause an increase of the overhead. The former can be enhanced reducing T_{DIO} , but the latter is caused by loops originated due to isolation from the DODAG, and therefore is mobility dependant. Our implementation can identify this loops by the rapid growth of rank, and force a disconnected state, which can be overcome when normal network condition arise.

The path setting process of our implementation is explained in Figure 1. First, an AP broadcasts a DIO message (rank=0) that two other vehicles receive. These vehicles can immediately forward data to the AP, and also broadcast a DIO message with the rank updated (rank=1). In 2, a third vehicle receives both DIOs and chooses one of them as the optimal path to the AP, considering both the rank, and the ETX. Finally in 3, one of the vehicles has lost connection to the AP, and therefore its rank, so when it receives a DIO message, it accepts it, and adopts a new rank (rank=3). With this protocol, the nodes have a path to the AP described by a set of nodes.

B. CAR Adaptation

The connectivity-aware routing protocol (CAR) [5] is a geographic routing protocol that establishes full paths before data transmission, updates them in order to adapt to dynamic conditions, and given the case, performs error correction procedures. However, it suffers from high overhead in data gathering scenarios, and some adaptations were performed to address this issue.

In order to set paths to the root node, CAR sends Path Discovery (PD) messages including information about its location and velocity vector in dissemination mode. Whenever a node receives a PD message, it compares its position and velocity vector with the ones in the message, and if there are substantial changes, it appends an anchor, that is, its own location and velocity vector to the message. Several PD messages are expected to arrive to the final destination, so it can decide which path is optimal. The destination creates a

Route Reply (RR) message, and sends it back to the origin following the optimal path so data transmission can begin. However, for data gathering applications the dissemination of PD messages from each origin node to the destination is detrimental to the overall performance. To address this issue, in our implementation PD messages travel until they find a node that has a working path to the destination node, when a unicast RR message will be sent to the origin. Let us recall that in a scenario where all the nodes transmit data, only one-hop PD and RR messages are necessary. Still, this solution is not enough to reduce the overhead. Once the network overcomes the transient state of disconnected nodes, after a PD broadcast most of the neighbors will reply with an RR message. Therefore, a probability p_{RR} is set, which defines the probability that a RR message is sent after receiving a PD message.

Data is sent to the root node through advanced greedy forwarding (AGF)[21]. With this method, data is forwarded to the node that is closer to the next anchor, that is, the path describes a set of anchor points instead of a set of nodes. In order to avoid unnecessary efforts to reach the exact location of each anchor, the protocol includes a threshold distance d_{min} . If the distance between the node and the next anchor is lower than d_{min} , then the node forwards towards the following anchor. The protocol suggests to use half of the maximum transmission distance on free-space air as d_{min} , but in dense urban scenarios this value should be smaller and adapted to the structure of the layout. This proves difficult, since the layout is not uniform, and transmission ranges depend on the conditions of the network.

Since nodes tend to distance from the anchors, *CAR* counts with methods to update the paths and under certain conditions, to correct them. As a source moves away from the location where it started the Path Discovery procedure, new anchors could be added to the path. An anchor is added if any of the following conditions is met: i) The node has moved away from the original position in an angle that is higher than 18° from the original velocity vector, or ii) The distance between the node and the first anchor of the path is higher than d_{max} . But this procedure alone does not lead to optimal performance, so in order to use the best path to the AP in our implementation the node demands the paths from its neighbors in a periodical basis T_{Upd} . Hence, if a neighbor has a route with less hops or with equal hops but newer, the node replaces its old path. In this way the network prioritizes shorter paths or at least those with active vehicular traffic. Finally, when the node is not able to forward data due to network disconnection it proceeds in two ways: First, it waits a given time before retransmitting hoping that the disconnection time is short and a next hop can be found soon. Then, if after the period expires a next hop is not found, a path discovery path procedure must be launched.

In Figure 2 we describe the path setting process of our adaptation of *CAR* or as it will be called from now on, *CAR_{ad}*. First, the vehicles send PD messages that are received by the AP, which replies with two RR messages. Then the vehicles are able to forward data to the AP, or to a node

that is closer to it, since the location and the velocity vector of the AP is known. Then in 2, a new vehicle broadcasts a PD message, and the two connected vehicles reply with RR messages. In 3, the vehicle sends data to the node closer to the next anchor point (the green point), and the procedure is repeated with disconnected vehicles. In this case, the path to the AP is described by a set of anchors (the green points in the Figure) that serve as references for data forwarding.

