World Wide Web (2020) 23:2303–2322
https://doi.org/10.1007/s11280-019-00749-y

Towards a smarter directional data aggregation in VANETs

Sabri Allani[1,2] · **Taoufik Yeferny**[2,3] · **Richard Chbeir**[1] · **Sadok Ben Yahia**[2,4]

Received: 17 December 2018 / Revised: 9 May 2019 / Accepted: 2 October 2019 /
Published online: 12 March 2020

Abstract

In the last decade, Vehicular Ad hoc NETworks (VANETs) have attracted researchers, automotive companies and public governments, as a new communication technology to improve the safety of transportation systems aiming at offering smooth driving and safer roads. In this respect, a new Traffic Information System (TIS) has benefited from VANET services. The ultimate goal of a TIS consists in properly informing vehicles about road traffic conditions in order to reduce traffic jams and consequently CO2 emission while increasing the user comfort. To fulfil these goals, traffic information data or Floating Car data (FCD) must be efficiently exchanged between mobile vehicles by avoiding as far as possible the broadcast storm problem. In this respect, data aggregation appears as an interesting approach allowing to integrate FCD messages to generate a summary (or aggregate), which undoubtedly leads to reduce network traffic. We introduce, in this paper, a new data aggregation protocol, called Smart Directional Data Aggregation (SDDA). The main idea behind our SDDA protocol is to select the most pertinent FCD messages that must be aggregated. To this end, we rely on three filters: The first one is based on the vehicle's directions. Indeed, every vehicle aggregates only FCD messages corresponding to its direction. Furthermore, it stores, carries and forwards uninteresting data. The second one is carried out by using road speed limitation. The third one relies on a suppression technique to remove duplicated FCD messages. Interestingly enough, our protocol works properly in both highway and urban conditions. The performed experiments show that SDDA outperforms the pioneering approaches of the literature in terms of effectiveness and efficiency.

Keywords VANET · Traffic information system (TIS) · Aggregation · Direction · Speed · Floating car data (FCD)

This article belongs to the Topical Collection: *Special Issue on Smart Computing and Cyber Technology for Cyberization*
Guest Editors: Xiaokang Zhou, Flavia C. Delicato, Kevin Wang, and Runhe Huang

✉ Sabri Allani
sabri.allani@univ-pau.fr

Extended author information available on the last page of the article.

1 Introduction

In the last decade, Vehicular Ad hoc NETworks (VANETs) have attracted researchers, automotive companies and public governments as a new communication technology to improve the safety of transportation systems aiming at offering smooth driving and safer roads. In fact, VANET has been adopted and supported by pioneering transportation agencies and automotive companies (e.g., the US Department of Transportation (DoT), Toyota and Honda, etc.). This support has participated to the emergence of new ideas for VANET-based applications related to entertainment, safety and non-safety information. Worthy mentioning that VANET also supports two communication models namely Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) [1]. These latter are based on the DSRC standard, which uses the dedicated frequency spectrum 75 MHz [4]. In fact, all VANET applications must share this allocated bandwidth.

Currently, most VANET applications rely on V2V communications since no costly infrastructure is needed. However, as VANET is a very dynamic network due the high mobility of vehicles, information must be exchanged between mobile vehicles in an efficient way by avoiding as far as possible the broadcast storm problem [6].

Indeed, the latter occurs whenever a huge number of vehicles broadcast messages in the same time leading to a network saturation, packet delay and collision issues. In this respect, data aggregation appears as an interesting approach allowing to integrate several data about similar events to generate a summary (*aka* aggregate) leading to reduce network traffic. Therefore, the design of an efficient data aggregation approach that combines correlated traffic information or Floating Car Data (FCD) is still a challenging issue. The data aggregation process has to deal with the following compelling challenges: *i*) which FCD message to aggregate? *ii*) when to aggregate data? *iii*) how to deal with information got from far vehicles? *iv*) how to separate out unneeded or duplicated FCD messages? and *v*) how to consider speed limitations and road traffic signals?

The literature witnesses a wealthy number of data aggregation approaches [11, 14, 22, 26]. Nevertheless, within highly complex urban and highway networks, an overwhelming traffic information data needs efficient selection criteria and smart filtering before the aggregation process. Besides that, all existing approaches mainly have been chiefly focused on combining correlated items. However, none of them tried to use a filtering technique to remove duplicated messages. Furthermore, they do not consider the vehicle directions and road speed limitations in the provided traffic information. To fulfil the previous requirements, we introduce here a new data aggregation protocol, called *Smart Directional Data Aggregation* (SDDA). Our protocol works properly in both highway and urban conditions. The main idea behind our SDDA protocol is to select the most appropriate FCD messages that have to be aggregated. To this end, we rely on three filters: The first one is based on the vehicle's directions. Indeed, every vehicle aggregates only FCD messages corresponding to its direction. Furthermore, it stores, carries and forwards uninteresting data. The second one is carried out by considering the road speed limitation. Therefore, when an average speed of received FCD messages surpasses the maximal permitted speed, it will be brushed aside, hence being replaced by the maximal permitted speed. Accordingly, there will be an increase in the aggregation accuracy. The third one relies on a suppression technique [24] to remove duplicated FCD messages. Indeed, we adapted the Slotted 1-persistence technique proposed in [24], in which different waiting time slots are assigned to the neighbouring vehicles depending on their locations. A shorter waiting time is assigned to the vehicles located in the furthest region from the broadcaster. Upon receiving a message, the receiver checks the packet ID and rebroadcasts it with a probability 1 at the assigned time slot. In the case, it receives the packet for the first time and has not received previously any duplicates before its assigned time slot; otherwise, it discards the packet.

The remainder of this paper is organised as follows. Section 2 thoroughly describes the existing approaches in the goal of highlighting their drawbacks regarding the aforementioned challenges. The description of the SDDA protocol is thoroughly detailed in Section 3. The penultimate section presents the simulation settings and the evaluation of our protocol. The results of the experimental results are also shown in this section. Section 5 concludes the paper and pins down some future directions.

2 Related work

In large cities, a huge amount of traffic information, that swiftly exceeds reasonable limits, can be gathered and exchanged between vehicles in motion. Consequently, an appropriate aggregation mechanism can reduce the communication cost while obtaining useful aggregated information. In the literature, data aggregation has been used in different ITS applications, therefore several related approaches have been proposed. Table 1 glances some applications of data aggregation and their related data.

