

DAPF: delay-aware packet forwarding for driving safety and efficiency in vehicular networks

ISSN 1751-8628
 Received on 26th April 2019
 Revised 29th August 2019
 Accepted on 7th January 2020
 doi: 10.1049/iet-com.2019.0427
 www.ietdl.org

Hamayoun Shahwani¹, Bien Aime Mugabarigira², Yiwen Shen², Jaehoon (Paul) Jeong³ ✉, Jitae Shin⁴

¹Department of Telecommunication Engineering, Faculty of Information and Communication Technology, BUITEMS, Quetta, Pakistan

²Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon, Republic of Korea

³Department of Interaction Science, Sungkyunkwan University, Suwon, Republic of Korea

⁴College of Information and Communication Engineering, Sungkyunkwan University, Suwon, Republic of Korea

✉ E-mail: pauljeong@skku.edu

Abstract: This study proposes an effective delay-aware packet forwarding (DAPF) for driving safety and efficiency in vehicular networks. Vehicular ad hoc networks have been an emerging technology for vehicular communication for the last few decades, but still, it has many challenging issues such as on-time dissemination of message at an emergency situation (e.g. accident and obstacle) to the vehicles having the same route to their destinations. This on-time dissemination can prevent further collision of vehicles and road traffic congestion. In this study, the authors propose an effective way of selecting the processing position of a message among a cluster head, road-side unit (RSU), and vehicular cloud, on the basis of total delivery time and cost. They further show that this effective selection and on-time dissemination helps the upcoming vehicles to select an appropriate route to their destinations. Through simulation results, it is shown that their DAPF outperforms other schemes in terms of packet delivery time.

Nomenclature

| | |
|-----------------|---|
| T_{Pro} | processing time |
| $T_{Pro_{ch}}$ | processing time of cluster head (CH) |
| $T_{Pro_{rsu}}$ | processing time of road-side unit (RSU) |
| $T_{Pro_{cld}}$ | processing time of cloud |
| T_{Com} | communication time |
| P_L | packet length |
| R | transmission rate |
| C_{Pro} | processing cost |
| $C_{Pro_{ch}}$ | processing cost at CH |
| $C_{Pro_{rsu}}$ | processing cost at RSU |
| $C_{Pro_{cld}}$ | processing cost at cloud |
| C_e | estimated cost |
| T_{ep} | estimated processing time |
| T_{cp} | actual processing time |
| D_{Pro} | propagation delay |
| d | distance between links |
| s | propagation speed |
| T_{tot} | total delivery time |
| $T_{tot_{ch}}$ | total delivery time for CH |
| $T_{tot_{rsu}}$ | total delivery time for RSU |
| $T_{tot_{cld}}$ | total delivery time for cloud |
| T_v | vehicle's time to reach a junction |
| C_{Pro} | lowest processing cost |
| T_{Te} | vehicle's estimated travelling time |
| T_{Tn} | vehicle's new travelling time |

1 Introduction

Vehicular ad hoc networks (VANET) for vehicular communication or connected cars have been a hot topic for researchers over the last few decades. IEEE has standardised dedicated short-range communications (DSRC) for vehicular networks. Although the basic purpose of VANET was to reduce road accidents, traffic congestion, and fuel consumption [1, 2], VANET can also provide infotainment services to drivers [3–5]. The three categories

of VANET have been researched as vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and infrastructure-to-vehicle (I2V) communications.

Classification of autonomous vehicles is done in the following five levels [6]. Vehicles with no automation are classified as level 0. Vehicles with limited automation are classified as levels 1 and 2. Vehicles with full automation and a limited safe environment are classified as levels 3 and 4. Fully automated vehicles without human intervention are classified as level 5.

The fully automated vehicle is the one that can move from one point to another without human intervention. A fully automated vehicle must be equipped with tens of devices such as radars, lidars, cameras, and global positioning systems (GPSs). With the combination of all these devices, the fully automated vehicle can recognise its surroundings and determine its path. The advantages of fully automated vehicles include the prevention of accidents and collision, congestion-free traffic flow, reduction in fuel consumption, and self-driving for physically handicapped persons.

Why do we need autonomous vehicles? The answer is to overcome the accidents caused by human errors. The three main error types made by a human while driving are as follows [7]:

- *Perceptual error:* This error is caused by the mis-perception of the driver due to dim light or bad weather.
- *Distraction error:* This error is caused when a driver is distracted by something (e.g. smartphone call and texting) and fails to notice the upcoming moment of danger. This error is also known as 'blindness due to inattention'.
- *Response error:* This error is caused actually by a driver who has full attention on the situation during driving, but he fails to react correctly in a difficult driving course such as mountain roadways and slippery roadways due to snow or rain. This error may lead to either sharp turn or quick braking.

Data dissemination has been a challenging issue in VANET for delay-intolerant services. On-time disseminating about emergency, accident, congestion, and obstacles to other vehicles can prevent further collision and congestion in the same route.

In this study, we present an effective delay-aware data forwarding (DAPF) for driving safety and efficiency in vehicular

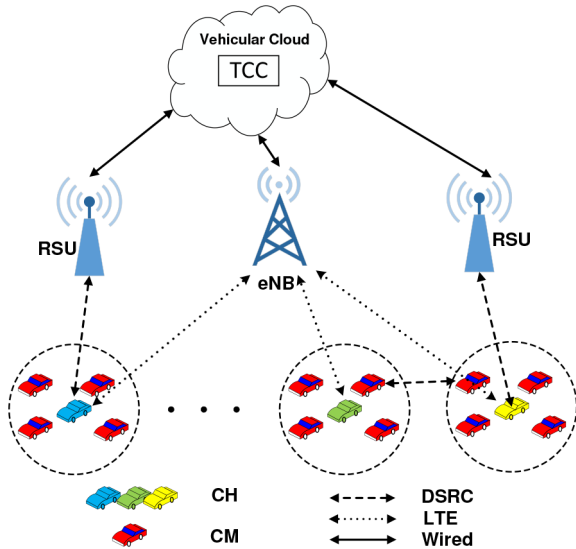


Fig. 1 Vehicular network architecture with VANETs, RSUs, eNodeBs, and vehicular cloud

networks. The proposed idea is based on the concept of processing and the timely delivery of the message. A most effective data processing position among a cluster head (CH), road-side unit (RSU), and traffic control centre in the vehicular cloud (shortly called cloud in this study) is selected to process the data and then the data is forwarded to the destination. Note that traffic control centre (TCC) is a cloud system having multiple servers to quickly process the data from the RSUs and deliver the response to a destination. We assume that the vehicles are moving in the form of clusters and the clustering follows our previous work in [8]. Clustering in VANET has been considered as an efficient means for communication, which helps efficient and safe driving and reducing data congestion [9]. To achieve better performance in clustering, much research has been done on the stable clustering with speed [10], trajectory [8], and traffic flow [11].