IV. EVALUATION SCENARIO

In this section, we describe the scenario for data collection and the simulation setup for the evaluation of the two described topology-oriented protocols: *RPL* and *CAR_{ad}*, and *GPSR* [4], a well-known geographic protocol which is used as a reference and available in the INET Framework[22].

We consider an AP placed in a concurred place of the city centre of Cologne, Germany, that collects information from the vehicles that lie inside an Area of Interest (AoI) formed by a circle with radius of 1 km. This is the fourth biggest city in Germany, and a realistic dataset [7] that represents 2 hours of traffic (6 am-8 am) is available. The simulation of the communication stack was done through OMNET++[23] in tandem with Veins [24] and SUMO [25] for the realistic mobility of vehicles. The physical and MAC layers follow 802.11p specifications, that is, 10 MHz channels in the 5.9 MHz band, with an alternating access of the control (CCH) and service (SCH) channels. Since 802.11p allows the use of different access categories (AC), we give the highest priority in the SCH to routing messages (DIO messages in RPL, Path Discovery and Route Reply messages in CAR), and the second highest to data. The CCH is used exclusively to send hello beacons. In order to reduce the known congestion issues of 802.11p in high density scenarios [26] we increased the minimum size of the contention window. Additionally, we used the two-ray ground model for path loss and accounted the buildings and constructions for shadowing. The parameters of the simulations are included in Table I. On each simulation, users transmit data for 300 seconds, and the simulation ran until no more data was buffered in the vehicles. Although data can only be generated inside the AoI, routing packets can be received (but not forwarded) by nodes outside so paths are available when requested. Data packets have a size of 1 KB and their generation period is uniformly distributed [0.5-2] s, that is, with mean 1.25 s, which are acceptable values for different applications [3]. It should be noted that vehicles that use RPL can only start generating data once they have a rank and are inside the AoI. When *CAR_{ad}* is used, vehicles can start forwarding data once an RR message is received and the vehicle lies inside the AoI. On the other hand, when GPSR is used, vehicles only need to be inside the AoI for data forwarding. It is known that MAC layer parameters have an important impact when evaluating routing protocols. In particular, the setting of the maximum number of retries before discarding a packet, $Retry_{MAX}$, is of crucial importance. Ideally, this parameter should adapt to the conditions of the network in order to transmit data without creating too much

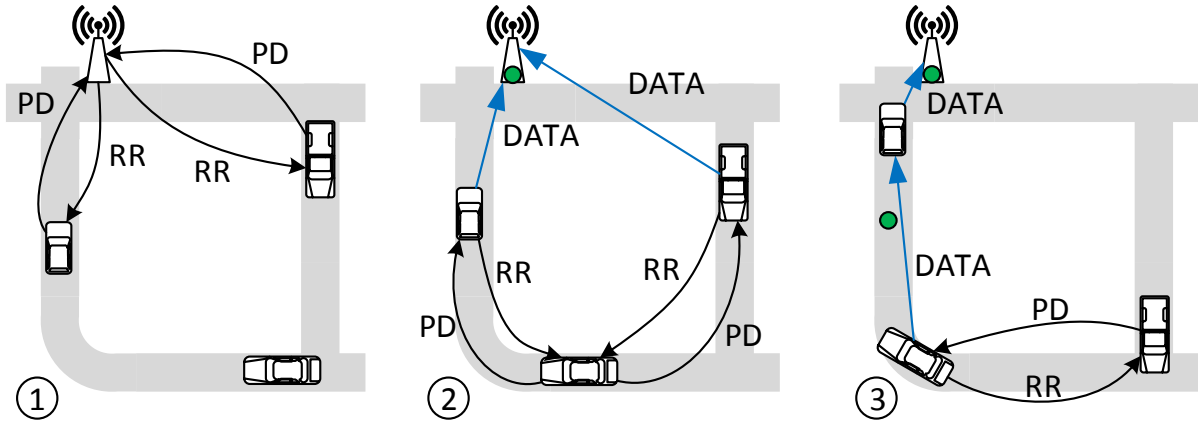


Fig. 2. Path setting and data forwarding to an AP using CAR_{ad} .

