SDN-Assisted Safety Message Dissemination Framework for Vehicular Critical Energy Infrastructure

Sahaya Beni Prathiba, *Student Member, IEEE*, Gunasekaran Raja, *Senior Member, IEEE*,
Ali Kashif Bashir*, Senior Member, IEEE*, Ahmad Ali AlZubi, and Brij Gupta, *Senior Member, IEEE*

***Abstract*—The proliferation of fifth-generation (5G) networks toward vehicle-to-everything (V2X) communication has paved the way for driverless autonomous vehicles (AVs) in vehicular critical energy infrastructures (CEI). Though technological advancements improve AVs, the safety-critical messages (SCMs) still play a vital role in reducing crashes, preventing injuries, and saving lives. AVs' high speed and complex network topology challenge disseminating SCMs with a highly successful delivery ratio and extremely low latency. Furthermore, the typical SCM dissemination schemes cause channel congestion and minimize the delivery ratio, making the systems incompatible with the AVs. Therefore, in this article, a software-defined-networking-assisted continuous clustering approach called migrating consignment region (MiCR) based on the federated K-means algorithm is proposed for disseminating SCMs to the AVs via 5G V2X communication. Unlike other methods that create clusters for every instance of SCM dissemination, MiCR continuously holds moving clusters for disseminating SCMs to AVs with ultrahigh reliability and low latency. The proposed MiCR approach has been simulated under real-time highway road maps and compared with other methods. The simulation results prove the superiority of MiCR in terms of network overload, SCM delivery ratio, latency, dissemination efficiency, and collision rate compared with the existing methods.**

***Index Terms*—Autonomous vehicles (AVs), federated K-means clustering algorithm, fifth-generation (5G) vehicle-to-everything (V2X), safety-critical messages (SCMs), software-defined networking (SDN), vehicular critical energy infrastructure (CEI).**

I. Introduction

ACCESS to the recent advancements in critical energy infrastructures (CEIs) has open up the way for vehicular CEI in heading toward emerging vehicles such as connected cars, autonomous vehicles (AVs), and Internet of Vehicles [1]. By 2025, the AVs will be the future among the various emerging vehicles [2]. The prominence of vehicle-to-everything (V2X) technology increases the level of certainty regarding a vehicle's surroundings and thereby seeding AVs for safer autonomous driving [3]. Furthermore, incorporating artificial intelligence and fifth-generation (5G) networks into V2X technology boosts the vehicular CEI's performance [4].

Though 5G will amp up the V2X services, its network functions are expected to run over a unified operating system, especially at its edge [5]. Software-defined networking (SDN) optimizes and highly simplifies network management operations. Thus, SDN-based 5G V2X brings resilience, elasticity, and programmability by efficiently exploiting the available network resources and simplifying network management [6]. The centralized SDN can manage one or more edge servers available at the base station. Therefore, the SDN-assisted 5G V2X will enable AVs' highly reliable safety-critical message (SCM) dissemination.

The SCMs are delay-sensitive warnings sent by the network to the AVs when detecting potentially dangerous situations on the road for enhancing road safety [7]. Traditionally, the backbone network transmits the SCM to the vehicles by the flooding mechanism, which leads to the broadcast storm problem and causes severe congestion and packet loss in the network [8]. To mitigate the broadcast storm problem and to deal with these challenges, the SDN-assisted 5G V2X technology requires an intelligent clustering technique [9].

Vehicular CEI is the indispensable commercial division that facilitates people, water, foods, medicines, and fuel to travel.

Sahaya Beni Prathiba and Gunasekaran Raja are with the Next Generation Network Laboratory, Department of Computer Technology, Anna University, Chennai 600025, India (e-mail: sahayabeni@ieee.org; dr.r.gunasekaran@ieee.org).

Ali Kashif Bashir is with the Department of Computing and Mathematics, Manchester Metropolitan University, M156BH Manchester, U.K. (e-mail: A.bashir@mmu.ac.uk).