In [23], the authors put forward Self-Organising Traffic Informational Systems (SOTIS) where vehicles periodically exchange their speed and position on the road. Each vehicle in SOTIS computes an average speed of neighbouring vehicles then rebroadcasts the aggregated speed. The main SOTIS limitation was related to the naive rebroadcast of the Floating Car Data (FCD), since it did not ensure or even indicate how duplicated [FCD] messages that came from the same road segment could be aggregated together, which leads to reduce the accuracy of the final aggregated records.

Traffic-View is another worth of mention approach described in [16]. It proposes a similar approach based on broadcasting beacons that contained the FCD, like the average speed of a current road segment and its traffic density. The main difference between SOTIS [23]

Table 1 Data aggregation use cases at a glance

Applications	Reference	Aggregation protocol	Events	Values	Mentioned in literature
Traffic information system	[10] [11] [16]	SOTIS TrafficView Cascada	Traffic jam	Speed and positions known as Flooding Car Data (FCD)	Very Often
Weather information systems	[9] [5]	Reconfigurable vehicle controller	Ice rain etc.	Average temperature, visibility, of road rain degree	Sometimes
Road condition warnings	[19] [15] [8]	Fusion and similarity functions	Broken road, icy road, etc.	Road id Location, condition and street address	Sometimes
Parking Spaces	[25] [13]	ADD spatio-temporal aggregation	Available parking space	Number and location of free parking spaces	Often
Trip Travel Time Prediction	[8] [7]	Spatio temporal Information Retrieval	Trip paths and time	Travel time and short paths	Sometimes

and Traffic-View was the computation of the average speed, since the aggregation process in Traffic View was an accumulative of average speeds in the road (starting from the quickest vehicle to the slowest one). On the contrary, within SOTIS, a vehicle in the centre could aggregate all neighbouring vehicular's data within its range. It was blatant that the aggregated record in Traffic View consisted of just one time-stamp value, one position and one speed, as well as vehicular IDs. Hence, this would use better the bandwidth when transmitting messages to every individual vehicle.

CASCADE was suggested in [10, 11] as an optimised Traffic View version [16]. It allowed compressing syntactic data in the aim of optimising the use of a wireless channel and at the same time guaranteeing accurate aggregated information. Furthermore, it would divide a road into 12 rows (16m × 126m) of a cluster leading to a 1.5 km visibility (named a *local view*). Thus, when vehicles in a local view cluster shared their FCD messages with another cluster, they had an extended view of the whole road segment.

Tsai et al. introduced in [22] a hybrid Aggregating Data Dissemination (ADD) approach that combined both the V2V and V2I models. This approach aggregates the number of available free parking spaces in a big region. To do that, the author split a map into a grid structure of square regions. The geodesic distance between these regions was in fact the Road Side Unit (RSU) communication range. The author defined four data aggregation levels in every region. In the first aggregation level, each vehicle would send its parking place, id, position and speed to the RSU centre of the region. The RSU centre would aggregate all received data in the second level before rebroadcasting it to all vehicles in the region. In the third level, the vehicles in the extreme regions would share their traffic information, with the RSU sink. Finally, this later would aggregate, in the fourth level, all information coming from various regions before rebroadcasting it.

Kumar and Dave introduced in [14] a new multi-criteria decision-making for data aggregation. As a matter of fact, the proposed approach assisted a vehicle to decide about the relevance between data, for instance vehicle speed, vehicle direction and free parking space. Hence, the suggested system could decide if two or more input data were similar enough (syntactically or semantically) to be aggregated or not. To achieve this, the authors represented the knowledge base as a KD-tree data structure in order to check the relevance between nodes using the graph characteristics. Although interesting, the provided approach had major drawbacks but mainly it considered the location of aggregates and ignored all other properties, e.g. vehicle directions, maximum allowed speed, etc.). Indeed, this aggregation decision would only consider data coming from the same road segment.

Time aggregated graphs were introduced by George et al. in [7]. They allowed formalising the road networks and the spatio-temporal properties of the road as a graph data model which would generally support shortest-path query graph algorithms. These time-aggregated graphs could annotate the properties of edges and nodes with the intervals during the time of vehicle presence. However, to decide whatever multiple items could be aggregated or not, the spatio-temporal model was only based on the vehicle travel time, the event life time, and the location. Thus, the major drawback of such model was that one could not exactly assess the vehicle travel time, since it could change its direction at any time.

In the same vein, Zekri et al. introduced in [25] an aggregation structure for events produced and exchanged in vehicular networks. More precisely, the proposed structure was based on a spatio-temporal model having two levels. The first one is a physical level consisting of a repository shared between all vehicles to share information without loss. The second one was a logical level where each driver would define his/her preferences for what information (s)he was interested in. This model would manipulate the same shared knowledge base between all vehicles. The ability of being able to guarantee loss-less exchanged information was a significant characteristic of their data structure. In addition, the storage space, mainly needed for the aggregation structure, was particularly limited to include an acceptable number of temporal dimensions. The main limitation of this solution was the maintenance and the privacy issues of such a shared knowledge base.

In [17], the aggregation of the FCD was only based on the geographical characteristics of the area. For that purpose, the authors introduced the Region-based Location protocol Service Management Protocol (RLSMP) which aimed to reduce the updated positions as well as reducing the number of messages generated to locate car positions. Although the aggregation solution clearly reduced the network overload, it resulted in: *i*) more packet collisions and consequently more re-transmissions essentially due to the fact that the exchanged packets had a large size; and *ii*) longer delays due to the processing carried out on the data.

In [9], authors introduced a re-configurable model of autonomous vehicle that can estimate and adapt to the behavioural of the driver in real-time. Nonetheless, in real conditions, vehicles have only a local information about their neighbours. Thus, the introduced model exploited the available knowledge of the system to have a similar behaviour to that achieved by human drivers, depending on the context. The results showed that this re-configurable model maintains a high degree of traffic quality, efficiency and safety under the variation of weather conditions.

Santamaria et al. [18] introduced a new multi-layered architecture for an efficient distribution of traffic task management. Indeed, the first layer is composed of On-Board Units (OBUs) and RoadSide Units (RSUs). The second, includes Inter Vehicle Communication (IVC) protocols. The third layer is the CPU processing nodes, that collects data from RSUs then aggregates and disseminates it to the Intelligent Transportation System management system. The proposed approach combines traffic information with other information like road conditions as well as weather.