TABLE I
SIMULATIONS PARAMETERS

| MAC and PHY | RPL | CAR_{ad} |
|-------------------------------|---------------------------|---------------------------|
| $f=5.9\text{MHz}$ | $T_{DIO} = 0.1 \text{ s}$ | $T_{Upd} = 0.1 \text{ s}$ |
| $\alpha=2.0$ | $T_{doublings} = 5$ | $p_{RR} = 0.3$ |
| Bitrate = 18 Mbps | $R_{DIO} = 4$ | $d_{min} = 50 \text{ m}$ |
| Sensitivity = -89dBm | | $d_{max} = 200 \text{ m}$ |
| $P_{Tx} = 10 \text{ mW}$ | | |
| CW = 512 slots | | |
| $Retry_{MAC} = 50 \text{ ms}$ | | |
| $T_{HELLO} = 0.1 \text{ s}$ | | |

interference, but in this work we changed it manually on each scenario with the objective of maximizing the PDR.

V. RESULTS

In this section we evaluate the three protocols at different times from the Cologne trace.

In Figure 3 the packet delivery ratio (PDR) for the three protocols at different times, each with a different mean number of vehicles inside the AoI, N is shown. It can be seen that the PDR increases with N , but for high vehicular density it drops considerably. Also, the PDR is bounded by 90%, which is the result of different conditions such as high mobility, low connection times and interference. Among the protocols, RPL is able to obtain the highest PDR for most scenarios, in fact it is only surpassed for $N=114$. This is due to the fact that vehicles with RPL only forward data packets if they have a rank, that is if they have an up to date path to the AP. This, however, is not enough to maintain a high PDR for high values of N , and therefore we see a drop of around 20% when N grows from 114 to 139. In high interference scenarios, it is possible that vehicles do have a rank, but they are incapable of successfully forwarding data, even as $Retry_{MAX}$ grows. In fact, if this value is too high, more interference is created, and the PDR drops. This applies for all the tested protocols, but it

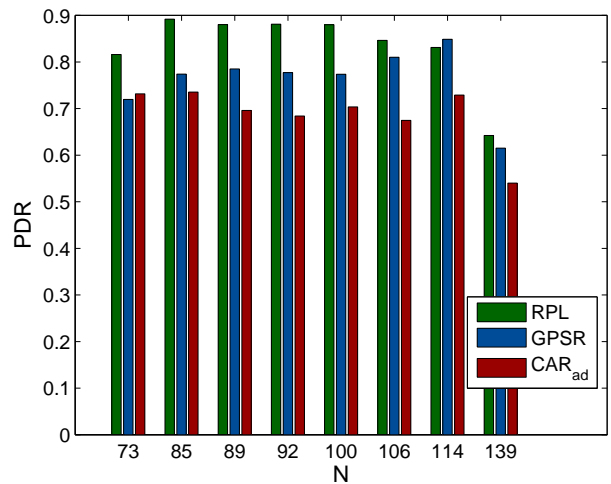


Fig. 3. Packet Delivery Ratio for the three protocols with different vehicular traffic values.

has a more acute effect in $GPSR$ and CAR_{ad} . In the former, because $GPSR$ has a higher average hop count due to the necessary retransmissions performed when a local optimal is found. In the latter because it is possible that it performs more hops than necessary in a given path since it always tries to forward to the node closest to the next anchor point. Therefore, to enhance the PDR, a higher value of $Retry_{MAX}$ was used for RPL than the other protocols.

A clear indicator of the performance of the protocols is given by the total overhead, shown in Figure 4. Here, the average number of messages per second sent is shown for different values of N . We consider that both, the total number of data retransmissions, and the routing messages contribute to the total overhead. Clearly, the lowest overhead is obtained