Ahmad Ali AlZubi is with the Department of Computer Science, Community College, King Saud University, Riyadh 11437, Saudi Arabia (e-mail: aalzubi@ksu.edu.sa).

Brij Gupta is with the Department of Computer Engineering, National Institute of Technology Kurukshetra, Kurukshetra 136119, India, with the Department of Computer Science and Information Engineering, Asia University, Taichung 41354, Taiwan, and also with Macquarie University, Macquarie Park, NSW 2109, Australia (e-mail: bbgupta@nitkkr.ac.in).

Various types of risks provoke safety in vehicular CEI. One significant risk is disseminating the SCMs to the AVs in vehicular CEIs, without any packet loss. Protecting the vehicular CEI in ensuring the resilience of the AV network is a key challenge [10]. Due to the heterogeneous infrastructure nature, the vehicular CEI requires an advanced approach than the traditional approaches. Federated learning is an advanced learning approach that trains an algorithm over various edge servers without sharing the local data. In the centralized federated learning approach, a central server like SDN coordinates all the edge nodes associated with the base stations for executing the algorithm. The edge nodes then transfer the updates to the central server.

In this article, we propose a continuous moving cluster framework called migrating consignment region (MiCR) for disseminating the SCMs. MiCR holds a centralized federated-K-means-clustering-based algorithm. Adopting the 5G network and the centralized SDN controller in MiCR offers high coverage and extremely low latency. The SDN controller acquires floating vehicle data such as unique ID, location, velocity, and destination and executes the proposed ConstructCR algorithm to group the AVs based on the collected floating vehicle data. Once the SDN groups the AVs, it selects an optimal head AV with high residual energy and proximity to the Seed-AV (SAV) center from each group. These groups are termed as consignment regions (CRs) containing the targeted AVs for delivering the SCMs. Once the SAVs are determined, SDN disseminates the SCMs to the SAVs; subsequently, the SAVs deliver the SCMs to the targeted AVs available in its CR via vehicle-to-vehicle (V2V) communication. On the contrary to the conventional approaches, which creates CRs for each SCM dissemination, MiCR creates once and maintains continuously through the proposed CombineCR and BreakCR algorithms. Thus, the proposed MiCR framework delivers the SCMs to the AVs with high reliability and significantly reduces the communication cost.

The key contributions of this article are as follows.

1) The proposed MiCR framework employs SDN-assisted beaconless 5G V2X communication to disseminate SCMs to AVs in vehicular CEI with ultrahigh reliability and low latency in the highway scenario.
2) The MiCR framework minimizes the overhead of the network, not by creating the CRs for each instance of data dissemination. Instead, the CRs are created once and maintained continuously by CombineCR, BreakCR, and UpdateCR algorithms, which handles the high mobility nature of AVs and delivers the SCMs efficiently.

The rest of this article is organized as follows. Section II reviews the related work. Section III presents the system model. Section IV delivers the various phases involved in the construction and maintenance of MiCR. Section V delivers the report of the experimental setup and the results. Finally, Section VI concludes this article.

II. Related Work

In recent years, increasing safety through the transmission of SCMs in autonomous driving plays a major role. A time-barrier-based emergency message dissemination (TB-EM) technique disseminates SCMs by clustering the vehicles [8]. The technique reduces the broadcast storm problem by allowing only the farthest vehicles to rebroadcast the SCMs after a particular time barrier expiration. Wang *et al.* [11] deliver fog-computing-enabled vehicular systems for managing real-time traffic loads. The approach locates the fog nodes at the road-side units (RSUs) and the parked and moving vehicles. In [12], a protocol is proposed for prioritizing and guaranteeing reliability in SCM transmission from a vehicle based on the evaluation of accident risk. The accident risk evaluation calculates the distance between the vehicle and the danger zone and, thus, transmits the SCMs with higher reliability.

