

# ADOPEL: ADAPTIVE DATA COLLECTION PROTOCOL USING REINFORCEMENT LEARNING FOR VANETS

Ahmed Soua and Hossam Afifi

Department of RST, Institut Mines-Telecom, Telecom SudParis, Evry, France

Received 2014-03-31; Revised 2014-04-01; Accepted 2014-07-22

## ABSTRACT

Efficient propagation of information over a vehicular wireless network has usually remained the focus of the research community. Although, scanty contributions have been made in the field of vehicular data collection and more especially in applying learning techniques to such a very changing networking scheme. These smart learning approaches excel in making the collecting operation more reactive to nodes mobility and topology changes compared to traditional techniques where a simple adaptation of MANETs propositions was carried out. To grasp the efficiency opportunities offered by these learning techniques, an Adaptive Data collection Protocol using reinforcement Learning (ADOPEL) is proposed for VANETs. The proposal is based on a distributed learning algorithm on which a reward function is defined. This latter takes into account the delay and the number of aggregatable packets. The Q-learning technique offers to vehicles the opportunity to optimize their interactions with the very dynamic environment through their experience in the network. Compared to non-learning schemes, our proposal confirms its efficiency and achieves a good tradeoff between delay and collection ratio.

**Keywords:** Data Collection, Vehicular Ad Hoc Networks (VANETs), Reinforcement Learning, Qlearning, Collection Ratio, Number of Hops

## 1. INTRODUCTION

Most VANET applications are based on a dissemination Process (Soua *et al.*, 2012; Badawy *et al.*, 2010; Bi *et al.*, 2010; Singh and Gupta, 2011; Chou and Yang, 2010) on which an information must be propagated to rather long distance so that drivers can be alerted in advance. Since each vehicle in a vehicular environment can detect a hazardous situation or a congestion zone, the number of messages pumped on the network might increase dramatically. Consequently, the network performances are severally affected leading to bandwidth waste, large overhead and a high probability of wireless collision. Thus, data gathering/collection is regarded as an important approach to circumvent these problems. It makes inter-vehicle communications more efficient and reliable and minimizes the bandwidth utilization.

In literature, there are several proposals studying data collection protocols in VANETs. However, in our

opinion, the existing related works are still not satisfactory and we feel it is possible to obtain better results. First of all, most the proposed techniques were adapted from MANET proposition and their adjustment to vehicular conditions raises a lot of discussions and critics. Furthermore, most of the proposed approaches ignore the fast topology changes of VANETs and therefore their performance and effectiveness in such conditions rise some doubts.

On the other hand, the use of learning techniques marked a fundamental and farsighted parting from preceding approaches dealing with information exchange in very dynamic networks. In fact, learning schemes deal with an on-line search to find an optimal decision policy and hence adapt it to the high mobility of nodes. In these scenarios, an agent optimally enhances its interactions with the very dynamic environment by taking actions and receiving reward for performing well or receive penalty for failure. By applying this approach in

**Corresponding Author:** Ahmed Soua, Department of RST, Institut Mines-Telecom, Telecom SudParis, Evry, France

information exchange between moving vehicles, a further efficiency can be achieved and thus the robustness of networking proposals can be strengthened against the very changing network topologies. Nevertheless, research efforts in applying learning techniques when designing data collection schemes for VANETs remain scanty.

To fill this gap, we propose in this study a novel data collection technique devoted to vehicular networks, denoted as ADOPEL, designed with the goal of making the collecting operation more reactive to nodes mobility and topology changes. It is based on a distributed Qlearning technique where a reward function is provided and defined to take into account the delay and the number of aggregatable packets.

We have to mention that in (Soua and Afifi, 2013), we presented a short description of ADOPEL with a limited performance study. Here, a refined description is carried out to explain in depth the different functioning steps of ADOPEL. In fact, we reveal how our proposal reacts towards the challenging characteristics of a VANET network: Some issues such as the high mobility of nodes, the selection of next relay, the stability of the route towards the control center, etc. In addition, the performance evaluation part will be enriched with additional details that confirm the efficiency of our scheme compared to non-learning techniques.

The remainder of this study is organized as follows. The next section presents the related work and outlines the different procedures undertaken to design efficient data collection protocols devoted to VANETs. The system specification is presented afterward followed by an underlining of the basic design of our proposal and its functioning principles. Finally, simulations results are presented and discussed in section IV to confirm the effectiveness of our technique.

## 2. RELATED WORK

Data gathering related literature reveals two main aspect for gathering issues. On one hand, some contributions focus on the manner of routing the aggregatable messages along farther distance in order to improve the aggregation ratio (data collection) (Yu *et al.*, 2010; Dieudonne *et al.*, 2012). On the other hand, other studies concentrate on expressing data to be aggregated differently by using compressing and merging methods to reduce the overhead (Cherfaoui *et al.*, 2008).

In our case, we focus on how to route the aggregatable packets to a specific destination node in

order to improve the data collection ratio and hence obtain more accurate global traffic information. Hence, we are not interested here on the mechanisms to express data differently.

Several works have been proposed to investigate the data collection concept by adopting different approaches.

Saleet *et al.* (2010), authors propose a location service management protocol that solves the location querying and updating problems by aggregating the location information data. In this scheme, the vehicle's mobility space is viewed as a grid network which is partitioned into several segments and each segment is divided into a number of cells. The central node of a segment plays the location server role. This server is responsible for storing current location information about all nodes belonging to the same segment. Then, the server aggregates this information and broadcast it to the neighbors. In addition, the protocol uses message aggregation in location querying. It introduces some delays before forwarding the queries in order to gather more queries and aggregate them. This proposal is based on poor flooding to disseminate data on the network which presents a great weakness for this approach. In addition, the choice of the grid structure for the vehicle's mobility space is not justified and makes some ambiguous in this study.