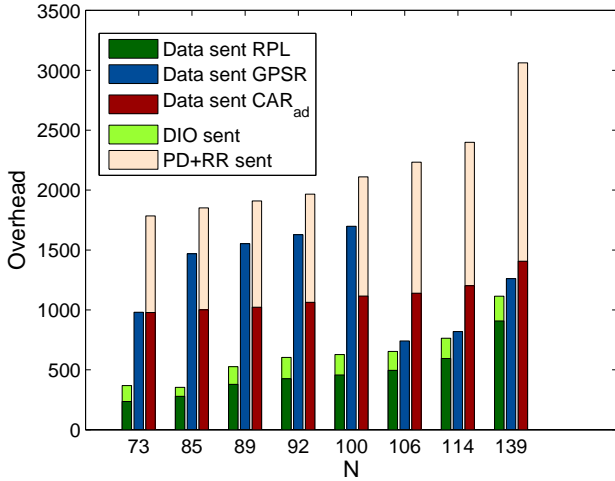


Fig. 4. Total overhead for the three protocols for different vehicular traffic values.

using RPL which implies that the use of the trickle timer effectively reduces the volume of routing messages (DIO) while maintaining an updated state of the network, which can be seen by the low number of necessary data retransmissions. Only when the overhead is similar to *GPSR* ($N=114$), this last protocol obtains a better PDR. In fact, the throughput, measured as the average number of data packets that arrive to the AP per second, is reduced when N grows from 106 to 114, as can be seen in Figure 5. This happens because *RPL* was not able to find a balance between the generation of DIO messages and $Retry_{MAX}$ in a scenario where different vehicular densities coexist. Therefore, a method that takes into account the vehicular density in the neighborhood for setting both T_{DIO} and $Retry_{MAX}$ is desirable.

In the case of *GPSR*, no routing messages are added, since the information for routing is appended to each data message. However, the existence of a local optimum in the path forces the performance of an ineffective perimeter mode. Hence, the number of retransmissions is very high, as can be seen in Figure 4, especially for low vehicular density values, when the probability of finding a local optimum is higher. When $N=106$, there are enough vehicles for connectivity in such a way that the value of $Retry_{MAX}$ that optimized the PDR was reduced. Therefore, a considerable reduction of data retransmissions is achieved. However, *RPL* continues to be more effective in most cases. Finally, the CAR_{ad} protocol is the less effective. Clearly, the amount of PD and RR messages is very high, and creates too much interference in the system. Also, the definition of d_{min} and d_{max} is tricky, and can have very different effects in different parts of the network. Let us recall that the AoI consists of a typical urban with narrow streets surrounded by buildings, but also bridges that cross the river, and a ringroad that is similar to a highway scenario. Therefore, it is difficult that this parameters are optimal for the whole network, causing two different effects if anchors are too

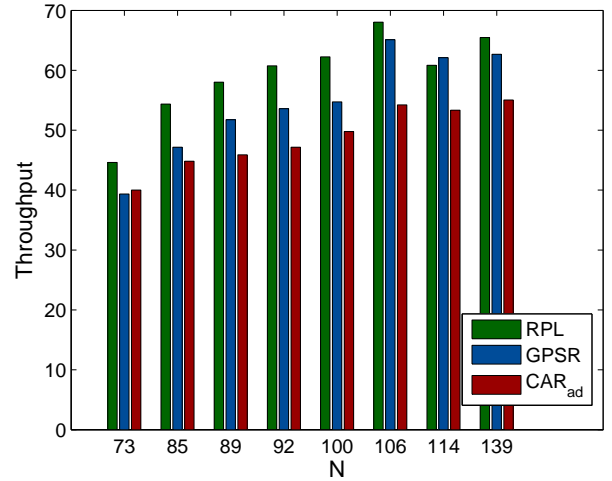


Fig. 5. Average throughput for data received in the AP in packets per second.

close or too far. In the former case, there might be unnecessary hops and hence too many retransmissions causing interference, and in the latter, local optimums due to the inability of vehicles to find paths in areas with obstacles. To evaluate the critical amount of overhead created by this protocol, it can be seen that in the worst case ($N=139$), the overhead is around 56 times the amount of data that arrives to the AP. Such a high relation not only affects the performance of the overall system, but also indicates a poor efficiency in terms of bandwidth use.