A deep recurrent neural network (DeepVM) is proposed in [13] for offering intelligent vehicle services. DeepVM predicts vehicle mobility with improved performance for the provision of vehicular services. The delay-constrained routing for BusNet (DCRB) protocol utilizes the public transportation system like buses for real-time routing in SDN-assisted hybrid IEEE 802.11p and cellular networks [14]. To support and compensate for the vehicle's computational insufficiencies, in [9], a service offloading with deep learning is proposed, where a digital twin-empowered edge computing is utilized. The deep Q-network in the mechanism uses the replay memory and target networks to improve the value function approximation and achieves the offloading strategy. In [15], the SCMs are disseminated through hybrid medium access control (MAC) protocol. The protocol suppresses the collisions and strengthens the reliability, therefore offering an increased packet delivery ratio. An adaptive beacon generation rate (ABGR) mechanism is proposed in [16]. The ABGR influences the beacon generation based on the density of vehicles and attempts to minimize channel congestion. To offer highly reliable safety applications, the ABGR mechanism holds a reliability assessment scheme known as T-Pro. In [17], the SCMs are disseminated by capture-aware TDMA-based MAC (CT-MAC) protocol. By utilizing the capture effect, the CT-MAC protocol fixes an optimal frame length. The discrete Markov chain is incorporated to evaluate channel utilization. In [18], the vehicles are distributed as a nonhomogeneous Poisson process. Initially, the system evaluates the SCMs' delivery at intersections. Second, vehicles at extreme positions and hidden areas were obtained. Finally, a Bare Bones Particle Swarm Optimization (BBPSO) mechanism is proposed to adjust multitransmission factors in the SCM dissemination dynamically. Though there are various solutions for disseminating SCMs, the AVs cannot receive the SCMs on time without high-speed network access.

The cellular network is the fastest radio access technology for transmitting data with high reliability and low latency. However, due to the losses in links associated with topology changes, SCM dissemination's quality of service decreases. Alghamdi [19] overcomes this issue by transmitting the SCMs over device-to-device communication in C-V2X technology. The routing mechanism selects the best forwarder by utilizing the stable matching mechanism. Another solution in [20] utilizes 5G and mobile edge computing that enables V2V-based traffic offloading. This approach divides the problem into three subproblems and solves power allocation, channel assignment, and task distribution problems. Ghazi *et al.* [21] provide a detailed review of recent contributions in emergency message dissemination in vehicular networks in a 5G environment. They also highlight various proposed methods based on SDN and fog computing. Nkenyereye *et al.* [22] propose an SDN-based multiaccess edge computing framework for vehicular networks. In the proposed solution, two algorithms are implemented. First is a fuzzy logic-based algorithm used to select the head vehicle for each 5G base stations (gNBs) collocated with an RSU. Afterward, an OpenFlow algorithm is deployed to update flow tables of forwarding devices at forwarding layers.

The existing solutions bring more complexity and network congestion due to transmission of both beacon and network signals. Furthermore, solutions form clusters at every instance of SCM transmission. Hence, analyzing the benefits of SDN and 5G network, in this article, an SDN-assisted 5G V2X network in the MiCR framework is proposed for disseminating the SCMs to the AVs with high reliability and low latency.

III. System Model

The proposed MiCR framework aims to enhance the vehicular CEIs' performance by delivering the SCMs to the AVs with high reliability and low latency, especially in highway scenarios. MiCR is a three-tier architecture encompassing a centralized SDN controller in the first tier, gNBs and edge server in the second tier, and AVs in the third tier.

A. Overview

Each AV in MiCR is equipped with a chipset (say Qualcomm 9150 C-V2X) integrated with the global navigation satellite system (GNSS) [23]. The chipset is accountable for networking and computing messages, wherein the GNSS generates the digital map. Besides, each AV has a unique identity, either a pseudonym or a real number.