Yu *et al.* (2010) focus on making similar reports broadcasted by vehicles meeting each other in order to be aggregated together. In fact, this technique dynamically changes the forwarding speed of nearby reports so that they can be delivered to the same node at the same time and then be merged into a single report. This adaptive forwarding is based on a distributed learning algorithm on which each node learns from local observations and chooses a delay based on the learning results. Simulation results outline the effectiveness of the proposed technique.

Ibrahim and Weigle, (2008), authors present their proposal, called CASCADE, where they expose a new clustering-based data aggregation technique. This protocol uses two types of reports: Primary and aggregated records. The first ones are broadcasted periodically by the nodes and comprise the local view of each vehicle. Then, the local view is grouped into clusters and used to compact and aggregate the local view data into an aggregated record. This aggregated record is then broadcast to neighboring vehicles to provide them information about vehicles beyond the local view. This technique allows vehicles to have an extended view of the road behind and then accurate information about upcoming traffic conditions.

However, their approach introduced a large overhead to build the global view.

Another effort carried out by Dieudonne *et al.* (2012) focuses on a distributed collection information for VANETs. It collects data produced by vehicles using inter-vehicle communications only. It is based on the operator *ant* allowing to construct a local view of the network and therefore to collect data in spite of the network topology changes. A theoretical proof of correctness and experiments confirm the efficiency of the proposed technique.

Nadeem *et al.* (2004) introduce a system for data dissemination and aggregation in a vehicular context namely Traffic View. In this system, an aggregate record is composed of specific information: Single speed, position, timestamp value and a list of vehicle's IDs. The authors propose two aggregation schemes: Ratio and cost based techniques. In the ratio-based, the most important parameter is the aggregation ratio which indicates the number of vehicles to be aggregated into one single frame. For the cost based technique, a specific cost function is defined for each aggregating vehicle. A high cost is assigned for the vehicles that are close to the aggregating node. Thus, the produced view of traffic is not useful to any vehicle unless it is in the proximity of the aggregating vehicle.

Lochert *et al.* (2007; 2010) focus on cooperative information gathering and sharing applications in VANETs and propose a hierarchical aggregation algorithm. Their proposal is based on probabilistic data representation Flajolet-Martin sketches, which they extended to a soft-state data structure. In their scheme, there is no longer a need to decide which aggregate contained more up-to-date information since the resulting aggregate comprises all the information from all aggregates that have been merged. Nevertheless, this study does not consider routing related-issues but focus only on data representation.

The aforementioned aggregation/collect approaches do not strictly consider the potentially mobility issue and the collection ratio in finding a suitable relay in the collect process. In fact, most of the listed works focus on the representation and the processing of the aggregated data and neglect how to obtain the raw information among the running vehicles.

Some works, such as (Saleet *et al.*, 2010; Ibrahim and Weigle, 2008), do not consider at all the effect of the mobility issue on their proposed scheme. In the other cited works, authors investigate the performance of their proposals with a dynamic network configuration. Nevertheless, in (Nadeem *et al.*, 2004; Lochert *et al.*, 2007; 2010), authors studied the representation of the aggregated information and neglect the importance of the routing mechanism and the improvement of the

collection ratio during the gathering process. Such a weakness can easily stop the message progress towards its final destination since the stability of link is not considered. Moreover, neglecting the improvement of the collection ratio can result in poor aggregated information which cannot provide a clear view of the total system to the user. The resting cited approaches neglect the stability of the route toward the destination. Compared to this literature, the advantages of our technique are threefold: First, we design a scheme that is based on a learning technique which allows each vehicle to dynamically adapt its forwarding strategy (i.e., the selection of the next relay) based on a two-step view of the network. This learning process allows a dynamic reaction of ADOPEL when the topology of the network is changing. Furthermore, this selection takes into account the stability of the links between vehicles (by introducing a new parameter in the learning algorithm) to achieve a rapid and efficient travel of the aggregated message toward the destination. Finally, to ensure a good quality of the aggregated information, ADOPEL selects relays that can enhance the collection ratio during the gathering process. To do so, next hop that is surrounded by a high number of neighbors will be selected as next relay.

In the next section, we present in depth our proposal which is interested on collecting aggregatable packets from vehicles taking into account the dynamicity of the network. We use Q-learning method to select next hops aiming at collecting more raw data.

### 3. PROPOSED TECHNIQUE

Hereinafter, we introduce ADOPEL technique-an Adaptive distributed data collection protocol using reinforcement learning for VANETs. The proposal is based on a distributed learning algorithm on which a reward function is defined. This latter takes into account the delay and the number of aggregatable packets and hence makes the collection operation more reactive to nodes mobility and topology changes. After describing the system specifications, we show the functioning algorithm of our technique to investigate in details the different working steps of ADOPEL.

#### 3.1. System Specifications

ADOPEL considers that each communicating vehicle knows its current position and speed using a positioning system such a Global Positioning System (GPS). Furthermore, we assume that vehicles exchange two types of messages: Beacons and event driven messages. Where the former aims at improving driver awareness of surrounding environment by exchanging information about position, velocity, direction, etc. and the latter is

triggered when a vehicle needs to collect traffic data toward a control center.

This collect operation is started by a node called *initiator* and involves a limited number of vehicles. Here, the *initiator* is a vehicle that is leading a group of nodes and running in a highway. The *initiator*, at each gathering operation, is randomly selected from vehicles. Thereby, the *initiator* has to collect the traffic data from vehicles and deliver it to a Traffic Control Center (TCC) in order to be processed and studied (Fig. 1). We assume here that TCCs are sufficiently deployed along the freeway.