In Figure 6, we evaluate the delay of the routing protocols for two scenarios. The first scenario was taken at 7:05 am, when $N=73$. The second scenario is taken at 6:25 am, when $N=139$. Clearly, *RPL* has the lowest delay, and although it does not fulfil the strict requirements of real-time applications (delay < 100 ms), it is able to transmit 99% and 90% of the data packets in less than 1 s for the low traffic and high traffic scenarios respectively. In the case of *GPSR*, only 80% and 65% of the traffic arrives with less than 1s of delay, and it should be noted that the amount of data that arrives is less than with *RPL*. Also, the fact that the delay can be as high as 35 s, is a consequence of the high number of retransmissions associated with this protocol, that make it ineffective as the number of vehicles grows. On the other hand, CAR_{ad} has the highest delay for both scenarios, which is consistent with the amount of overhead that it creates. However, unlike the other protocols, the delay does not improve for the scenario with low traffic. This happens because in the low traffic scenario, in order to enhance the PDR, the vehicles have to wait until they find a neighbor that is closer to the next anchor than themselves. This is effective in terms of PDR, since the performance of CAR_{ad} is better than that of *GPSR*. As traffic increases, vehicles should have more neighbors available, waiting less to transmit, and thus reducing the delay. However, this is not an improvement by itself if we consider

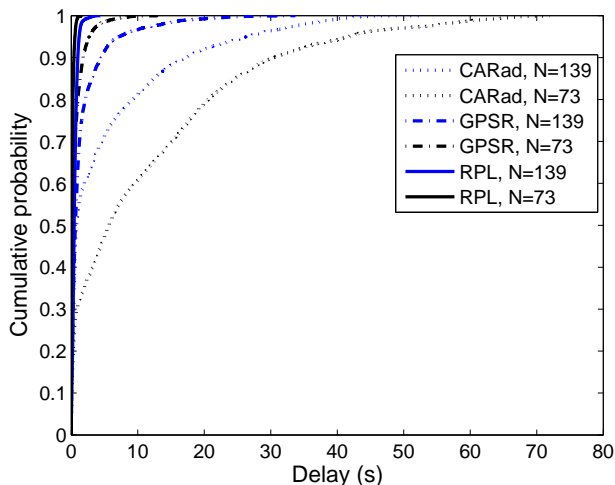


Fig. 6. Delay cumulative probability in two different scenarios.

that the PDR of this protocol for the high traffic scenario is the lowest of all.

VI. CONCLUSIONS

In this work we have adapted two routing protocols for data gathering applications. The first routing protocol, RPL, is a well known topology-oriented protocol for WSNs. It has the advantage of creating a tree from the AP towards the vehicles, establishing paths from each node through a single type of message, and also it is able to control the overhead through the use of a trickle timer. We set an optimization function based on the quality of links to adapt to dynamic conditions of VANETs and after identifying different types of inconsistencies that may arise in data forwarding, included appropriate correction methods. The second adaptation was done over a hybrid protocol, CAR, which creates paths towards the AP as a group of locations, where data must be forwarded following a greedy fashion. We replaced the path setting process by a one-hop exchange that scales better in data gathering scenarios and controlled the overhead with a probabilistic function. Several simulations using a realistic trace of the city of Cologne were performed to compare the two protocols with a geographic-based protocol (GPSR). Results showed that RPL was able to reduce the overhead associated to topology maintenance while at the same time keeping a low delay. On the other hand, the overhead created by the two step path setting of CAR is difficult to reduce, and also the protocol suffers from excessive retransmissions due to its inability to adapt its location based paths to different coexisting geographic conditions in the area of interest. Finally, results showed that it would be desirable to include some adaptive characteristics to the protocols in order to enhance their resiliency to vehicular density changes.

REFERENCES

- [1] U.Lee, B. Zhou, M. Gerla, E. Magistretti, P. Bellavista and A. Corradi. "Mobeyes: smart mobs for urban monitoring with a vehicular sensor network" *IEEE Wireless Communications*, October 2006.