Fig. 1 shows the vehicular network architecture with VANETs, RSUs, eNodeBs (shortly eNBs), and vehicular cloud. It supports VANET communications (i.e. V2V, V2I, and I2V). In this figure, N vehicles have M CHs and only a CH can communicate with a RSU. The task of data processing and decision-making can be done at CH, RSU, or cloud, depending on the processing time and transmission time. Note that this study is the enhanced version of our previous conference paper [12]. Our contributions in this study are as follows:

- An effective method is proposed for selecting an appropriate processing position among CH, RSU, and cloud. The purpose of this selection is to timely send data to vehicles approaching accident area (see Section 4).
- A mathematical model is proposed based on processing cost and total delivery time (see Section 3).
- Simulation with realistic environments is compared with our mathematical analysis (see Section 6).

The rest of the paper is organised as follows. Section 2 summarises related work. Section 3 explains the overall system model of our work that includes the VANET model, processing data at CH, RSU, and cloud. Section 4 describes the optimisation of our work. Section 5 elaborates on the rerouting of vehicles, once they get the information related to an accident on their navigation path. Section 6 evaluates the performance of our scheme. Finally, in Section 7, we conclude this paper along with future work.

2 Related work

Much research has been done on data dissemination in VANET, and most of them consider only delay-tolerant data dissemination [13–15] and some consider delay-intolerant data dissemination [16–18]. Data dissemination of data means to send the data from a

source to a destination by considering delay and reliability. In the data dissemination, messages can be sent towards a destination or destination in unicast or multicast, i.e. sending them to either a specific vehicle or all the vehicles in a region.

In static-node assisted adaptive data dissemination protocol for vehicular networks (SADV) [19], Ding and Xiao proposed to deploy some static nodes at intersections. The function of the static node is to store a packet for some time and wait for a vehicle having the best delivery path to forward the packet. The concept of SADV is that the best path is not always available at the arrival time of a packet at an intersection, so it is better to store it in a static node for a while and then forward it to an appropriate next-hop vehicle when the best path is available.

In vehicle-assisted data delivery (VADD) [15], Zhao and Cao proposed the idea of carry-and-forward, where a vehicle carries a packet until it finds a new vehicle in its range to forward the packet to. VADD describes two different forwarding protocols, such as location first probe and direction first probe. The former selects a vehicle to forward the data on the basis of distance. The later selects a vehicle to forward the data on the basis of the direction towards the destination.

In trajectory-based data forwarding (TBD) [13], Jeong *et al.* proposed a new scheme to select the next-hop vehicle to forward the data on the basis of a trajectory in multi-hop V2I data delivery. TBD also uses a carry-and-forward method. A packet carrier vehicle in TBD determines whether it can forward the packet to another appropriate vehicle or carry it by itself towards the destination.

In trajectory-based statistical forwarding (TSF) [20], Jeong *et al.* proposed to forward data to a moving destination multi-hop I2V data delivery. TSF forwards packets to a target point where the destination vehicle is expected to pass through. The strategy of TSF is to send a packet earlier to a target point than the destination vehicle arrives there.

In [16], Wu *et al.* proposed a delay-sensitive data dissemination scheme. It uses two cooperative paths for the dissemination of data with short delays and high delivery ratios. In the scheme, a relay node selects two nodes as another relay node (as the main forwarder) and an auxiliary node (as a secondary forwarder), and then these two nodes broadcast the data. The next-hop relay node will repeat this process by selecting two forwarders (i.e. relay node and auxiliary node).

In [17], Wu *et al.* proposed a MAC protocol for delay-sensitive data dissemination. It uses Q-learning algorithm in order to avoid packet collisions by adjusting a contention window size. In [18], Li and Boukhatem proposed a new routing protocol using ant colony optimisation to find optimal route with minimum delay for delay-sensitive data dissemination.

Different from the literature, in this study, we propose a method that effectively selects a data processing position to forward data packets to by considering data processing costs (i.e. processing time, communication time, and the number of links). Specifically, in this study, we consider a situation where emergency packets need to be forwarded towards the vehicles that will pass through the roadway having an emergency.

3 System model

We consider a scenario where we have three data processing positions such as CH, RSU, and cloud. Suppose that there happens a collision on a heavy traffic road in an urban city. As shown in Fig. 2, two front vehicles in a cluster are two cluster members (CMs, i.e. CM_1^1 and CM_2^1) of the first cluster. They capture the multimedia data (e.g. image and video) and send their data to their cluster head (CH), CH_1 . On the basis of processing time and propagation delay, CH_1 then decides to either process the data by itself or send it to a RSU.

3.1 Multimedia data processing decision

For a decision-making process, a CH at first decides whether to process the data by itself or send it to a RSU. This decision of the CH initially depends on the processing time (T_{Pro}), communication