In a vehicular context, the collect of traffic related data is periodically carried out and transmitted to a TCC in order to have an up to date big picture of the road. Thus, ADOPEL triggers periodically a collect operation (aggregation request) from the *initiator* toward a TCC. The aggregation operation is done step by step until arriving to TCC. At each step, the best neighbor is selected as a next relay. This selection is based on the Q-value, determined by the Qlearning algorithm. The collected data is provided to the TCC when this latter is reachable by the node ending the gathering operation. To limit the collection process, we use a  $d_{collect}$  parameter, representing the depth of the collecting operation, i.e., the maximal distance in meters from the *initiator*. Indeed, this parameter reflects the zone that will be concerned by the collecting process. Thus, each additional meter increases the total duration of the collect as well as the number of messages to collect;  $d_{collect}$  is then an interesting parameter, impacting directly the performance of our proposal. The type of data to be collected is specified by the *initiator* and included in the collect packets. For instance, in our scenarios, ADOPEL deals with collecting the speed of surrounding vehicles with the aim of computing the average speed of the concerned road. However, this data type can be extended to other useful information as well as real-time fuel consumption, pollution indicators and parking lots availability services, etc. As mentioned previously, we focus on the manner of routing the aggregatable messages (selecting the appropriate relay) along farther distance in order to improve the data collection ratio.

### 3.2. Protocol Design

#### 3.2.1 Distributed Qlearning in ADOPEL

The frequent topology changes in the vehicular context make it necessary to adapt the aggregation and the forwarding policy to the network state. In fact, it is difficult to predict in advance the set of rules that will adjust the actions of each vehicle when the vehicular environment's variables are changing.

Fortunately, the reinforcement learning techniques (Panait and Luke, 2005) can tackle these problems. In reinforcement learning, each vehicle is a learner. Each vehicle tries to optimize its interactions with the very dynamic environment through its experience. The experience here is expressed in terms of errors and rewards. In addition, the vehicles collaborate with each other to share their feedbacks and establish the distributed learning system.

In this study, we model the aggregation operation in VANET as a Markov Decision Problem (MDP) that can be solved by reinforcement learning. Each vehicle (agent) decides at each state which action to take based on its experience. After taking an action, the agent gets a reward or a cost from the environment.

The Markov decision problem is defined as a tuple  $\{s, a, r\}$ :

- $s$  is the states set; In our work, the packet state is the current vehicle
- $a$  is the set of actions a vehicle can perform: In our scheme, the action of a node is to select the next relay that will maximize the aggregation ratio. Hence, the possible set of actions allowed at each node is nothing but the set of neighbors
- $r$  is the immediate reward a vehicle may receive after taking an action  $a$

To solve this MDP model, we propose to use a reinforcement learning algorithm. The literature provides a large number of reinforcement learning approaches, such as temporal difference learning, direct utility estimation and Q-learning (Russel and Norvig, 2009). We are motivated to use Q-learning algorithm since it allows comparing the expected utility of the available actions without requiring knowledge of the environment's model.

A  $Q(s, a)$  matrix is used to store the learned reward/cost for each state and action pair. For example  $Q(s, a)$  is the expected reward for taking an action  $a$  at state  $s$ . The updating function of  $Q(s, a)$  is defined as Equation 1:

$$Q(s, a) = (1 - \alpha) * Q(s, a) + \alpha * (r + \gamma * \max_{a'} Q(s', a')) \quad (1)$$

Where:

- $\alpha$  = denotes the learning rate which model here how quickly the Q-values can change with a dynamic network topology
- $\gamma$  = refers to the discount factor. It models the fact that immediate reward is or not more valuable than future reward. If

this value is high, future rewards are more valued than immediate reward. In the opposite case, the learning algorithm count immediate rewards more strongly

$r$  = represents the expected immediate reward of executing action  $a$  at state  $s$   
 $\max_a Q(s', a')$  = models the maximum expected future reward when the system reaches the state  $s'$  after taking the action  $a$

The most important challenge to successfully achieve the collection performance is to define the suitable reward function. In fact, the vehicle will use this function to update its forwarding policy.

For immediate rewards, we consider the most relevant parameters effective in decision. First factor is based on the number of neighboring vehicles that each node possesses in its transmission range. In fact, the reward should be more for a vehicle with a high number of neighbors. Secondly, the aggregation proposal must route the packet to the destination in a limited delay. Thus, the node has to choose the node that offers the most relevant advance to the destination. It is worth saying that our proposal focuses on a collection process rather than a rapid propagation of a packet in the network. This

observation has to be considered on the reward function.

Based on these decision factors, we formulate the reward function as follow Equation 2:

$$r = \begin{cases} \beta * (1 - \frac{1}{data(i)}) + (1 - \beta) * (\frac{adv(i)}{adv(i)_{avg}}) & \text{if next hop is not the destination} \\ K1 & \text{if next hop is the destination} \\ -K1 & \text{if the vehicle hasn't neighbors} \end{cases} \quad (2)$$

The reward function considers several routing scenarios to improve the aggregation ratio and guarantee a steady advance to the destination.

The first item in Equation 2 combines the normalized number of neighboring nodes that the next hop possesses and its normalized progress toward the destination.  $adv$  refers to the advance of the node  $i$  (current node) to the destination vehicle  $D$ , situated at a distance  $d_{collect}$  from the initiator, by choosing the neighboring node  $j$  as the next hop. This parameter  $d$  can be seen as the depth in distance of the collection process.