- [2] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden and H. Balakrishnan. "The pothole patrol: using a mobile sensor network for road surface monitoring" in *Proceedings of the 6th international conference on Mobile systems, applications, and services*. 2008.
- [3] J. Rodrigues, A. Aguiar, F. Vieira, J. Barros and J. Silva Cunha. "A mobile sensing architecture for massive urban scanning" in *14th International IEEE Conference on Intelligent Transportation Systems*, 2011.
- [4] B. Karp and H.T. Kung. "Greedy Perimeter Stateless Routing for Wireless Networks", in *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, Boston, MA, August, 2000, pp. 243-254
- [5] V. Naumov, T.R. Gross. "Connectivity-Aware Routing (CAR) in Vehicular Ad-hoc Networks", *Proc. of the 26th IEEE International Conference on Computer Communications. INFOCOM 2007*.
- [6] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur, R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks" *IETF RFC 6550*, March 2012.
- [7] "Vehicular mobility trace of the city of Cologne, Germany", <http://kolntrace.project.citi-lab.fr/>.
- [8] S. Uppoor, O. Trullols-Cruces, M. Fiore and J.M. Barcelo-Ordinas, "Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset", *IEEE Transactions on Mobile Computing*, Vol.13, No.5, May 2014.
- [9] C.Lochert, M. Mauve, H. Fussler and H. Hartenstein "Geographic routing in city scenarios" *ACM SIGMOBILE Mobile Computing and Communications Review*, January 2005.
- [10] K.C. Lee, J. Haerri, U. Kee and M. Gerla. "Enhanced perimeter routing for geographic forwarding protocols in urban vehicular networks" in *proc. of IEEE GLOBECOM workshops 2007*.
- [11] B-C. Seet, G. Liu, B-S. Lee, C-H. Foh, K-J. Wong and K-K. Lee "A-STAR: A mobile ad-hoc routing strategy for metropolis vehicular communications" *Lecture Notes in Computer Science 3042 (0302-9743)* January 2004.
- [12] Q. Yang, A. Lim, S. Li, J. Fang and P. Agrawai "ACAR: Adaptive Connectivity Aware Routing for Vehicular Ad Hoc Networks in City Scenarios", *Mobile Networks and Application*. February 2010.
- [13] A.Maidorawa and K. A. Bakar "An enhanced connectivity Aware Routing Protocol for Vehicular Ad-hoc Networks", *Research Journal of Applied Sciences, Engineering and Technology*, 2014.
- [14] Z. Mo, H. Zhu, K. Makki and N. Pissinou. "MURU: A multi-hop routing protocol for urban vehicular ad-hoc networks" In *proc. of Mobile and Ubiquitous Systems: Networking and Services*, 2006.
- [15] M, Zarmehri, A. Aguiar. "Supporting Sensing Application in Vehicular Networks", in *Proc. of ACM MobiCom Workshop on Challenged Networks - CHANTS*, Istanbul, Turkey, August, 2012.
- [16] Z. He and H. Zhang. "Density adaptive urban data collection in vehicular sensor networks" *Journal of networks*, August 2014.
- [17] C. Palazzi, F. Pezzoni and P. Ruiz. "Delay-Bounded data gathering in urban vehicular sensor networks", *Pervasive and Mobile Computing*, August 2011.
- [18] D. De Couto, D. Aguayo, J. Bicket, R. Morris. "A high-throughput path metric for multi-hop wireless routing", *Proc. of the 9th annual international conference on Mobile computing and networking MobiCom 2003*.
- [19] P. Levis, T. Clausen, J. Hui, O. Gnawali and J. Ko. "The Trickle Algorithm", *IETF RFC 7206*, March 2011.
- [20] K.C. Lee, R. Sudhaakar, J. Ning, L. Dai, S. Addepalli, J.P. Vasseur and M. Gerla "A comprehensive Evaluation of RPL under Mobility", *International Journal of vehicular technology*, Vol 2012.
- [21] V. Naumov, R. Baumann and T.R. Gross. "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces", *Proc. of the 7th ACM international symposium on Mobile ad hoc networking and computing*, 2006.
- [22] "INET Framework", <https://inet.omnetpp.org/>.
- [23] A. Varga, "The omnet++ discrete event simulation system" in *Proceedings of the European Simulation Multiconference 2001*.
- [24] C. Sommer, R. German and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10 (1), pp. 3-15, January 2011.
- [25] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker. "Recent Development and Applications of SUMO - Simulation of Urban Mobility", *International Journal On Advances in Systems and Measurements*, 5 (3-4) 128-138, December 2012.

- [26] C. Campolo, A. Vinel, A. Molinaro and Y. Koucheryavy, "Modeling Broadcasting in IEEE 802.11p/WAVE Vehicular Networks", IEEE Communications Letters, Vol.15, No.2, February 2011.