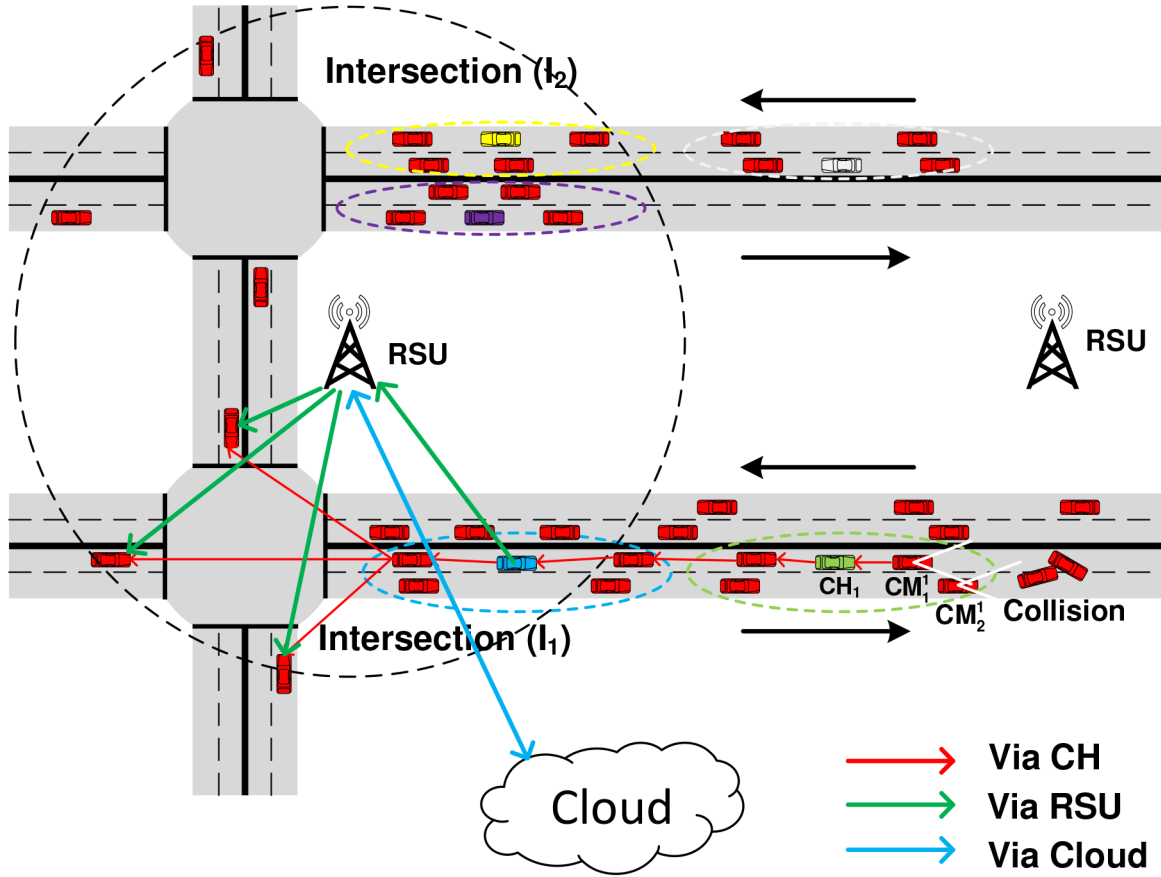


Fig. 2 An accident happening scenario in an urban traffic environment

time (T_{Com}), processing cost (C_{Pro}), and number of links (L). Then, it will check the communication range and channel capacity of the RSU. Once the CH sends the data to the RSU, the RSU decides to process the data by itself or send it to the cloud, depending on the processing rate of the RSU and traffic density around its coverage area. Lastly, if the RSU sends the data to the cloud for processing, the cloud will process it and broadcast a warning message to vehicles in the vicinity of the possible collision area in order to avoid traffic congestion and support the smooth flows of road traffic. Among the cloud, RSU, and CH, one is selected, which has the minimum value as in 1

$$X = \min \{ \alpha, \beta, \gamma \}, \quad (1)$$

where

$$\alpha = \left[T_{Pro_{cld}} + \sum_{l=1}^L T_{Com_l} \right] \times C_{Pro_{cld}}, \quad (2)$$

$$\beta = \left[T_{Pro_{rsu}} + \sum_{l=1}^L T_{Com_l} \right] \times C_{Pro_{rsu}}, \quad (3)$$

$$\gamma = \left[T_{Pro_{ch}} + \sum_{l=1}^L T_{Com_l} \right] \times C_{Pro_{ch}}. \quad (4)$$

The processing cost depends on the processing time of the processor of a CH, RSU or cloud. The longer the delay is, the lower the cost is. The processing cost is

$$C_{Pro} = C_e(1 - dv), \quad (5)$$

where C_{Pro} , C_e , and dv are the current processing cost, estimate cost, and depreciation value, respectively. Depreciation is known as the decrease in the value of cost. Depreciation depends upon the estimate processing time (T_{ep}) and actual processing time (T_{cp}),

where T_{ep} can be the same as T_{ch} , T_{rsu} or T_{cld} , and T_{cp} can have some delay in addition to T_{ep} . The ratio of T_{ep} to T_{cp} can be ≤ 1 , which indicates that the current processing cost C_{Pro} is less than or the same as the estimated cost C_e

$$\frac{T_{ep}}{T_{cp}} \begin{cases} < 1, & \text{for less cost} \\ = 1, & \text{for the same cost} \end{cases}$$

The depreciation value depends on the above processing time as follows:

$$dv = \begin{cases} 0, & \text{if } T_{ep} = T_{cp} \\ \frac{T_{ep}}{T_{cp}}, & \text{otherwise} \end{cases}$$

Depending on the value of (1), we can get the position for data processing, which is given by the following equation:

$$X_{ch_i} + X_{rsu_j} + X_{cld} = 1, \quad \text{for } i = 1, \dots, M \quad \text{and} \quad j = 1, \dots, J, \quad (6)$$

where M is the total number of CHs and J is the total number of RSUs, respectively.

For every CH X_{ch_i} , RSU X_{rsu_j} , and cloud $X_{cld} \in \{0, 1\}$, it is shown that the data can be processed by CH, RSU or cloud and only one of X_{ch_i} , X_{rsu_j} , and X_{cld} can be 1 because one place should be selected for data processing.

3.1.1 VANET model: In our VANET model, we consider an urban area road network where vehicle density is high. We assume that vehicles move in the form of clusters. The aim of our idea is to disseminate the accident information efficiently and effectively to intersection points adjacent to the accident area for a given road network. This disseminated information can prevent further traffic congestion on the accident road segment and its neighbouring road segments. Thus, this traffic congestion prevention can help a rescue

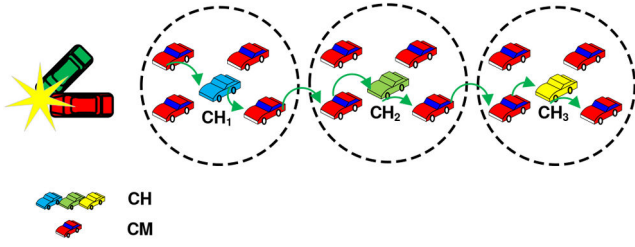


Fig. 3 A packet flow route via CH in vehicular networks

team to immediately reach the accident area and rescue the injured people.

All the notations and symbols used in this study are outlined in Nomenclature section. Some general facts and assumptions are as follows:

- Vehicles are moving in clusters.
- Each vehicle is equipped with lidar, radar, sensors, cameras, GPS receiver etc.
- Vehicles have two modes of communication; one is DSRC and the other is cellular communication (e.g. 4G-long-term evolution). Vehicles can switch to any mode according to need.
- RSUs are installed in such a way that each RSU can cover at least two clusters.
- CH can obtain road statistics (e.g. average speed and link delay) and a vehicle density from a RSU that it met recently.
- Once a vehicle at a junction (i.e. intersection) receives the information about an accident, it will follow a new route according to a scheme in Section 5.
- All vehicles' communication range is r such that $r_{CH} = r_{CM} = r$ for CH and CM, respectively.
- Wireless networks between vehicles and RSUs as well as wired networks between RSUs and TCC in the vehicular cloud are protected from security attacks by cloud-based security service systems [21]. Malicious packets from malicious vehicles can be filtered out by such cloud-based security service systems.