Hence, this advance can be expressed as follows Equation 3:

$$adv(ij) = dist(i, d) - dist(j, d) \quad (3)$$

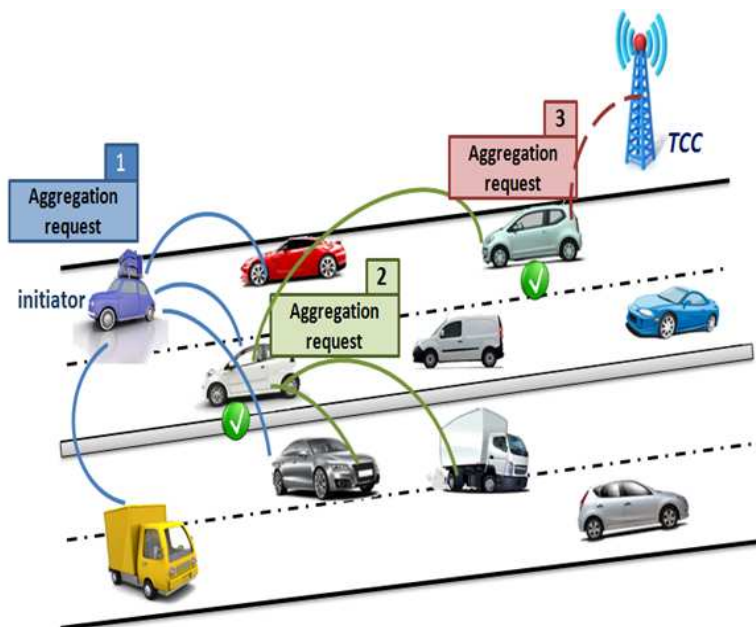


Fig. 1. ADOPEL functioning overview

The average advance is given by Equation 4:

$$adv(i)_{avg} = (\sum_{k=1}^t adv(ik)) / nb_{neighbor(i)} \quad (4)$$

where,  $t = nb_{neighbor(i)}$  is here the total number of neighbors of node  $i$ . Thus, more reward is assigned to the next hop with more neighbors and larger relative advance. In fact, a node with a higher number of surrounding vehicles and a higher advance toward the TCC allows respectively a larger quantity of collected data and a faster delay to reach the destination.

The second item in Equation 2 denotes the reward if the node can reach directly the destination  $D$ . In this case, the reward is a positive constant  $K1$ .

Finally, the last item is to solve the “void” problem in geographic routing. In fact, when a node receives a packet and cannot find a neighboring vehicle, it drops the packet and sends a negative reward to the sending node to inform a forwarding failure. Then, the sending node will choose another vehicle to send the packet based on the Q-values. The node with the highest Q-value will be selected.

As an important feature in our proposal, we use a variable discount factor called  $\gamma'$  to handle the instability of the vicinity. This parameter depends on the link stability. In fact, the node selected as a next relay is the vehicle that will spend more time in the vicinity of the sending vehicle. In this way, we ensure that the route we select is more stable. For that purpose, we define a stability factor  $SF_i$  as:

$$SF_i = \begin{cases} \sqrt{\frac{|N_i| \cap |N_{i+1}|}{|N_i|}} & \text{if } |N_i| \neq 0 \\ 0 & \text{Otherwise} \end{cases} \quad (5)$$

where,  $N_i$  ( $N_{i+1}$  respectively) is the current neighbor set of the sending node  $i$  (the forwarding node  $i+1$  respectively). Neighbor list can be attached to the hello messages exchanged between vehicles. As aforementioned, the  $SF$  will reflect a higher value for a relatively stable couple of neighbors. Then, a node calculates the discount factor  $\gamma'$  as Equation 6:

$$\gamma' = \gamma * \sqrt{SF_i} \quad (6)$$

Therefore, every time a node has a packet to send, it calculates the reward for its neighboring set, the stability factor and updates the Q-values of its matrix using the following equation:

$$Q(s,a) = (1 - \alpha) * Q(s,a) + \alpha * (r + \gamma' * \max_{a'} Q(s',a')) \quad (7)$$

The vehicle with the highest Q-value will be selected as next hop.

### 3.2.2. Exploration Vs Exploitation

In reinforcement learning there is a balance between exploitation and exploration. Exploitation occurs when the action selection strategy is based on the highest value of the Qtable. In this case, exploitation will lead to locally optimal policies since the selection is greedy. In the case of most of the optimization problems, this will not lead necessary to a global optimum.

On the other hand, exploration consists on taking risk by choosing the non-optimal action and exploring other choices to obtain more knowledge about the network. Obviously, excessive exploration degrades the performance of the Qlearning approach.

Thus, convergence is an important issue for our proposed algorithm. Nevertheless, in (Watkins and Dayan, 1992), authors demonstrate that a Q-learning scheme converges to the optimum actions-values provided that “all actions are repeatedly sampled in all states and action-values are represented discretely”. Here, the conditions of convergence are insured. In fact, ADOPEL uses hello messages to sample all its neighbors by computing the  $\gamma'$  factor. In addition, the action-values (Q-values) are represented discretely in ADOPEL. As a result of that, we can say loudly that our proposed technique converges to the optimum action values.

### 3.2.3. ADOPEL Algorithm Overview

Based on the description given in the previous section, we summarize hereafter the different steps of ADOPEL.

As stated above, each node uses the received “hello” messages from neighbors to a build a neighboring node table. The “hello” messages contains in addition to the usual information the list of neighboring nodes. This way, each vehicle can maintain its two-hop neighbor list and can easily compute the stability factor given by Equation 5.