Singh et al. [19] introduced a new information dissemination scheme based on the fuzzy logic. The introduced scheme was composed of four tasks: decision, fusion, aggregation and dissemination. Indeed, the main contribution consists on the definition of a fuzzy logic function used within the fusion task.

To sum up, the aggregation process in the aforementioned approaches operates through the following three phases:

1. The decision phase, where decision regarding the selection of data items to be merged is made;
2. The fusion phase, which is related to the function of fusion. Therefore, all the similar data will be merged in one record;
3. The dissemination phase, in which aggregated data is broadcasted to other vehicles.

With respect to these phases, a scrutiny of the existing aggregation schemes puts highlights on several limitations:

- **Security** : If the aggregation has the ability to decrease bandwidth consumption problems, it might make security issues harder to manage (e.g., the encryption and the decryption of multiple aggregated and compressed packets) [15];
- **Scalability**: The existing schemes have medium aggregation time as well as low scalability. This is due to the fact that when the number of the duplicated exchanged messages goes up, the number of collision problems rises as well [14];
- **Genericity**: Only few approaches [14, 22] have proposed a generic model for both aggregation and dissemination mechanisms. In fact, the combination and synchronisation of both aggregation and dissemination mechanisms are of paramount importance to avoid the broadcast storm problem; [24] and to decrease the network overhead;

– **Filtering**: The input items are not fully filtered out. Indeed, many duplicated items and irrelevant items are not neglected and consequently leading to a high level of network overload.

Table 2 shows at a glance the existing protocols and highlights the key differences between them and SDDA: our introduced protocol that we thoroughly describe in the following section.

3 SDDA: Smart directional data aggregation protocol

Several concepts and definitions are presented in the next subsection before the description of our aggregation protocol.

3.1 Preliminaries

Definition 1 Lane (l): It is a one-way path having a paved surface which connects two spatial points on the map. Formally, l:⟨ Id, S, E, $\overrightarrow{SE}$, Speed$_{min}$, Speed$_{max}$, Status, Location ⟩ where:

– Id: represents the identifier of the lane;
– S and E: are the start and end points connected by the lane, respectively. Every point is represented by spatial coordinates (e.g., (x,y));
– $\overrightarrow{SE}$: is the directed vector segment from the point S to E;
– Speed$_{min}$ and Speed$_{max}$ are respectively the minimal and maximal speed limits;

Table 2 Comparison between SDDA and other aggregation protocols

Criteria	Message structure	Aggregated data	Duplicated data
SOTIS [23]	Vehicle Id, Speed position	Average speed	No filter
TrafficView [16]	Vehicle Id, Speed position	Average speed	No filter
CASCADE [10, 11]	Vehicle Id, Speed position	Average speed	No filter
SDDA	Vehicle Id, Speed, position and direction	Average speed	Suppression technique, direction filter and speed limitation filter
ADD [22]	Road events, Speed, position safety events	Safety and non safety events	No filter
Autonomous vehicle [9]	Position, Weather conditions Road conditions	Weather and road conditions	No filter
Multi-layered aggregation architecture [18]	Vehicle Id, Speed, position and road conditions	Average speed and road conditions	No filter

- Status: is the lane situation (for example: restricted, open, closed, etc.);
- Location: is the geographical coordinates of the lane on the map. ⧫

Definition 2 Road (r): It has no less than one or several lanes with similar and dissimilar directions. Formally, r: ⟨ Id, L, Type, Name, Network Coverage ⟩ where:

- Id: is the road identifier;
- L: the lanes set of the road;
- Type: is used to indicate whether it is a street, urban, highway, etc;
- Name: is used for the description of the road (e.g., street name);
- Network Coverage: indicates the types of network covered within the road (for instance 3G, 4G, WIFI, GSM, etc.). ⧫

Definition 3 Vehicle (v): It is defined in our approach as follows: v:⟨ Id, Driving, Speed_{max}, Positioning System, Brand, Type, Size, Environment, dedicated short-range communications (DSRC) Range, Destination ⟩ where:

- Id: is the vehicle identifier;
- Driving: refers to the set of the driving settings (e.g., Preferred path, Deriving mode: Economic/sport, etc.);
- Speed_{max}: represents the maximal speed of the vehicle;
- Positioning system: is the geographical GIS-system used by the car (Bing Maps, StreetMaps, Google Maps, ...);
- Brand: indicates to the vehicle manufacturer of the vehicle (for example, Toyota, Jeep, BMW,...);
- Type: is used for the indication of the car type (for instance, light truck, sport, mini-van,...);
- Size: refers to the vehicle size;
- Environment: indicates geographical location, weather, and vehicle speed;
- DSRC Range: is the signal power of dedicated wireless short-range communication technology (e.g., 300m, 400m, etc.);
- Destination: is the location to which a vehicle travels.

A vehicle can perform three actions:

- **Broadcast**: A vehicle disseminates FCD messages using the suppression broadcasting technique defined in [24];
- **StoreCarryandForward**: It can store, carry and then rebroadcast the same message using the rebroadcasting technique defined in [24];
- **CalculateAverageSpeed**: It computes the average speed using the aggregation function defined in Section 3.2;
- **Receive**: It receives all types of sent messages using the DSRC protocol [4];
- **CalculateDirection**: It can determine its direction via a positioning system (e.g., Google Maps) based on its location and its destination;
- **LocateLane**: It can locate the current driving lane based on its geographic location. ⧫

It is acquired that just one type of messages has the ability to be generated and sent, namely the Floating Car Data (FCD) known also as Floating Cellular Data [23].

Definition 4 FCD message (f): The adopted FCD messages header structure is therefore defined as a 4-tuple: f: ⟨ SenderId, SenderPosition, AverageSpeed, Destination ⟩ where:

- SenderId: is the unique identifier of vehicle that sends the message;
- SenderPosition: contains the spatial position of the sender;
- AverageSpeed: is the average speed (computed using a function defined in Section 3.2);
- Destination: contains the future location of the vehicle sending the FCD message. ⧫

The message size is less than 2,321 bytes, which is the maximal allowed size as defined by 802.11p standard [4].