Initially, each AV in MiCR transfers its floating vehicle data such as unique ID, residual energy, location, velocity, direction, and destination in a quintuple format to the SDN controller. The quintuple has $\langle \text{AV}_{\text{ID}}, \varphi, \lambda,\ v_{i,t_i},\ \delta \rangle$, where AV_{ID} is the unique ID of the AV holding values from 1 to n for each gNBs, φ is the current position, λ is the direction, δ is the destination position, and v_{i,t_i} is the velocities at various time instants t_i. In MiCR, the SDN maintains a global database [24] (GDB_{AV}) to store and update the floating vehicle data. The GDB_{AV} retains a separate table for each gNBs to manage the network without any agitation effectively. Similarly, the edge server holds a local database (LDB_{AV}). Each table in the LDB_{AV} gets updated whenever the direction or the velocity changes. The edge server then applies the trained federated-K-means-clustering-inspired ConstructCR algorithm for creating moving CRs, which are the clusters comprising a single SAV and multiple member AVs (MAVs).

B. AVs' Mobility

Let the AVs initially move with a predefined travel time (T_M) to the destination. For a particular time slot, $t \in [1,\ T_M]$, the distance between the SDN controller (b) and the ith AV is given as

$$d_{b,i,t} = \left| (x,y)_{b,t} - (x,y)_{i,t} \right| < r_{\max}(a) \tag{1}$$

where $d_{b,i,t}$ is the distance between b and AV_i calculated at the time t, and $(x,y)_{b,t}$ and $(x,y)_{i,t}$ are the coordinates of the position of SDN controller (b) and the ith AV, respectively.

Similarly, the distance between SDN controller (b) and the gNB (a) is given as

$$d_{b,a,t} = \left| (x,y)_{b,t} - (x,y)_{a,t} \right| < \mathbb{R}_{\max}(a) \tag{2}$$

where $d_{b,a,t}$ denotes the distance between b and a at time t. Also, $(x,y)_{b,t} - (x,y)_{a,t}$ represents the location coordinates of b and a at time t. $\mathbb{R}_{\max}(a)$ indicates the maximum transmission range of the gNB a.

When the AV is driving on the real road environment, the unexpected circumstances such as accidents and congestion will actuate the AV to adjust the velocity. Thus, based on the Gauss–Markov mobility model [25], the velocity of the AV_i (v_i) at time t is given as

$$v_{i,t} = p_s \left(M_{v,t-1} \sqrt{1 - p_c^2} \right) + p_c v_{i,t-1} + (1 - p_c)\, \overline{v} \tag{3}$$

where p_c is the flexible velocity parameter that changes in accordance with the change in velocity and $\overline{v}$ is the mean of the velocity at time t. p_s represents the switching parameter to control the randomness. Furthermore, $M_{v,t-1}$ denotes the static Gaussian variable, which is independent of each other. The location of the AV_i is given as

$$(x,y)_{i,t} = (x,y)_{i,t-1} + \Delta(x,y)_i \tag{4}$$

where $\Delta(x,y)_i$ is the incremental vector of the position that combines two independent coordinates $\Delta(x)_i$ and $\Delta(y)_i$.

C. 5G V2X Channel Model

Considering the 3GPP suggested [26] line-of-sight (LOS) and non-line-of-sight (NLOS) data transmission, the path attenuation between AV $_i$ and j th gNB is

$$\tau_{i,j,t} = \begin{cases} d_{\text{ref}}^{L} (td_{i,j,t})^{-\xi L}, & \text{for LOS } P_L\,(td_{i,j,t}) \\ d_{\text{ref}}^{\text{NL}} (td_{i,j,t})^{-\xi \text{NL}}, & \text{for NLOS } P_{\text{NL}}\,(td_{i,j,t}) \end{cases} \tag{5}$$