#### Algorithm 1: ADOPEL algorithm

1. For each node  $i$  do
2. Send a data collection request to neighboring nodes.
3. L1, L2, L3 are 3 lists initialized by NULL.
4.  $Q_i$  is initialized based on the number of surrounding vehicles and the advance toward the destination.
5.  $N_i$  is the neighboring nodes set of node  $i$
6. If ( $N_i \neq \emptyset$ ) then
7. For ( $j \in N_i$ ) do

8.  $i$  compares each of its neighboring node  $j$  as follows :
9. If  $(nb_{neighbor(j)} \neq 0 \text{ and } advance(j) > advance(i))$  then
10.  $L1 \leftarrow j$
11. end If
12. If  $(nb_{neighbor(j)} = 0 \text{ and } advance(j) > advance(i))$  then
13.  $L2 \leftarrow j$
14. end If
15. If  $(nb_{neighbor(j)} \neq 0 \text{ and } (advance(j) < advance(i)))$  then
16.  $L3 \leftarrow j$
17. end If
18. end For
19. If  $(L1 \neq \emptyset)$  then
20. Next hop will be the one with largest Q-Value
21. end If
22. If  $(L1 = \emptyset \text{ and } L2 \neq \emptyset)$  then
23. The node with largest Q-Value will be the next Hop
24. end If
25. If  $(L1, L2 = \emptyset \text{ and } L3 \neq \emptyset)$  then
26. Next Hop will be chosen from L3 with the highest Q-Value.
27. end If
28.  $i$  relays the message to the selected next hop after making aggregation (computing average value)
29. Else
30. /\*  $N_i = \emptyset$  \*/
31.  $i$  generates a negative reward.
32.  $i$  chooses its previous source as the next hop.
33. end If
34.  $i$  computes the reward after making the relaying process based on Equation 2

35.  $i$  updates the Q-Value  $Q(s,a)$  using Equation 7.
36. End For

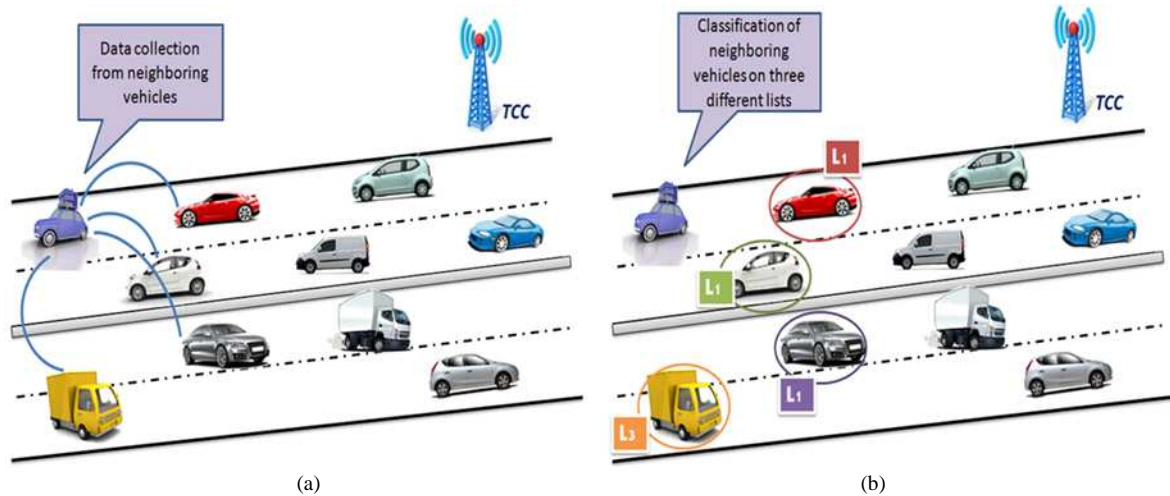
Algorithm 1 shows the different steps of the execution of ADOPEL on a each node  $i$  whenever this latter receives a collect request. This execution is triggered periodically by an *initiator* node.

As illustrated by **Fig. 2**, upon receiving a relaying request, the first step undertaken by a node  $i$  aims at collecting data from neighbors by sending them a collect data request. Afterward, the node processes the data received (e.g.: It computes the average value of the received ones) and starts the relaying process.

For the relaying process, it classifies the neighboring nodes on three different lists. Highest priority is attributed to vehicles that are more surrounded and closest to the final destination node situated at a distance  $d_{collect}$ . Notice that a vehicle with a large number of neighboring nodes leads to a larger quantity of collected data.

The second phase consists on selecting the appropriate relay node based on the previous classification (**Fig. 2c**). This operation depends on the Q-values of each candidate node. In fact, nodes with high values of Q are prosperous. Once the selection of the relay vehicle is performed, the sending node computes the immediate reward  $r$  and then calculate the total expected reward  $Q(s,a)$ .

Since the collection process is periodically initiated by the *initiator*, a node  $i$  is involved in this operation for few times before leaving the concerned road. Thus, the vehicle learns from its acquired experience (rewards or costs) to select the appropriate relay node ensuring a good collection ratio and a faster propagation toward the destination.



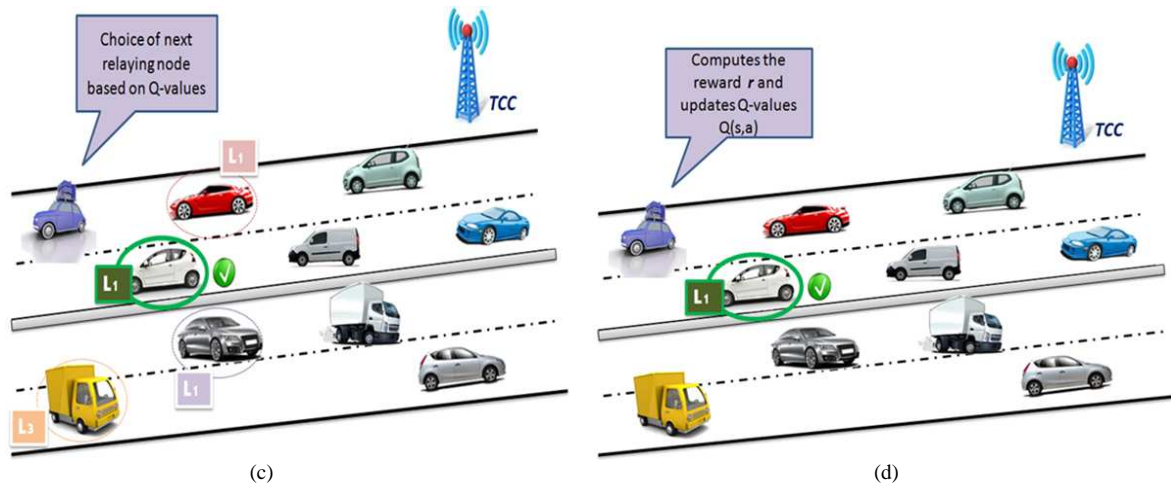


Fig. 2. ADOPEL Algorithm Overview (a) Collecting data from vehicles (b) Classification of neighboring vehicles (c) Selection of the next relay (d) Updating Q-Values