The wireless communication between vehicles V_i and V_j within one cluster is possible when the Euclidean distance (ED) between the vehicles is less than or equal to r_C , which is the radius of a cluster, considering one-hop DSRC communication range. A connectivity property [22] of CMs for a cluster is given by the statement S_{CH} :

$$S_{CH} = [ED_{V_i, V_j} \leq r_C, \forall (V_i, V_j)], \quad (7)$$

where ED_{V_i, V_j} is the Euclidean distance between two vehicles V_i and V_j . On the other hand, the communication between CH_i and RSU_j depends on the communication range of a RSU. If CH_i is in the range of RSU_j , then CH_i can communicate with RSU_j . The Euclidean distance ED_{CH_i, RSU_j} between CH_i and RSU_j must be less than or equal to the radius of a RSU, i.e. r_{RSU} . A connectivity property of a CH for a RSU is given by the statement S_{RSU} :

$$S_{RSU} = [ED_{CH_i, RSU_j} \leq r_{RSU}, \forall (CH_i, RSU_j)]. \quad (8)$$

The inter-arrival rate of vehicles follows a Poisson distribution [23, 24], and RSUs share the information of a vehicle density and road statistics with CHs. The probability of k vehicles passing through an intersection is denoted as

$$P(k) = \frac{e^{-\lambda t} \lambda t^k}{k!}, \quad (9)$$

where λ is the expected arrival rate of vehicles in a road segment as $\lambda = E(V)$ (i.e. the expected number of vehicle arrivals for unit time t), and k is the number of vehicles passing through an intersection for unit time t .

3.1.2 Processing data at CH: There are two cases in which data can be processed at CH. The first case is that from (1) and (6), CH is selected to process the data. The second case is that from (1) and (6), either RSU or cloud is selected to process the data, but the CH is not in the range of RSU or that the currently required up-link and down-link capacities (C_u, C_d) are less than those of up-link and down-link data rates (R_u, R_d) of user i . In these two cases, the CH should be selected to process the data and forward it to the other vehicles. Thus, the selection of $X_{ch_i} = 1$ and $X_{rsu_i} = X_{cl_d} = 0$ holds if $CH_i \notin RSU_R$. This selection means that the i th CH, CH_i , is not in the range of a RSU, i.e. RSU_R .

The up-link and down-link data rates can also play a role in selecting a CH for processing the data in the following conditions:

$$\sum_{i=1}^N R_{u_i} \geq C_u, \quad (10)$$

$$\sum_{i=1}^N R_{d_i} \geq C_d. \quad (11)$$

A route for the delivery of multimedia data from a source to a destination for CH is given as follows [22, 25]:

$$DP = \{P_S, CM_i^1, CH_1, CM_j^1, CM_h^2, CH_2, CM_k^2, \dots, P_D\}, \quad (12)$$

where DP is the packet's delivery path from a source to a destination, including the sub-path from a cluster to another cluster, and P_S is a packet source, P_D is a packet destination, CM_i^s represents the i th CM of cluster s and CH_s is the CH of cluster s .

The processing time for CH is denoted as $T_{Pro_{ch}}$ and $T_{Pro_{ch}} = \text{Data_size}/P_{CH}$, where P_{CH} is the processing rate of CH. The propagation delay and transmission time are calculated as $D_{Pro} = d/s$ and $T_{Com} = P_L/R$, respectively, where d is the distance between two nodes, s is the signal propagation speed, P_L is the packet length, and R is the transmission rate. The total delivery time when the CH is selected as a processing position can be calculated as follows:

$$T_{tot_{ch}} = \sum_{m=1}^M T_{Pro_{ch_m}} + \sum_{l=1}^L (T_{Com,l} + D_{Pro,l}), \quad (13)$$

for $m = 1, \dots, M$ and $l = 1, \dots, L$,

where M is the total number of CHs and L is the total number of transmission links (i.e. clusters). Fig. 3 shows the explanation of (13), where it can be seen that the packet is delivered to the CHs.

3.1.3 Processing data at RSU: The data received by a RSU is either processed by the RSU or sent to the cloud, depending on the processing rate of the RSU, and traffic density in the vicinity is given as follows:

$$P_r \leq P_{RSU}, \quad \text{for } i = 1, \dots, N, \quad (14)$$

where P_r is the processing rate assigned to the i th multimedia data packet and P_{RSU} is the total processing rate of RSU and

$$D_T \leq T_H, \quad (15)$$

where D_T and T_H are the density of traffic and a threshold value, respectively. If the density is greater than the threshold value, the RSU will send the data to the cloud for processing the data and broadcasting a warning message to control further traffic congestion. A delivery path of the multimedia data from a source to a destination for a RSU is given as follows:

$$DP = \{P_S, CM_i^1, CH_1, CM_j^1, CM_h^2, CH_2, RSU_1, RSU_2, \dots, P_D\}. \quad (16)$$

The packet delivery time at RSU (denoted as $T_{\text{tot}_{\text{rsu}}}$) is given as follows:

$$T_{\text{tot}_{\text{rsu}}} = \sum_{m=1}^M T_{\text{Pro}_{\text{ch}_m}} + \sum_{j=1}^J T_{\text{Pro}_{\text{rsu}_j}} + \sum_{l=1}^L (T_{\text{Com},l} + D_{\text{Pro},l}), \quad (17)$$

for $m = 1, \dots, M, j = 1, \dots, J, \text{ and } l = 1, \dots, L,$

where $T_{\text{Pro}_{\text{rsu}}}$ is the processing time for RSU, which is computed as $T_{\text{Pro}_{\text{rsu}}} = \text{Data_size}/P_{\text{RSU}}$, where P_{RSU} is the processing rate of RSU, M is the total number of CHs, J is the total number of RSUs, and L is the total number of transmission links (i.e. clusters).

Fig. 4 shows the explanation of (17), where the packet is delivered from CH₁ to RSU₁ if CH₁ is in the communication range of a RSU called RSU₁. This figure shows a single-hop data forwarding from CH₁ to RSU₁. However, if CH₁ is not in any RSU's communication range, it will send the packet to CH₂, which is directly connected to another RSU called RSU₂.