where $td_{i,j,t}$ is the distance between AV $_i$ and j th gNB at time t represented in the 3-D coordinates. Also, P_L and P_{NL} is the path loss calculated when the distance between AV $_i$ and j th gNB at time t ($d_{i,j,t}$) is 1, with the path loss factors ξL and ξNL. Here, the probability of LOS and NLOS at time t is denoted as $P_L(td_{i,j,t})$ and $P_{\text{NL}}(td_{i,j,t})$, respectively. Thus, the value of $P_{\text{NL}}(td_{i,j,t})$ is estimated by,

$$P_{\text{NL}}\,(td_{i,j,t}) = \begin{cases} 5e^{\left(\frac{-s_1}{td_{i,j,t}}\right)}, & 0 < td_{i,j,t} < \overline{td} \\ 1 - 5e^{\left(\frac{td_{i,j,t}}{-s_2}\right)}, & td_{i,j,t} > \overline{td} \end{cases} \tag{6}$$

where $\overline{td} = \frac{s_1}{\ln 10}$, and s_1 and s_2 represent the shape parameters. Correspondingly, the value of $P_L(td_{i,j,t})$ is estimated by $1 - P_{\text{NL}}(td_{i,j,t})$.

With the gain values and the probability values of LOS and NLOS, the path loss between AV $_i$ and j th gNB at time t is updated as

$$\begin{aligned} \tau_{i,j,t} = {} & d_{\text{ref}}^{L} (td_{i,j,t})^{-\xi L} P_L\,(td_{i,j,t}) \\ & + d_{\text{ref}}^{\text{NL}} (td_{i,j,t})^{-\xi NL} P_{\text{NL}}\,(td_{i,j,t}) \end{aligned} \tag{7}$$

where $d^L_{\text{ref}}(td_{i,j,t})^{-\xi L}$ is the LOS gain and $d^N L_{\text{ref}}(td_{i,j,t})^{-\xi NL}$ is the NLOS gain. Thus, the 5G V2X communication channel model is formulated.

D. Data Transmission Model

In the MiCR framework, all the gNBs are considered to have a common signal (cs), which is transmitted to the AVs over the transmission region of the corresponding gNB. Thus, the signal received by AV$_i$ at time t is given as

$$DL_{i,t}=\sqrt{\wp}cg_{i,a,t}bv^n\mathbb{G}_t+\sum_{n'\neq n}\sqrt{\wp'}cg_{i,a,t}bv^{n'}\mathbb{G}_t'+z_{k,t} \quad (8)$$

where $\mathbb{G}_t$ is the information about the Gaussian distribution possessing, bv^n is the normalized beamforming vector, and $\wp$ denotes the variance. Besides, $z_{k,t}$ is the Gaussian noise with values $z_{k,t}\sim\mathcal{N}(0,\sigma_0^2)$. The channel gain $cg_{i,a,t}$ from the gNB to the AV$_i$ is

$$cg_{i,a,t}=\left[cg_{i,1,t},\dots cg_{i,|a|,t}\right]. \quad (9)$$

Accordingly, the AVs in the MiCR framework receive transmission signals from the gNB of the particular location with the assistance of SDN.

IV. Migrating Consignment Regions

For creating the CRs, the centralized federated K-means algorithm [27] based ConstructCR algorithm is exercised in MiCR. In ConstructCR, the SDN is used to orchestrate the various operations of the algorithm through the edge servers and establish coordination among them. The SDN then aggregates the model updates received from the edge servers.