#### 4. PERFORMANCE EVALUATION

In this section, we show our simulations results and investigate the performance of our proposal in terms of collection ratio and number of hops. We compare our scheme to a non-learning protocol. We call a collection technique “nonlearning” when a first part of relays are selected based on the number of their neighboring cars and the other part are selected based on their advance toward the final destination. The destination is situated at a distance  $d_{collect}$  behind the *initiator*.

##### 4.1. Simulation Design

We used MATLAB to conduct simulations using Freeway mobility model. The freeway mobility model emulates the motion behavior of vehicles in a freeway. In our study, we use a freeway which has two lanes in each direction. All lanes of the freeway are 20 Km in length (Fig. 3).

To make the proposed scheme tractable, we make the following assumptions:

- We assume an ideal MAC layer without contention and collision
- All nodes have the initial transmission range equal to 200 m
- The number of vehicles was varied from 200 to 400.
- All vehicles are initially positioned at the entrance of the freeway
- We respectively assigned to  $\alpha$ ,  $\beta$  and  $\gamma$  the following values: 0.8, 0.7 and 0.8
- For the data collection depth, we set  $d_{collect}$  equal to 1500 m

In addition, each vehicle stores its own Q-values and the ones received from neighboring nodes (using hello messages) in a matrix to be used in the relaying process.

Furthermore, we compare our scheme to a non-learning version. For that purpose, we suppose for the non-learning schemes that at each relaying operation, a node has a 20% (respectively 40%) probability of choosing the most surrounding vehicles as a next relay and 80% (respectively 60%) to choose the node with the largest advance toward the destination node.

##### 4.2. Simulation Results

In this section, we focus on the performance of our technique both for the average data collecting ratio and the average number of hops required to reach the final destination node.

Figure 4 depicts the average data collection ratio when varying the density of nodes for the two techniques. We can observe that our proposed scheme outperforms loudly the nonlearning versions. In fact, in all cases, ADOPEL achieves a gain of over than 20% compared to the other techniques. This can be explained by the fact that in very dynamically changing networks as VANETs, ADOPEL can change adaptively to better relaying nodes to increase the collect ratio as the network topology changes, whereas the others non-learning protocols find major difficulties to adapt to the dynamicity of the network.

To make a fair analysis, we investigate in Fig. 5 the average number of hops needed to travel the collect distance  $d_{collect}$ . Indeed, a good collect ratio might have a heavy cost and then can be a real weakness for the algorithm. However, Fig. 5 shows that the gap between



the three techniques is very tight even ADOPEL achieves higher values than the others schemes. This

clearly implies that our technique achieves a good tradeoff between delays and collection ratio.