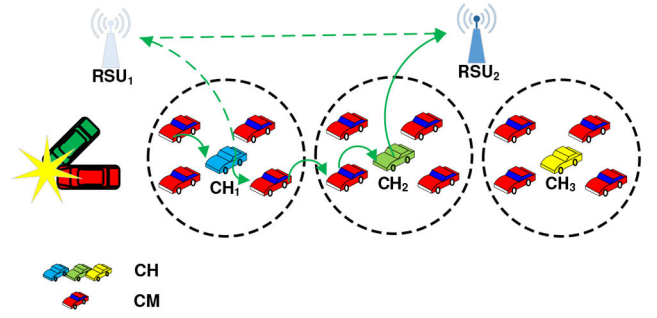


Fig. 4 A packet flow route via RSU in vehicular networks

3.1.4 Processing data at cloud: Now, we consider the case where the data is finally sent to the cloud. The cloud starts processing it and decides to broadcast a response packet for the data to the vehicles approaching the vicinity of the accident area, where they will get stuck due to traffic congestion. This packet will allow the approaching vehicles to avoid such a traffic-congestion area with efficient detour paths. Packet delivery time at the cloud is given as follows:

$$T_{\text{tot}_{\text{cid}}} = \sum_{m=1}^M T_{\text{Pro}_{\text{ch}_m}} + \sum_{j=1}^J T_{\text{Pro}_{\text{rsu}_j}} + T_{\text{Pro}_{\text{cid}}} + \sum_{l=1}^L (T_{\text{Com},l} + D_{\text{Pro},l}), \quad (18)$$

for $m = 1, \dots, M, j = 1, \dots, J, \text{ and } l = 1, \dots, L,$

where $T_{\text{Pro}_{\text{cid}}}$ is the processing time for the cloud, which is computed as $T_{\text{Pro}_{\text{cid}}} = \text{Data_size}/P_{\text{CLD}}$, where P_{CLD} is the processing rate of the cloud.

Fig. 5 shows the explanation of (18), where the process is the same as RSU. However, the difference is that once the RSU receives the packet, it sends it to the cloud, and then a response packet (i.e. a warning message) will be broadcasted to the vehicles in the vicinity of the target location.

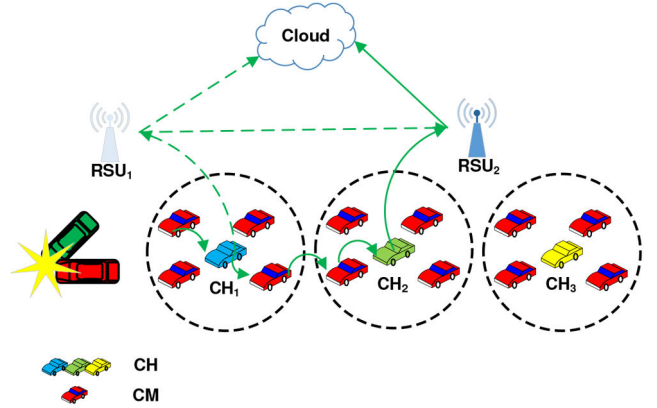


Fig. 5 A packet flow route via the cloud in vehicular networks

4 Optimisation problem

Our goal is to minimise the transmission and processing time in consideration of cost. The other task is to efficiently deliver the data to the nearby vehicles for congestion control and smooth traffic flow. The optimisation problem is formulated as follows:

$$\min \left(\sum (T_{\text{tot}_{\text{ch}}} \cdot X_{\text{ch}} + T_{\text{tot}_{\text{rsu}}} \cdot X_{\text{rsu}} + T_{\text{tot}_{\text{cid}}} \cdot X_{\text{cid}}) \right), \quad (19)$$

such that

$$T_{\text{tot}} < T_v, \quad (20)$$

$$C_{\text{Pro}} > C'_{\text{Pro}}, \quad (21)$$

where T_{tot} and T_v are the total delivery time ($T_{\text{tot}_{\text{ch}}}$, $T_{\text{tot}_{\text{rsu}}}$, or $T_{\text{tot}_{\text{cid}}}$) and the time a vehicle takes to arrive at the intersection, respectively. As explained earlier, C_{Pro} decreases as the processing time increases, i.e. C_{Pro} must be greater than the lowest cost (i.e. C'_{Pro}), which means that it corresponds to the maximum time the processor can take to process the data for usefulness, where $C'_{\text{Pro}} = C_e(1 - d_v')$ and $d_v' = (T_{\text{ep}} + (T_v - T_{\text{tot}}))/T_{\text{cp}}$.

Algorithm 1 (see Fig. 6) can be explained through the procedure of selecting a processing position among a CH, RSU, and cloud, as shown in Fig. 7, where three variables such as α , β , and γ , which

```

1: function DATA DISSEMINATION( $\alpha, \beta, \gamma$ )
2:   Set  $E \leftarrow 1$  and  $\text{flag} \leftarrow 1$ 
3:   while  $\text{flag}=1$  do
4:     Vehicles encountering the accident will capture
       and send the multimedia data to the CH.
5:     if  $\gamma < \alpha$  and  $\gamma < \beta$  then
6:       CH will process the data and send it to the
       other following groups.
7:     else if  $\beta < \alpha$  and  $\beta < \gamma$  then
8:       if  $\text{CH}_R \notin \text{RSU}_R$  or  $\sum_{i=1}^N R_{u_i} \geq C_u$  &
        $\sum_{i=1}^N R_{d_i} \geq C_d$  then
9:         CH will process the data and send to the
       other following groups.
10:      else if  $P_{r_i} \leq P_{\text{RSU}}$  or  $D_T \leq T_H$  then
11:        RSU will process the data and broadcast
       it to the vicinity.
12:      else
13:        Send the data to the cloud.
14:      end if
15:      else if  $\alpha < \beta$  and  $\alpha < \gamma$  then
16:        The data is processed by the cloud, and then
       the cloud broadcasts the response packet to all the
       vehicles that can be effected in near future by this
       accident.
17:      end if
18:    end while
19:    return X
20: end function

```

Fig. 6 Algorithm 1: data dissemination algorithm

are obtained from (2), (3), and (4), are compared and then the processing position is selected accordingly. If γ is less than α and β , CH is selected as a processing position. If β is less than α and γ , RSU is selected as a processing position. If α is less than β and γ , the cloud is selected as a processing position. The time complexity of Algorithm 1 (Fig. 6) is $O(3M - 1)$, where M is the number of

clusters and each cluster has at least three communication links because the data can be sent from the vehicle that encounters the accident to the CH, the CH broadcasts the data to its CMs, and it lets one of its members closest to the next cluster send the data to the next cluster. This time means that the total time taken to disseminate the data to the destination depends on M . In line 8 in Algorithm 1 (Fig. 6), if a CH is not in the range of a RSU or the uplink and downlink capacities are less than the uplink and downlink data rates of the users, then the data is still processed by the CH.