3.2 Aggregation protocol

Our protocol considers three scenarios: unidirectional road, bidirectional road, and an urban scenario. The aggregation function of use is the same average function used in SOTIS [23]. We have opted for the main average speed aggregation function of SOTIS as our contribution is in fact considered as an optimisation of SOTIS. On the other hand, only three approaches, CASCADE [10], Traffic-View [16] and SOTIS [23], as mentioned in the related work, focus on the vehicle-speed aggregation without any combination of the other driver preferences. The average speed aggregation function of SOTIS is defined as follows [23]:

$$\hat{V}_{r,new} = \hat{V}_{r,prev} + \hat{V}_r \tag{1}$$

where $\hat{V}_{r,new}$ is the new average speed for the road r, $\hat{V}_{r,prev}$ is its previous average speed, and $\hat{V}_r$ stands for the average speed of the vehicles on the road r. Each vehicle has three aggregation cases: unidirectional-road, bidirectional-road and urban-city. Algorithm 1 illustrates the behaviour of a vehicle v upon receipt of an FCD message.

3.2.1 Unidirectional road case

As depicted in Figure 1, unidirectional road contains more than one lanes having different speed limitations. It is important to note that in unidirectional road, all vehicles drive in the same direction. Therefore, vehicles ahead must collect traffic information, aggregate and disseminate it to other vehicles located behind.

Algorithm 1 Aggregation algorithm of a vehicle v.

```
VehilceLocation ← v.Environment.Location
switch VehilceLocation do
    case Unidirectional Road
        Use Algorithm 2
    case Bidirectional Road
        Use Algorithm 3
    case Urban City
        Use Algorithm 4
```

To deal with this scenario, we rely on Algorithm 2. Briefly, when a vehicle v receives an FCD message, it checks whether it comes from a farther vehicle on the same road. If it is

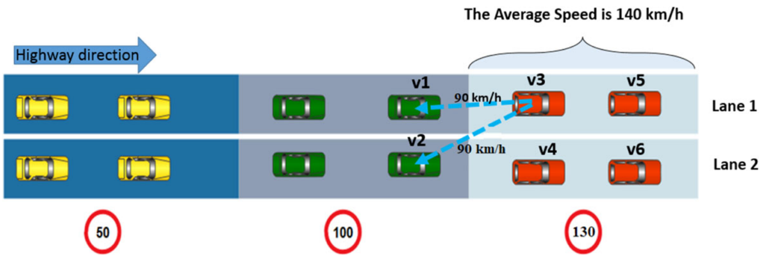


Figure 1 Unidirectional road case

the case, then it computes the new updated average speed $\hat{V}_{r,new}$ within (1). If the received average speed is greater than the maximum allowed speed of the current road r, then the vehicle receiving the message will keep its previous average speed $\hat{V}_{r,prev}$ and rebroadcast the value of the road maximum speed. In Figure 1, the FCD message sent by vehicle v_3, will be only received by vehicles v_1 and v_2, since they are in the broadcast range of v_3 and drive behind it. However, the average speed, 140 km/h, will be ignored by vehicles v1 and v2, and they will broadcast the road maximum allowed speed (of 130 km/h). Therefore, all the disseminated traffic information will follow the legal speed.

3.2.2 Bidirectional road case

In our protocol, vehicles moving in opposite directions must collect traffic information, and then aggregate and disseminate it to other opposite vehicles, as drawn in Figure 2. Doing so, this would inform and warn all drivers about the traffic conditions ahead, which leads to avoid traffic jam and road accidents. In fact, the main difference between this case and the previous one is that in a bidirectional scenario, vehicles can accept FCD messages that come form the opposite side or vehicles ahead in the same lane and ignore other messages that come from behind.

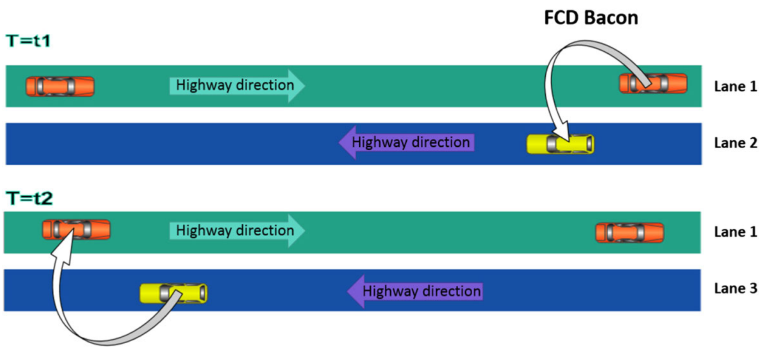


Figure 2 Bidirectional road case

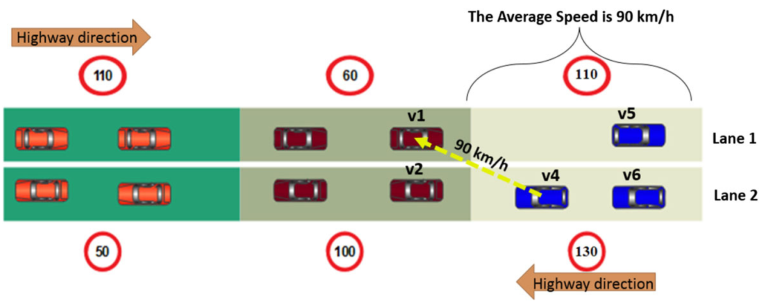


Figure 3 FCD message propagation on a bidirectional road

Algorithm 2 Aggregation algorithm in unidirectional road with speed limitations.

```
P ← v.Environment.Location
VD ← v.Destination
VDirection ← v.CalculateDirection(P,VD)
VSpeed ← v.Environment.Speed
LaneSpeedMax ← v.LocateLane.SpeedMax
procedure VEHICLE ON UNIDIRECTIONAL ROAD(fcd)
    PS ← fcd.SenderPosition
    DS ← fcd.Destination
    FAD ← fcd.getAverageSpeed()
    SenderDirection ← v.CalculateDirection(PS,DS)
    SenderDirection ← v.CalculateDirection(PS,DS)
    if Same(SenderDirection, VDirection) and VSpeed ≤ SenderSpeed then
        if SenderSpeed ≤ LaneSpeedMax then
            AverageSpeed ← v.CalculateAverageSpeed(VSpeed, FAD)
            v.Broadcast (AverageSpeed)
        else
            v.Broadcast (LaneSpeedMax)
        end if
    end if
end procedure
```

Actually, the vehicles in the opposite side have a larger overview on the opposite traffic conditions, since they have been passed in front of it. Doing that allows guaranteeing that the FCD contains an aggregated value of the whole opposite lanes. Pseudo-code of Algorithm 3 deals with this scenario. Figure 3 depicts the propagation of an FCD message on a bidirectional road. Indeed, vehicle v_4 will share its FCD average speed (90 km/h) with vehicle v_1, which will broadcast this information to other vehicles, standing behind, since the average speed is less than the maximum allowed speed.