Initially, the ConstructCR divides the AVs based on their directions. The direction angle θ_i of an AV$_i$, whose current position $\varphi=(x_i,y_i)$ and destination position $\delta=(x_i',y_i')$, is

$$\theta_i=\tan^{-1}\frac{y_i'-y_i}{x_i'-x_i}. \quad (10)$$

In the angular range $\theta_i\in[0°,90°)$, the AV moves in a northern plane denoted by λ_N. For $\theta_i\in[90°,180°)$, the ConstructCR algorithm switches the cardinality to the eastward λ_E. Progressing clockwise, the range $[180°,270°)$, denotes the southern λ_S. The remaining range $[270°,360°)$ forms the western λ_W. Once ConstructCR determines the direction, it randomly selects a set of k values as the centroids or the centers of the CRs, where $k\leq n$. To create nonoverlapping CRs, the ConstructCR iteratively figures out the squared Euclidean distance (SED) between AV and centroid and updates the centroid of a CR. The SED between AV$_i$ and the centroid C_k is

$$sd_{(\text{AV}_i,\,C_k)}=\|\varphi_i-\,C_k\|^2 \quad (11)$$

where φ_i is the position of the AV$_i$ and C_k is the centroid of CR$_k$. Subsequently, the centroid of each CR is updated by

$$C_k=\frac{1}{|C_k|}\sum_{\varphi_i\in C_k}\varphi_i. \quad (12)$$

To form optimal CRs, (11) and (12) get iterated until no AVs move from its CR. Consequently, the ConstructCR computes the radius of CR by

$$r_{(\text{CR}_k)}=\sqrt{\frac{1}{m}\sum_{p=1}^{m}\left[sd^i_{(\text{AV}_p,\,C_k)}\right]}. \quad (13)$$

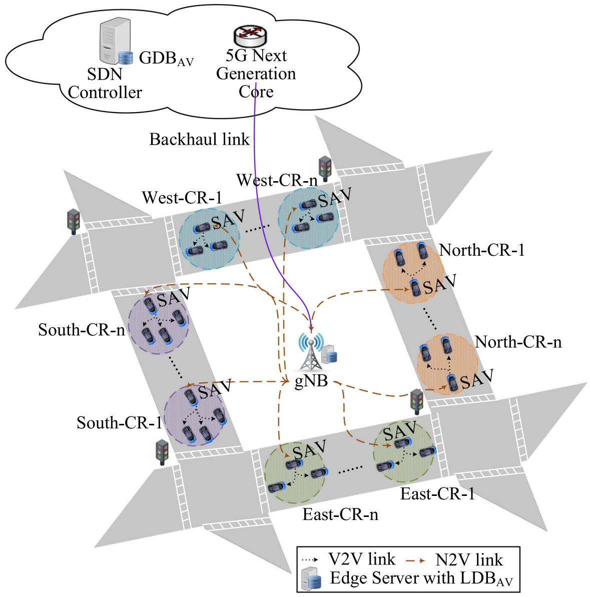


Fig. 1. MiCR architecture for SCM dissemination.

The ConstrutCR algorithm then selects an SAV for each CR consecutively. An optimal SAV should reach all the MAVs of the CR in one V2V communication. That is, the SAV should possess a minimum transmission range for the SCM dissemination. Thus, the transmission range and the position of the AVs are the critical features for ensuring the successful delivery of the SCMs. Therefore, the ConstructCR algorithm creates an empty set called $temp$ and stores the AVs nearer to the centroid C_k. The algorithm then selects the AV with a high transmission range to cover the MAVs in one V2V communication. After this, an optimal AV is chosen as an SAV by validating the sorted AVs against their velocity (v_{i,t_i}) over a small time-period ($\Delta t=t_j-t_i$).

In ConstructCR, the direction angle's determination ensures the moving direction, and the analysis of SED ensures the same moving velocity of the AVs in a CR. Thus, each AV in each CR moves in the same direction and with almost the same velocity. Furthermore, the optimal SAV selection ensures the highly successful delivery ratio of the SCMs. Finally, the ConstructCR algorithm appends the CRs of four directions α, β, γ, and ε and updates the k value by

$$k=C_k^{\alpha}\cup C_k^{\beta}\cup C_k^{\gamma}\cup C_k^{\varepsilon},\quad k<n. \quad (14)$$