5 Rerouting of vehicles coming towards the affected area

Once the information is disseminated to the vehicles at an intersection heading towards the affected area, the vehicles start to follow a new route calculated by a CH, a RSU or the TCC in the vehicular cloud. The vehicle sends a request to TCC for searching for another appropriate route for it. A new route is selected to minimise the congestion near the accident area. A lot of research has already been done on rerouting algorithms for VANET [26, 27]. This is not the major part of this study, so this study just explains that the vehicles will change their routes after receiving the accident information. Assume that when a path used by a vehicle from a source to a destination denoted as $P_{S,D}$ is an optimal route with distance D_{old} . Assume that when an accident occurs in its route, a vehicle has to change its route. For the purpose of changing the route, the vehicle sends a request for a new route to TCC. The new route not only depends on the congestion of alternate routes but also considers traffic flow q that is calculated in hours; we have multiplied the hours with 3600 to convert them into seconds. Let K be density and V_m be mean velocity. The estimated travelling time (T_{Te}), new travelling time (T_{Tn}), and end-to-end (E2E) delay D_{E2E} by selecting a new route are given as follows:

$$T_{Te} = \frac{D_{old}}{V_m} + (T_{st}), \quad (22)$$

where T_{st} is the total expected traffic signal timing. If $T_{tot} < T_v$ and $C_{Pro} < C'_{Pro}$ are satisfied, T_{Tn} is calculated for a new route having a distance D_{new}

$$T_{Tn} = \frac{D_{new}}{V_m} + (T_{st}), \quad (23)$$

$$D_{E2E} = T_{Tn} - T_{Te}. \quad (24)$$

A new route is selected, depending on parameters q , K , and V_m that are given as follows:

$$q = \frac{N}{Avg(t)} \times 3600, \quad (25)$$

$$K = \frac{N}{L_{S,D}}, \quad (26)$$

$$V_m = \frac{q}{K}, \quad (27)$$

where N , $Avg(t)$, and $L_{S,D}$ are the total number of vehicles on the route, average time, and the length of the roadway from the source to the destination, respectively. Equation (25) shows the number of vehicles passing through a point in a unit of time. Equation (26) calculates the number of vehicles in a specific length of road as density. Equation (27) shows the average velocity of vehicles that depends on traffic flow and density.

6 Performance evaluation

This section evaluates the performance of our scheme by selecting the appropriate position for data processing and then dissemination.

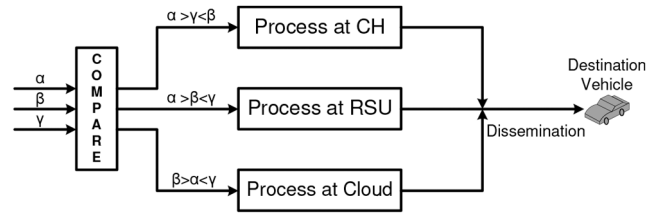


Fig. 7 Procedure of selecting a processing position

Table 1 Simulation parameters

| Parameter | Value |
|--------------------------|-------------|
| road network | 1 km × 1 km |
| the number of Vehicles | 50 |
| the number of RSUs | 4 |
| data rate | 6 Mbps |
| simulation time | 100 s |
| processing rate of CH | 3 Mbps |
| processing rate of RSUs | 7 Mbps |
| processing rate of cloud | 10 Mbps |
| data packet size | 2 MB |
| the range of cluster | 60 m |
| the range of RSU | 120 m |
| the range of cloud | 500 m |

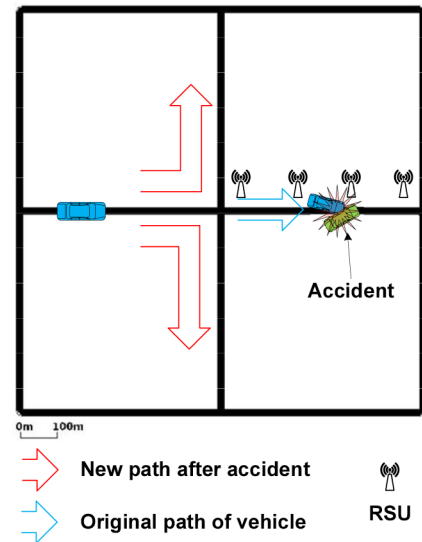


Fig. 8 A grid map road network for simulation

The data rate is set to 6 Mbps, the number of vehicles is set to 50, the number of RSUs is set to 4, and the speed of vehicles is set to 14 m/s. Also, the communication ranges of CHs (vehicles) and RSUs are set to 60 and 120 m, respectively.

Table 1 shows simulation parameters. We use a road network with nine intersections and only focus on one road segment where the accident occurs and we deploy four RSUs to cover the whole road segment, which is 500m. The communication range of one RSU can cover two clusters. We use OMNeT++ [28], Veins [29], and SUMO [30] for the simulation in a grid-map road network, as shown in Fig. 8. OMNeT++ is an open source software to simulate computer networks including vehicular networks, and Veins supports IEEE 802.11p communication as an OMNeT++ package. SUMO is an open source software for realistic vehicle mobility in road networks.

At the beginning of the simulation, we fixed the routes of vehicles. After passing the centre junction, vehicles start to make clusters and we change an accident spot to check the performance for different numbers of clusters. Once an accident occurred, the vehicles approaching the accident spot decelerates and finally stop. If the front vehicle encounters an accident it will start sending

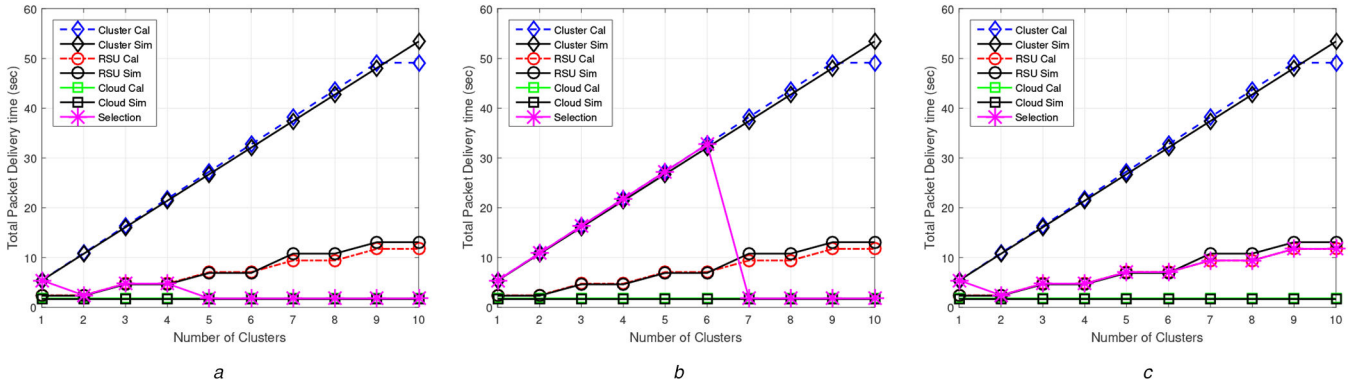


Fig. 9 Scenario 1: total packet delivery time with respect to the number of clusters, where the number of clusters increases (a) Processing cost ratio for CH, RSU, and cloud is 0.2:0.6:2, (b) Processing cost ratio for CH, RSU, and cloud is 0.1:0.6:2, (c) Processing cost ratio for CH, RSU, and cloud is 0.1:0.3:2