3.2.3 Urban network case

As depicted in Figure 4, in urban scenario, vehicles move in many different directions where they can meet over cross roads and junctions and thus exchange their traffic information. Using a blind aggregation method, vehicles will aggregate the average speed of other vehicles that are not going to the same direction, which would badly affect the accuracy and the

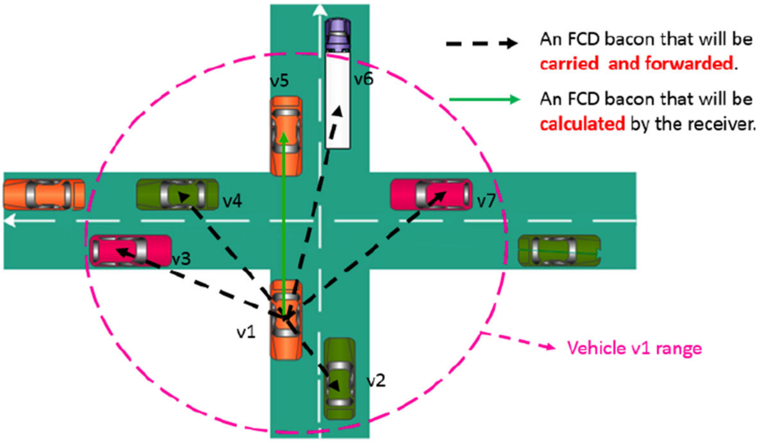


Figure 4 Urban scenario case

precision of the aggregated traffic information. To overcome this issue, in our aggregation model, we rely on the following three filters:

1. A direction filter to check that the FCD messages targeting the same road and direction;
2. A suppression technique filter that ignores all duplicated FCD messages. In our case, we rely on the slotted 1-persistence suppression technique [24];
3. An aggregation filter that ignores all the received FCD messages that exceed the road maximum speed. In fact, if the average speed of any FCD message is greater than the maximum allowed speed, then the vehicle receiving the message will ignore it and broadcasts the maximum speed instead.

Algorithm 3 Aggregation algorithm on a bidirectional road with speed limitations.

```
P ← v.Environment.Location
VD ← v.Destination
VDirection ← v.CalculateDirection(P,VD)
VSpeed ← v.Environment.Speed
LaneSpeedMax ← v.LocateLane().SpeedMax
procedure VEHICLE ON BIDIRECTIONAL ROAD(fcd)
    PS ← fcd.SenderPosition
    DS ← fcd.Destination
    FAD ← fcd.getAverageSpeed()
    SenderDirection ← v.CalculateDirection(PS,DS)
    if NOT Same(SenderDirection , VehicleDirection) then
        if SenderSpeed ≤ LaneSpeedMax then
            AverageSpeed ← v.CalculateAverageSpeed(VSpeed,FAD)
            v.Broadcast (AverageSpeed)
        else
            v.Broadcast (LaneSpeedMax)
        end if
    end if
end procedure
```

Algorithm 4 Aggregation algorithm in urban network.

```
P ← v.Environment.Location
VD ← v.Destination
VDirection ← v.CalculateDirection(P,VD)
VSpeed ← v.Environment.Speed
LaneSpeedMax ← v.LocateLane().SpeedMax
procedure VEHICLE IN URBAN CITY(fcd)
    PS ← fcd.SenderPosition
    DS ← fcd.Destination
    FAD ← fcd.getAverageSpeed()
    SenderDirection ← v.CalculateDirection(PS,DS)
    if Same(SenderDirection , VDirection) then
        if SenderSpeed ≤ LaneSpeedMax then
            AverageSpeed ← v.CalculateAverageSpeed(VSpeed, FAD)
            v.Broadcast (AverageSpeed)
        else
            v.Broadcast (LaneSpeedMax)
        end if
    else
        StoreCarryandForward (fcd)
    end if
end procedure
```

In urban networks, when a vehicle receives an FCD message, it checks firstly the direction of the sender. Consequently, if the received average speed is less than the road maximum speed limit,then the vehicle will aggregate and disseminate the received FCD message. Otherwise, the received message will be ignored.

Indeed, whenever a vehicle v receives a message from another vehicle driving in the opposite direction, it will store, carry, and then forward it to other vehicles that may be going to this direction. Doing so, the traffic information will be disseminated to all vehicles in the road intersections. Algorithm 4 describes a vehicle behaviour upon receiving an FCD message in an urban situation.

4 Experimental evaluation

We present in this section the performance evaluation carried on in order to evaluate our aggregation protocol. For that purpose, both SOTIS [23] and Traffic-View [16] have been opted for as baseline approaches due to the fact that they focused on the aggregated FCD without combining other data (such as the example of safety and non-safety events). It is worth mentioning that CASCADE [11] is not taken into consideration in our evaluation for the reason that it uses on top of Traffic-View a compression algorithm [16] with the aim of optimising the MAC-Layer use (which in fact is out of our study scope; but by way of contrast, our approach can apply it after aggregation). Our experiments have been performed using the Veins simulator framework.[1] Veins is an open source framework. In particular, it is used for Inter-Vehicular Communication (IVC) suitable for the combination of one

[1] http://veins.car2x.org/

(a) Corniche

(b) Zarzouna

(c) Menzel Bourguiba

Figure 5 Example of different road topologies

event-based network simulator and one road traffic micro-simulation model. In the goal of carrying out the experiments, for our performance evaluation, we select real-world road topologies from three cities at the governorate of Bizerte in the north of Tunisia. We consider the three road topologies depicted in Figure 5, representing portions of the urban areas of Corniche, Zarzouna and Menzel Bourguiba cities. Table 3 summarises the characteristics of each map. The amount of vehicles operating and travelling upon the map varies between 200 and 1000, with a range of low to highly traffic.