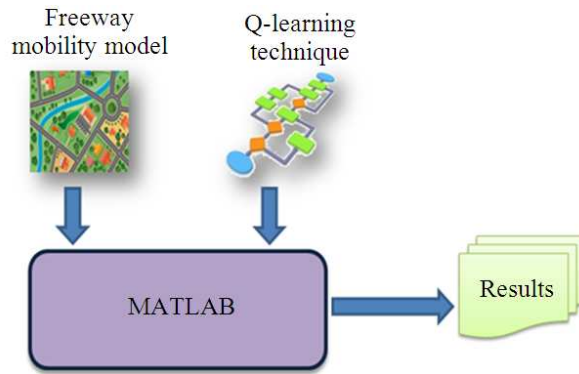


Fig. 3. ADOPEL simulation framework

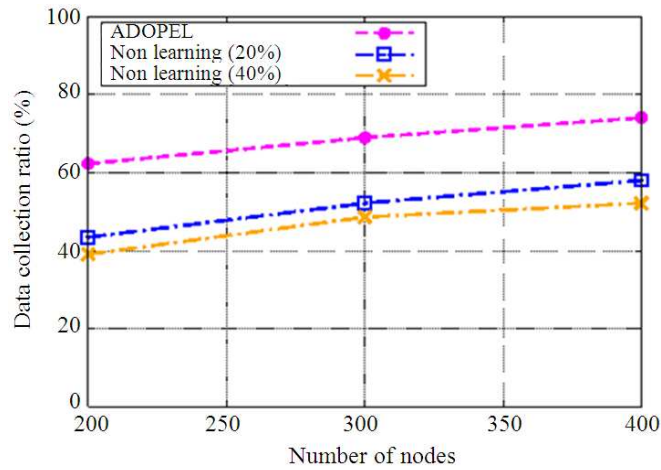


Fig. 4. The average data collect ratio: ADOPEL Vs non-learning versions

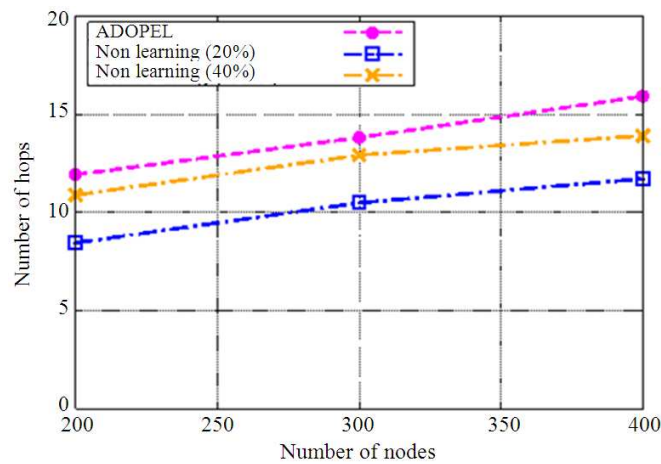


Fig. 5. The average number of hops: ADOPEL Vs non-learning versions

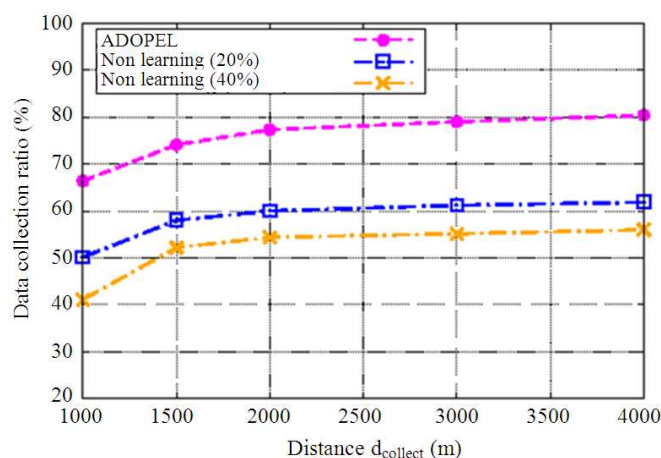


Fig. 6. Effect of the variation of  $d_{collect}$  on the ratio of collected packets

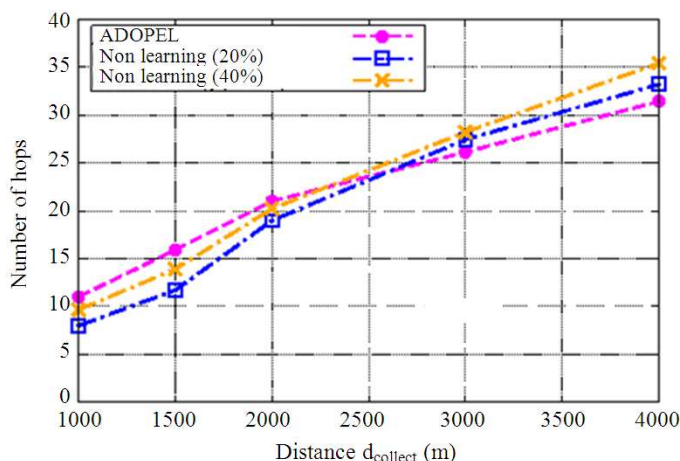


Fig. 7. Effect of the variation of  $d_{collect}$  on the number of hops

This is because ADOPEL takes the stability of vicinity into account which yields in a higher probability of using nodes moving in the same direction as the destination node to relay aggregated messages. On the other side, in non-learning versions, the source node may select a node moving in opposite direction as a next hop which can be very vulnerable. As a result, many data collection operations may be penalized when relaying vehicles became far away from the destination.

To unravel the impact of the depth of the collecting operation on the performance of our proposed technique, we investigate afterward the variation of the parameter  $d_{collect}$  and how it will affect the collection ratio and the total number of hops required to reach the TCC. The total number of moving vehicles is equal to 400 for all the following scenarios.

In Fig. 6, we show the variation of the collection ratio as a function of the distance  $d_{collect}$ . We clearly observe, for the three collecting schemes, that the greater the distance  $d_{collect}$  the higher the ratio of collected packets. This observation is perfectly expected because when the gathering operation will be extended to additional parts of the vehicular network and hence it will touch more vehicles.

Therefore, the ratio of the implicated vehicles in the collecting operation increases which results in a higher gathering percent. However, our proposal performs better performance than the non-learning techniques. It can be observed also that the outperformance of ADOPEL is more clear for longer distance. This can be explained by the fact that longer distances permit to the learning operation to be more efficient when updating the Q-

learning values since we have in this case a more global view of the network which affects the learning process.

We present in **Fig. 7** the impact of  $d_{collect}$  on the number of hops to reach the control center. Obviously, as  $d_{collect}$  increases, the number of required relays to reach.

TCC becomes more important. This observation underlines the need of a good trade-off between the two metrics: Collection ratio and end-to-end delays. Higher delays may be accepted for non sensitive delay applications (like e-traffic and infotainment applications, etc.), however, for e-safety applications, delays have to be less as possible. The comparison, as shown in **Fig. 7** between learning and non-learning techniques reveals that our technique achieves better results than the other approaches specially when the distance  $d_{collect}$  is very important. This can be explained by the fact that with a higher distance non learning distance encounters several difficulties in finding the good path toward the TCC regarding the instability of wireless links between neighboring nodes and the higher probability to choose the vehicle moving in the wrong direction. On the other side, ADOPEL, with its stability approach to choose next relays ( $SF_i$  factor) and learning technique, overcomes the negative effects of large distances and achieves better results in all cases.

## 5. CONCLUSION

In this study, we have tried to tackle an inherent challenging problem related to vehicular communications by developing a data collecting technique aiming at gathering raw data from moving vehicles. We proposed a total distributed scheme, namely ADOPEL, based on a Qlearning technique making the collecting operation more reactive to nodes mobility and topology changes. For that purpose, we defined a reward function to take into account the delay and the number of aggregatable packets. In addition, a novel expression of the discount factor  $\gamma$  was provided to handle the instability of the vicinity and to choose the most stable route toward the control center where the raw data will be treated. The Q-learning technique offered to vehicles the opportunity to optimize their interactions with the very dynamic environment through their experience in the network. Compared to other techniques present in literature, such as (Saleet *et al.*, 2010; Ibrahim and Weigle, 2008), our scheme gives vehicles the possibility to auto-adapt their gathering process based on their experience in the network and hence adds the dynamicity aspect to the data collection parameters rather than fixing all of them since the

beginning. Moreover, adding the instability of links in the learning process variables represents another important feature of our method that distinguishes it from previous works (Yu *et al.*, 2010).