Table 2 Processing cost ratio for simulation scenarios

| Processing cost ratio | CH | RSU | Cloud |
|-----------------------|-----|------------|-------|
| | | scenario 1 | |
| a | 0.2 | 0.6 | 2 |
| b | 0.1 | 0.6 | 2 |
| c | 0.1 | 0.3 | 2 |
| | | scenario 2 | |
| a | 0.4 | 0.75 | 1 |
| b | 0.5 | 0.75 | 1 |
| c | 0.5 | 0.9 | 1 |

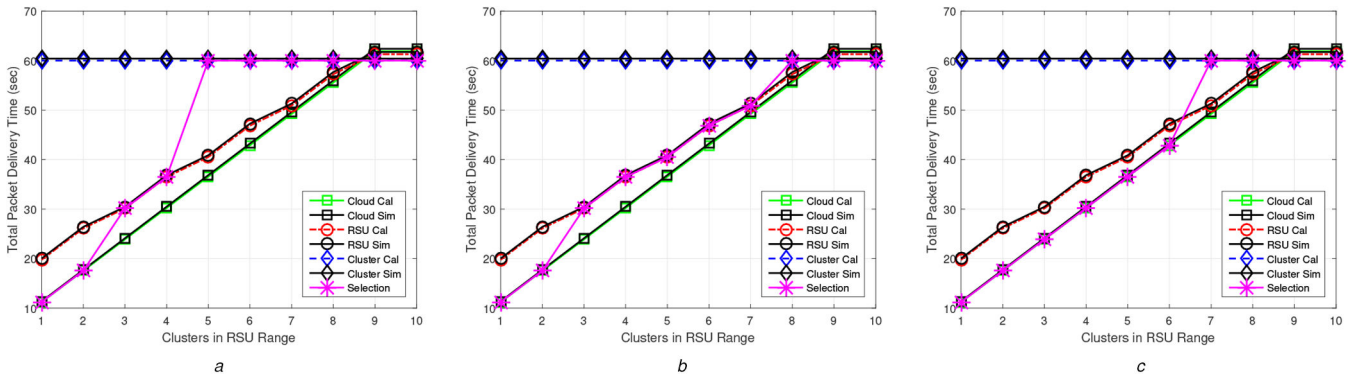


Fig. 10 Scenario 2: total packet delivery time with respect to the number of clusters, where we search for a cluster among ten clusters until the cluster is within a RSU's communication range, from the first cluster up to the last cluster (a) Processing cost ratio for CH, RSU, and cloud is 0.4:0.75:1, (b) Processing cost ratio for CH, RSU, and cloud is 0.5:0.75:1, (c) Processing cost ratio for CH, RSU, and cloud is 0.5:0.9:1

accident information to its CH, the CH then decides to process it by itself, or send it to either a RSU or the cloud, depending on the estimated delivery time and estimated processing cost.

Our goal is to effectively select an appropriate position for data processing and then response data for the detour is disseminated to vehicles at a target intersection. Thus, the vehicles can follow alternate routes other than the accident road segment.

Performance metrics: We use total packet delivery time as a metric for the performance of our scheme. We use two different scenarios: (i) total packet delivery time with respect to the number of clusters, where the number of clusters increases, and (ii) total packet delivery time with respect to the number of clusters, where we search for a cluster among ten clusters until the cluster is within a RSU's communication range, from the first cluster up to the last cluster.

(i) *Total packet delivery time with respect to the number of clusters, where the number of clusters increases:* As mentioned above, we use two scenarios. As shown in Fig. 9, the first scenario is used to measure the total delivery time while selecting CH, RSU or cloud.

The selection of any of them is performed according to the processing cost and delivery time. We use different random processing cost ratios for selecting processing position for each, and then compare our results with mathematical analysis.

Table 2 shows the processing cost ratio among CH, RSU, and cloud for two simulation scenarios (i.e. scenarios 1 and 2). The effective selection of CH, RSU or cloud is marked by '*' in Figs. 9 and 10.

In Fig. 9a, we use the processing costs for CH, RSU, and cloud with a ratio of 0.2:0.6:2, respectively. In Fig. 9b, we use the processing costs for CH, RSU, and cloud with a ratio of 0.1:0.3:2, respectively. In Fig. 9c, we use the processing costs for CH, RSU, and cloud with a ratio of 0.2:0.6:2, respectively. From Fig. 9, it can be seen that the selection of processing point is different for different cases.

(ii) *Total packet delivery time with respect to the number of clusters, where we search for a cluster among ten clusters until the cluster is within a RSU's communication range, from the first cluster up to the last cluster:* We elaborate on the second scenario where we have ten clusters and four RSUs. Suppose that initially,

the vehicle in CH₁ that encounters the accident is within the range of the RSU, it sends the packet to its CH and CH will send it to RSU and then it follows the same procedure shown in Fig.4, but if CH₁ is not within a RSU's communication range, then it will send it to CH₂. Now CH₂ will check whether it is within a RSU's communication range or not. This search for a RSU over the VANET having the clusters is repeated until a CH can connect to a RSU by the DSRC communication range. As shown in Fig. 10, when data is sent by clusters, the total delivery time is much higher than that of a RSU. However, the total delivery time of the cloud remains almost similar to that of a RSU.

In Fig. 10a, we use the processing costs for CH, RSU, and cloud with a ratio of 0.4:0.75:1, respectively. In Fig. 10b, we use the processing costs for CH, RSU, and cloud with a ratio of 0.5:0.75:1, respectively. In Fig. 10c, we use the processing costs for CH, RSU, and cloud with a ratio of 0.5:0.9:1, respectively.

7 Conclusion

In this study, we proposed a delay-aware data processing (called DAPF) for effective and delay-sensitive data processing and dissemination in vehicular networks. One of the main issues in vehicular communication is on-time dissemination of message at the emergency situation (e.g. accident or obstacle) to the vehicles having the same route to their destination. This on-time dissemination can prevent further vehicle collision and traffic congestion from happening. In this study, we proposed an effective way of selecting a processing place of the message among a CH, a RSU, and the cloud, on the basis of total delivery time and cost. We also elaborated that this effective selection and on-time dissemination helps the upcoming vehicles to select appropriate routes to their destinations. To evaluate the effectiveness of our DAPF, simulation results were compared with mathematical results. As future work, we will extend our work to multi-vehicles-to-multi-vehicles communication where multiple vehicles can communicate with each other at the same time and they can collaborate to effectively process tasks.

8 Acknowledgments

This research was supported by the Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (Ministry of Science and ICT, MSIT) (No. 2016-0-00078, Cloud-based Security Intelligence Technology Development for the Customized Security Service Provisioning).