Table 3 Road topologies characteristics

Map	Number of roads	Number of junctions	Dimensions
Corniche	503	214	2.2km x 2.5 km
Zarzouna	681	312	2.2km x 2.5 km
Menzel Bourguiba	919	468	2.2km x 2.5 km

Table 4 Simulation settings

Physical layer	Frequency band	5.9 GHz
	Transmission power	30 mW
	Bandwidth	10 MHz
Link Layer	Bit rate	6 Mbit/s
	CW	15.1023
	Slot time	13 us
	SIFS	32 us
	DIFS	58 us
	Routing protocol	AODV/GPSR
Scenarios	message size	2312 Bytes
	Message frequency	0.5 Hz
	#Runs	30 times

4.1 Experimental settings

The road traffic simulation has been carried out by SUMO [12], whereas the network simulation has been done using OMNeT++ [2] along with the physical layer modelling toolkit MiXiM,[2] which has allowed employing accurate models for radio interference. In addition to that, using static and moving obstacles has been shadowed. Through these two well-established simulators, it is noted that nodes simulated by OMNeT++ 5.1.0 have the ability to interact with SUMO for the purpose of simulating IVC influence upon on road traffic and mobility. We take advantage of both veins simulators with the objective of providing realistic models for 802.11p MAC, PHY and DSRC layers. The MAC and PHY parameters are defined in agreement with those basic specifications of the 802.11p standard defined in [21]. The simulation settings are summarised in Table 4. We set the vehicle transmission power in the MAC layer to 30mW with the target of achieving 300m of interference range, approximately.

4.2 Evaluation metrics

The assessment of the performances of our protocol is carried out through the following metrics:

1. **Overload:** It stands for the total number of sent packets. Interestingly enough, the ultimate goal of any aggregation protocol is to avoid the overload problem [4] by looking for minimising the number of messages exchanged in the network. The average overload is defined as follows:
$$Overload = \frac{\sum sent\ \ packet}{\sum vehicle} \tag{2}$$
2. **Co2Emission:** It refers to the amount of Co2 getting out from vehicles and affecting the environment [3]. Indeed, the ultimate goal of any traffic information system, and especially any aggregation protocol, is to reduce the total Co2 emission [3] by looking for minimising as much as possible traffic jams and decreasing the active waiting time

[2] http://mixim.sourceforge.net/

of vehicles in cross roads and on highways. The average Co2 emission is defined as follows:

$$Co2Emission = \frac{\sum Vehicle\ Co2Emission}{\sum\ vehicle} \quad (3)$$

3. **Latency:** It is the time needed to deliver the aggregated message to an interested vehicle. The average latency, AL, is defined as follows:

$$AL = \frac{\sum(t_{v_i} - T)}{\sum Interested\ \ Vehicle} \quad (4)$$

where t_i stands for the arrival time of the event message to a vehicle v_i, and T is the time-stamp of the event.

4. **FCD duplication ratio:** It is the number of messages that are already sent and aggregated for a given message f. It is defined as follows:

$$Duplication\ \ ratio = \frac{\sum Duplicated\ f}{\sum\ f} \quad (5)$$

5. **Aggregation Precision:** It assesses to what extent our aggregation protocol is able to only aggregate an appropriate FCD message f for a given road r without duplication and to take into consideration the road maximum speed. Hence, the challenge will be to obtain precision values of the average speed propagated to vehicles compared to the real road traffic state. It is defined as follows:

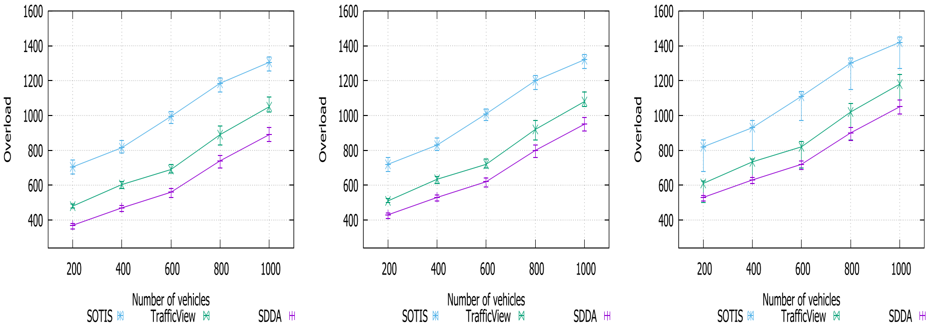
$$Precision(f) = \frac{|IIV|}{|AIV|} \quad (6)$$

where IIV stands for the set of interested informed vehicles (i.e., only appropriate vehicles for a message f), and AIV stands for the set of all informed vehicles that will aggregate the received FCD message f (i.e., interested as well as not interested vehicles for an FCD f). The average precision is defined as follows:

$$AveragePrecision = \frac{\sum Precision(f)}{\sum f} \quad (7)$$

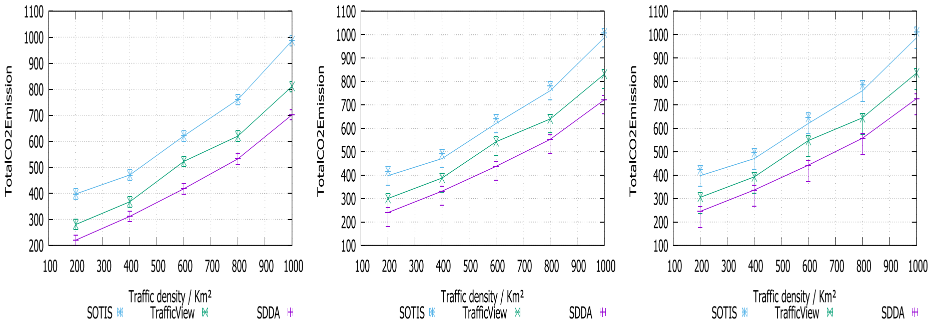
4.3 Results

As expected, using our aggregation protocol, the overload level is decreased (cf. Figure 6). That has been because the duplicated messages number is eliminated and the propagation



(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 6 Variation of average overload values w.r.t. the number of vehicles



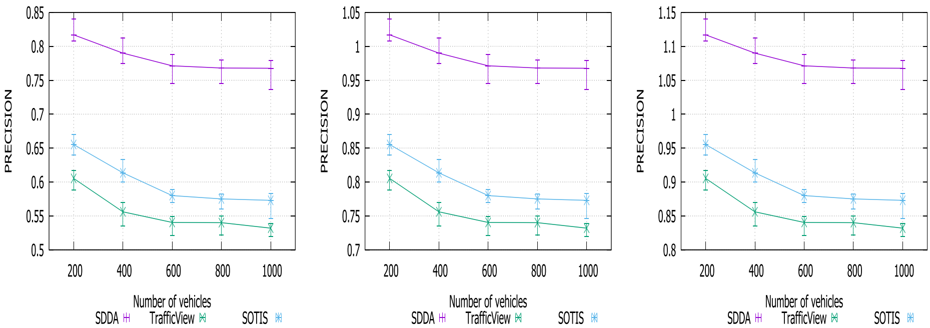
(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 7 Variation of average Co2 emission values w.r.t. the number of vehicles

of unnecessary aggregated FCD messages to uninterested vehicles is decreased. Because of that, our protocol SDDA makes easy to keep a low overload. In addition to that, Figure 7 depicts that the SDDA CO2 emission value is not high within different network densities. As a matter of fact, that our protocol reduces the CO2 emission value when compared with the to SOTIS and Traffic-View by approximately 70% due to the fact that vehicles will have precise traffic data, hence avoiding any traffic jams (Figure 8).