To analyze the performance of our proposal, we compared it to a non-learning version to study the effect of the learning technique. We used two important metrics which are directly linked with the efficiency of our collecting approach: The collection ratio and the number of hops. A good technique must achieve a tradeoff between these two metrics to guarantee its success. Our simulation results showed that our technique far outperforms other propositions and achieves a good tradeoff between delay and collection ratio.

In terms of future work, we are interested in applying learning scheme in dissemination-related issues in VANETs and study the effect of the past experience-based dynamic learning on the efficiency of the broadcast process. For that purpose, we intend to evaluate how the reward and sanction mechanisms can lead to a faster propagation of the emergency message toward the risk zone.

## 6. REFERENCES

- Badawy, G., J. Mistic, T. Todd and D. Zhao, 2010. Performance modeling of safety message delivery in vehicular ad hoc networks. Proceedings of the IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications, Oct. 11-13, IEEE Xplore Press, Niagara Falls, Canada, pp: 188-195. DOI: 10.1109/WIMOB.2010.5644987
- Bi, Y., L.X. Cai, X. Shen and H. Zhao, 2010. A cross layer broadcast protocol for multihop emergency message dissemination in inter-vehicle communication. Proceedings of the IEEE International Conference on Communications, May 23-27, IEEE Xplore Press, Cape Town, pp: 1-5. DOI: 10.1109/ICC.2010.5501865
- Cherfaoui, V., T. Denoeux and Z. Cherfi, 2008. Distributed data fusion: Application to confidence management in vehicular networks. Proceedings of the 11th International Conference on Information Fusion, Jun. 30-Jul. 3, IEEE Xplore Press, Cologne, pp: 1-8.
- Chou, L.D. and Y.T. Yang, 2010. Location-based directional broadcast for inter-vehicle communications. Proceedings of the IEEE 72th Vehicular Technology Conference Fall, Sept. 6-9, IEEE Xplore Press, Ottawa, pp:1-5. DOI: 10.1109/VETECF.2010.5594431

- Dieudonne, Y., B. Ducourthial and S.M. Senouci, 2012. COL: A data collection protocol for VANET. Proceedings of the IEEE Intelligent Vehicles Symposium, Jun. 3-7, IEEE Xplore Press, Alcalá de Henares, pp: 711-716. DOI: 10.1109/IVS.2012.6232266
- Ibrahim, K. and M.C. Weigle, 2008. CASCADE: Cluster-based accurate syntactic compression of aggregated data in VANETs. Proceedings of the IEEE GLOBECOM Workshops, Nov. 30-Dec. 4, IEEE Xplore Press, New Orleans, pp: 1-10. DOI: 10.1109/GLOCOMW.2008.ECP.59
- Lochert, C., B. Scheuermann and M. Mauve, 2007. Probabilistic aggregation for data dissemination in VANETs. Proceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks, Sept. 9-14, ACM New York, pp: 1-8. DOI: 10.1145/1287748.1287750
- Lochert, C., B. Scheuermann and M. Mauve, 2010. A probabilistic method for cooperative hierarchical aggregation of data in VANETs. *Ad Hoc Netw.*, 8: 518-530. DOI: 10.1016/j.adhoc.2009.12.008
- Nadeem, T., S. Dashtinezhad, C. Liao and L. Iftode, 2004. TrafficView: Traffic data dissemination using car-to-car communication. *ACM Mobile Comput. Commun. Rev.*, 8: 6-19. DOI: 10.1145/1031483.1031487
- Panait, L. and S. Luke, 2005. Cooperative multi-agent learning: The state of the art. *Autonomous Agents Multi-Agent Syst.*, 11: 387-434. DOI: 10.1007/s10458-005-2631-2
- Russel, S.J. and P. Norvig, 2009. *Artificial Intelligence: A Modern Approach*. 3rd Edn., Prentice Hall, ISBN-10: 0136042597, pp: 1152.
- Saleet, H., O. Basir, R. Langar and R. Boutaba, 2010. Region-based location-service-management protocol for VANETs. *IEEE Trans. Vehicular Technol.*, 59: 917-931. DOI: 10.1109/TVT.2009.2033079
- Singh, R.P. and A. Gupta, 2011. Information dissemination in VANETs using zone based forwarding. Proceedings of the IFIP Wireless Days, Oct. 10-12, IEEE Xplore press, Niagara Falls, Canada, pp: 1-3. DOI: 10.1109/WD.2011.6098138
- Soua, A. and H. Afifi, 2013. Adaptive data collection protocol using reinforcement learning for VANETs. Proceedings of the 9th International Wireless Communications and Mobile Computing Conference, Jul. 1-5, IEEE Xplore Press, Sardinia, pp: 1040-1045. DOI: 10.1109/IWCMC.2013.6583700
- Soua, A., W. Ben-Ameur and H. Afifi, 2012. Broadcast-based directional routing in vehicular ad-hoc networks. Proceedings of the 5th Joint IFIP Wireless and Mobile Networking Conference, Sept., 19-20, IEEE Xplore Press, Bratislava, pp: 48-53. DOI: 10.1109/WMNC.2012.6416146
- Watkins, C.J.C.H. and P. Dayan, 1992. Q-learning. *Mach. Learn.*, 8: 279-292. DOI: 10.1007/BF00992698
- Yu, B., C. Xu and M. Guo, 2010. Adaptive forwarding delay control for VANET data aggregation. *IEEE Trans. Parallel Distributed Syst.*, 23: 11-18. DOI: 10.1109/TPDS.2011.102