9 References

- [1] Chisalita, I., Shahmehri, N.: 'A peer-to-peer approach to vehicular communication for the support of traffic safety applications', IEEE Int. Conf. on Intelligent Transportation Systems, Singapore, Singapore, 2002
- [2] Somasundaram, G., Kavitha, , Ramachandran, K.I.: 'Lane changing detection and tracking for a safe-lane approach in real time vision based navigation systems', Int. Conf. On Computer Science, Engineering and Applications (CCSEA), Chennai, India, 2011, pp. 345–361
- [3] Amadeo, M., Campolo, C., Molinaro, A.: 'Enhancing IEEE 802.11p/WAVE to provide infotainment applications in VANETS', *Ad Hoc Netw.*, 2012, **10**, pp. 253–269
- [4] Baiocchi, A., Cuomo, F.: 'Infotainment services based on push-mode dissemination in an integrated VANET and 3G architecture', *J. Commun. Netw.*, 2013, **15**, (2), pp. 179–190
- [5] Salvo, P., Felice, M.D., Cuomo, F., *et al.*: 'Infotainment traffic flow dissemination in an urban VANET', Globecom – Ad Hoc and Sensor Networking Symp., Anaheim, CA, USA, 2012
- [6] 'National highway traffic safety administration, preliminary statement of policy concerning automated vehicles', 2013
- [7] Sam, D., Cyril Raj, V., Esther Evangelin, T.: 'A vehicle control system using a time synchronized hybrid VANET to reduce road accidents caused by human error', *Veh. Commun.*, 2016, **6**, pp. 17–28
- [8] Shahwani, H., Bui, T.D., Jeong, J.P., *et al.*: 'A stable clustering algorithm based on affinity propagation for VANETS', Int. Conf. on Advanced Communication Technology (ICACT), Bongpyeong, South Korea, 2017
- [9] Jia, D., Lu, K., Wang, J., *et al.*: 'A survey on platoon-based vehicular cyber-physical systems', *IEEE Commun. Surv. Tutor.*, 2016, **18**, (1), pp. 263–284
- [10] Rawashdeh, Z.Y., Mahmud, S.M.: 'A novel algorithm to form stable clusters in vehicular ad hoc networks on highways', *EURASIP J. Wirel. Commun. Netw.*, 2012, **2012**, pp. 1–13
- [11] Almalag, M.S., Weigle, M.C.: 'Using traffic flow for cluster formation in vehicular ad-hoc networks', Local Computer Networks (LCN), Denver, CO, USA, 2010
- [12] Shahwani, H., Mugabarigira, B.A., Shin, J., *et al.*: 'An effective data processing and data dissemination in vehicular networks', Int. Conf. on Ubiquitous Information Management and Communication (IMCOM), Langkawi, Malaysia, January 2018
- [13] Jeong, J.P., Guo, S., Gu, Y.J., *et al.*: 'Trajectory-based data forwarding for light-traffic vehicular ad hoc networks', *IEEE Trans. Parallel Distrib. Syst.*, 2011, **22**, (5), pp. 743–757
- [14] Jeong, J.P., Kim, J., Hwang, T., *et al.*: 'TPD: travel prediction-based data forwarding for light-traffic vehicular networks', *Comput. Netw.*, 2015, **93**, pp. 166–182
- [15] Zhao, J., Cao, G.: 'VADD: vehicle-assisted data delivery in vehicular ad hoc networks', *IEEE Trans. Veh. Technol.*, 2008, **57**, (3), pp. 1910–1922
- [16] Wu, C., Ohzahata, S., Kato, T.: 'A broadcast path diversity mechanism for delay sensitive VANET safety applications', Vehicular Networking Conf. (VNC), Amsterdam, Netherlands, 2011
- [17] Wu, C., Ohzahata, S., Ji, Y., *et al.*: 'A MAC protocol for delay-sensitive VANET applications with self-learning contention scheme', 11th Consumer Communications and Networking Conf. (CCNC), Las Vegas, NV, USA, 2014
- [18] Li, G., Boukhatem, L.: 'An intersection-based delay sensitive routing for VANETS using ACO algorithm', 23rd Int. Conf. on Computer Communication and Networks (ICCCN), Shanghai, China, 2014
- [19] Ding, Y., Xiao, L.: 'SADV: static-node-assisted adaptive data dissemination in vehicular networks', *IEEE Trans. Veh. Technol.*, 2010, **59**, (5), pp. 2445–2455
- [20] Jeong, J.P., Guo, S., Gu, Y.J., *et al.*: 'Trajectory-based statistical forwarding for multihop infrastructure-to-vehicle data delivery', *IEEE Trans. Mob. Comput.*, 2012, **11**, (10), pp. 1523–1537
- [21] Hyun, S., Kim, J., Kim, H., *et al.*: 'Interface to network security functions for cloud-based security services', *IEEE Commun. Mag.*, 2018, **56**, (1), pp. 2–9
- [22] Wang, Y., Huang, L., Gu, T., *et al.*: 'Data-driven traffic flow analysis for vehicular communications', IEEE INFOCOM – IEEE Conf. on Computer Communications, Toronto, Canada, 2014, pp. 1977–1985
- [23] Ge, X., Cheng, H., Mao, G., *et al.*: 'Vehicular communications for 5G cooperative small-cell network' *IEEE Trans. Veh. Technol.*, 2016, **65**, (10), pp. 7882–7894
- [24] Chi, J., Do, S., Park, S.: 'Traffic flow-based roadside unit allocation strategy for VANET', Int. Conf. on Big Data and Smart Computing (BigComp), Hongkong, China, 2016, pp. 245–250
- [25] Zhang, L., Yu, B., Pan, J.: 'GeoMob: a mobility-aware geocast scheme in metropolitans via taxicabs and buses', IEEE INFOCOM – IEEE Conf. on Computer Communications, Toronto, Canada, 2014, pp. 1779–1787
- [26] Jeong, J., Jeong, H., Lee, E., *et al.*: 'SAINT: self-adaptive interactive navigation tool for cloud-based vehicular traffic optimization', *IEEE Trans. Veh. Technol.*, 2016, **65**, (6), pp. 4053–4067
- [27] Shen, Y., Lee, J., Jeong, H., *et al.*: 'SAINT+: self-adaptive interactive navigation tool+ for emergency service delivery optimization', *IEEE Trans. Intell. Transp. Syst.*, 2018, **19**, (4), pp. 1038–1053
- [28] OMNeT++. Network simulation framework. Available at <https://omnetpp.org/> accessed on 19 April 2017
- [29] Sommer, C., German, R., Dressler, F.: 'Bidirectionally coupled network and road traffic simulation for improved IVC analysis', *IEEE Trans. Mob. Comput.*, 2011, **10**, (1), pp. 3–15
- [30] Krajzewicz, D., Erdmann, J., Behrisch, M., *et al.*: 'Recent development and applications of SUMO – simulation of urban mobility', *Int. J. Adv. Syst. Meas.*, 2012, **5**, (3 & 4), pp. 128–138