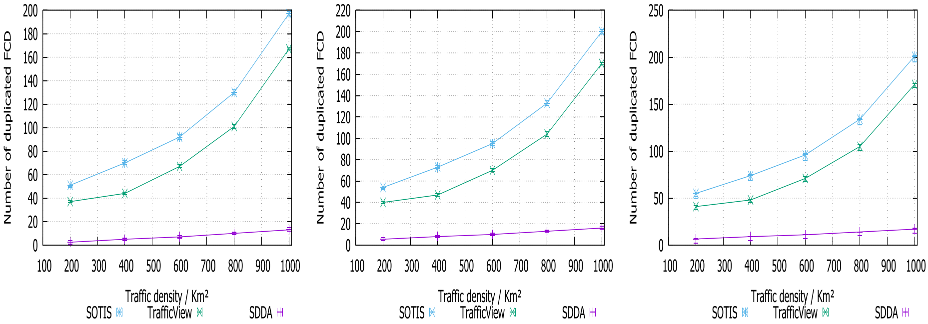
It is demonstrated in Figures 6 and 7 that the overload and CO2 emission values are able to increase for all protocols in the condition that the number of vehicles increases. It is worth mentioning that this interesting performance is because our strategy makes it possible to ignore all the unnecessary FCD messages in the case that vehicles move on the basis of on direction and speed limitation filters. Furthermore, it is indicated in Figure 9 that in our proposed solution, the number of duplicated messages is 99% less in comparison to its competitors. This is owe to a Slotted-1 persistence suppression technique [24] is capable of eliminating all the duplicated messages.

As a last step, Figure 10 shows that the SDDA strategy latency is has been a bit lower than that of its competitors. That has been because the SOTIS and Traffic-View overload level is able to raise the network collisions, thus badly affecting the latency time [20].



(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 8 Variation in average aggregation precision FCD values w.r.t. the number of vehicles



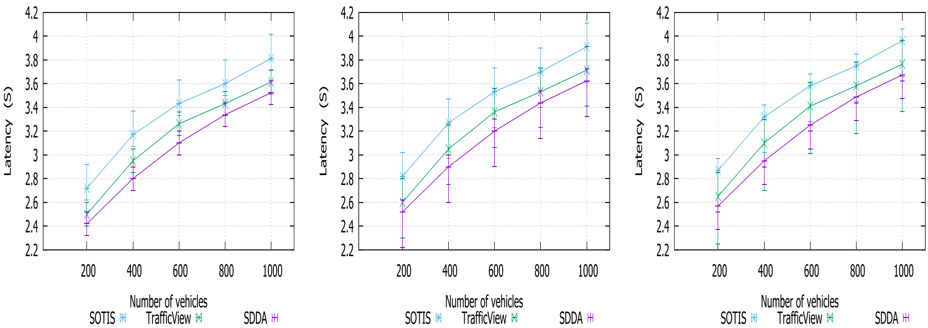
(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 9 Variation in average aggregation number of duplicated FCD values w.r.t. the number of vehicles

According to what is anticipated, the packet loss ratio, which has been illustrated Figure 11 in an inverse manner proportional to the traffic density. As a matter of fact, it falls in relation to the rise in the number of vehicles. Roughly speaking, this can be explained by the fact that the message reception errors and the communication overload will increase in case the number of vehicles goes up within a network, hence resulting in raising the ratio of lost packets. As a consequence, that fact will increase in a certain way the number of vehicles that wont receive the messages or receiving damaged packets (unreadable, corrupted, etc.).

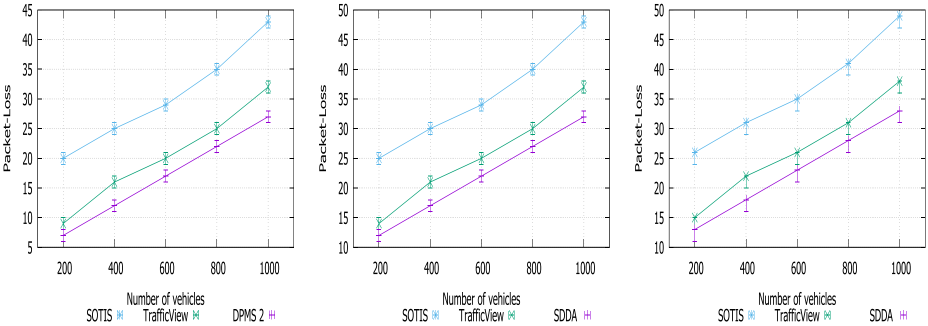
Figure 8, shows a good precision results compared to SOTIS and TrafficView . This is owning to the suppression technique filter that ignores all duplicated FCD messages. Added to that, the direction filter will increase the accuracy the average speed aggregated due to the direction filter. However, the using of a blind aggregation method will affect the precision and the accuracy of the exchanged traffic information.

As a summary, these simulation results show that our strategy has performed well the baseline strategies from various perspectives (latency, packet loss, aggregation precision, CO2 emission, overload). Added to that, this strategy is generic and is capable of being adapted to any aggregation context, such as trip travel time, commercial advertisements, and road conditions.



(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 10 Variation of average latency values w.r.t. the number of vehicles



(a) Corniche (b) Zarzouna (c) Menzel Bourguiba

Figure 11 Variation of the packet loss values w.r.t. the number of vehicles

5 Conclusion

We have introduced in this paper SDDA , which is in fact a generic intelligent directional information aggregation model protocol with the aim of exchanging traffic data in a VANET order that we overcome several limitations as regards existing approaches. In view of fact, the main thrust of this protocol is equivalent to the adequate targeting of the use of a direction and road speed limitation. Accordingly, many objectives have been achieved, namely reaching a high aggregation precision and a low overload ratio. It is demonstrated through extensive experimental work that SDDA has had good results compared to the results provided by pioneering literature approaches. Avenues of future work are as follows:

1. Working on real-word validation scenarios and planning to extend the model to deal with other aggregation issues in VANET, such as trip travel time, road conditions, and parking spaces;
2. Providing a generic compression algorithm for FCD messages to reduce the bandwidth usage;
3. Integrating other dissemination protocols in our aggregation model to provide a more generic aggregation and dissemination protocol.

References

1. Allani, S., Yeferny, T., Chbeir, R., Yahia, S.B.: Dpms: A swift data dissemination protocol based on map splitting. In: Proceedings - international computer software and applications conference, vol. 1, pp. 817–822 (2016)
2. Andras, V.: Omnet++. In: Wehrle, K., Gunes, M., Gross, J. (eds.) Modeling and tools for network simulation, pp. 4–60 (2010)
3. Cappiello, A., Chabini, I., Nam, E.K., Lue, A., Abou Zeid, M.: A statistical model of vehicle emissions and fuel consumption. In: IEEE 5th conference on intelligent transportation systems, pp. 801–809 (2002)
4. Commission, F.C.: Amendments of the commission's rules regarding dedicated short-range communication services in the 5.9 ghz band. Federal Communications Commission Proceedings (2004)
5. Duan, X., Liu, Y., Wang, X.: Sdn enabled 5g-vanet: Adaptive vehicle clustering and beamformed transmission for aggregated traffic. IEEE Commun. Mag. **55**(7), 120–127 (2017)
6. Feukeu, E., Zuva, T.: Overcoming broadcast storm problem in a vehicular network. In: 2017 13th international conference on signal-image technology internet-based systems (SITIS), vol. 13, pp. 402–407 (2017)

7. George, B., Kim, S.: Spatio-temporal networks. Database Management and Information Retrieval (2013)
8. George, B.: Shekhar S. Springer, Time-aggregated graphs for modeling spatio-temporal networks (2008). Berlin
9. Horcas, J.M., Monteil, J., Bouroche, M., Pinto, M., Fuentes, L., Clarke, S.: Context-dependent reconfiguration of autonomous vehicles in mixed traffic. Journal of Software: Evolution and Process **30**, 1–15 (2018)
10. Ibrahim, K.: Data aggregation and dissemination in vehicular ad-hoc networks by data aggregation and dissemination in vehicular ad-hoc networks. 4 Old Dominion University, pp 1–120 (2011)
11. Ibrahim, K., Weigle, M.C.: Cascade: Cluster-based accurate syntactic compression of aggregated data in vanets. 2008 IEEE Globecom Workshops, GLOBECOM 2008 pp 1–10 (2008)
12. Kaisser, F., Gransart, C., Berbineau, M.: Simulations of vanet scenarios with opnet and sumo. Springer, Berlin (2012)
13. Kumar, R., Dave, M.: Knowledge based framework for data aggregation in vehicular ad hoc networks. Computational Intelligence and Information Technology, pp 722–727 (2011)
14. Kumar, R., Dave, M.: A framework for handling local broadcast storm using probabilistic data aggregation in vanet. Wirel. Pers. Commun. **72**, 315–341 (2013)
15. Mohanty, S., Jena, D.: Secure data aggregation in vehicular-adhoc networks: A survey. 2nd International Conference on Communication. Computing and Security ICCCS-2012 **6**, 922–929 (2012)
16. Nadeem, T., Dashtinezhad, S., Liao, C., Iftode, L.: Trafficview: Traffic data dissemination using car-to-car communication. ACM SIGMOBILE Mobile Computing and Communications Review **3**, 1–8 (2004)
17. Saleet, H., Langar, R., Naik, K., Boutaba, R., Nayak, A., Goel, N.: Intersection-based geographical routing protocol for vanets: A proposal and analysis. IEEE Trans. Veh. Technol. **60**, 4560–4574 (2011)
18. Santamaria, A.F., Tropea, M., Fazio, P.P., Raimondo, P., Rango, F.D., Voznak, M.: A decentralized its architecture for efficient distribution of traffic task management. In: 2018 11th IFIP wireless and mobile networking conference (WMNC), vol. 11, pp. 1–5 (2019)
19. Singh, J., Singh, K.: Advanced vanet information dissemination scheme using fuzzy logic. In: 2018 IEEE 8th annual computing and communication workshop and conference (CCWC), vol. 8, pp. 874–879 (2018)
20. Sommer, C., Dressler, F.: Information Dissemination in Vehicular Networks (2015)
21. Sommer, C., German, R., Dressler, F.: Bidirectionally coupled network and road traffic simulation for improved ivc analysis. IEEE Trans. Mob. Comput. **10**, 3–15 (2011)
22. Tsai, H.W.: Aggregating data dissemination and discovery in vehicular ad hoc networks. Telecommun. Syst. **50**, 285–295 (2013)
23. Wischoff, L., Ebner, A., Rohling, H., Lott, M., Halfmann, R.: Sotis - a self-organizing traffic information system. In: The 57th IEEE semiannual vehicular technology conference, 2003. VTC 2003-Spring., pp 1–6 (2003)
24. Wisitpongphan, N., Tonguz, O., Parikh, J., Mudalige, P., Bai, F., Sadekar, V.: Broadcast storm mitigation techniques in vehicular ad hoc networks. IEEE Wireless Commun. **14**(6), 84–94 (2007)
25. Zekri, D., Defude, B., Delot, T.: Building, sharing and exploiting spatio- temporal aggregates in vehicular networks. Hindawi Mobile Information Systems **14**, 259–285 (2014)
26. Zhu, Y., Zhao, Q., Zhang, Q.: Delay-constrained data aggregation in vanets. IEEE Trans. Veh. Technol. **64**, 2097–2107 (2015)

Affiliations

Sabri Allani[1,2] · **Taoufik Yeferny**[2,3] · **Richard Chbeir**[1] · **Sadok Ben Yahia**[2,4]

Taoufik Yeferny
taoufik.yeferni@nbu.edu.sa

Richard Chbeir
rchbeir@acm.org

Sadok Ben Yahia
sadok.ben@taltech.ee

1 University Pau & Pays Adour, UPPA E2S, LIUPPA, Anglet, France

2 LIPAH-LR 11ES14, University of Tunis El Manar, 2092 Tunis, Tunisia

3 College of Science, Northern Border University, Arar, Saudi Arabia

4 Department of Software Sciences Akadeemia tee 15a, Tallinn University of Technology, 12618 Tallinn, Estonia