Thus, the ConstructCR algorithm partitions n AVs of a gNB into k nonoverlapping CRs and selects an optimal SAV for each CR. Algorithm 1 represents the pseudocode for the ConstructCR algorithm. Fig. 1 depicts the architecture of MiCR. As shown

Algorithm 1: ConstructCR.

```
Input: Number of AVs $n$, Velocity of the AVs $v_{i,t_i}$,
  Direction of the AVs $\lambda$
Output: Number of CRs $k$
while $loc(\mathrm{AV}_i) \in loc(gNB_i)$ do
  Compute direction by (10)
  Assign $\alpha \leftarrow \mathrm{AV_{ID}} \in \lambda_N$, $\beta \leftarrow \mathrm{AV_{ID}} \in \lambda_S$,
  $\gamma \leftarrow \mathrm{AV_{ID}} \in \lambda_W$, and $\varepsilon \leftarrow \mathrm{AV_{ID}} \in \lambda_E$
  for $x = \alpha, \beta, \gamma$, and $\varepsilon$ do
    get Velocity of $\mathrm{AV}_i$ $(v_{i,t_i})$
    procedure Server-ConstructCR
      Initialize number of CRs, $k = 0$
      Initialize set of CRs, $\mathrm{CR} = \{\emptyset\}$
      Centroids $k \leftarrow \mathrm{Random}(n)$, where $k < n$
      for $i = 1$ to $k$ do
        Assign centroid as, $C_i \leftarrow i$
        repeat
          Compute $sd_{(\mathrm{AV_{ID}},\, C_i)} = \|\varphi_{\mathrm{ID}} - C_i\|^2$;
          Assign $C_i \leftarrow \mathrm{AV_{ID}}$ closest to $C_i$
          Compute $C_i = \frac{1}{|C_i|} \sum_{\mathrm{AV_{ID}} \in C_i} \mathrm{AV_{ID}}$;
          Update centroid $C_i$
        until $C_i$ is not updated
        Compute $r_{(\mathrm{CR}_k)} = \sqrt{\frac{1}{m} \sum_{p=1}^{m} [sd^i_{(\,\mathrm{AV}_p,\, C_k)}]}$;
        temp $\leftarrow \{\{\mathrm{AV}_1, \mathrm{AV}_2, \ldots, \mathrm{AV}_i\} \approx \varphi(C_k)$
        repeat
         for $\forall \mathrm{AV_{ID}} \in$ temp do
           Choose $\mathrm{AAV_{ID}}$ having $\max(\mathbb{R})$
           if $\mathrm{AAV_{ID}}$ has same velocity as $C_k$ then
             $\mathrm{SAV_{ID}} \leftarrow \mathrm{AAV_{ID}}$
               else
               temp $\leftarrow$ temp $-$ $\mathrm{AAV_{ID}}$;
        until velocity matches
      return $C_i$, $\mathrm{SAV_{ID}}$ and $r_{(\mathrm{CR}_k)}$
      $\mathrm{CR}_k^x \leftarrow C_i$

      return CR for a direction $\mathrm{CR}_k^x$
      Add the current CR to the CR set, CR$\leftarrow$ CR $\cup$ $\mathrm{CR}_k^x$
      Number of CRs, $k = n(\mathrm{CR})$;
      return $k$

    procedure Client-ConstructCR
     Initialize the radius of CR $r(\mathrm{CR}_k) = 0$
     for $\forall \mathrm{AV_{ID}} \in$ temp do
      for $\forall$direction $\in$ temp do
        $\alpha, \beta, \gamma$, and $\varepsilon = \arg\min_k \|\mathrm{AV_{ID}} - k\|^2$;
        for each CR's centroid $C_i$ do
          Compute $k = \sum_{i=1}^{k} = 1_{\{\alpha=k\}}.\mathrm{AV}_i$;
          $r_{(\mathrm{CR}_k)} = \sqrt{\frac{1}{m} \sum_{p=1}^{m} [sd^i_{(\,\mathrm{AV}_p,\, C_k)}]}$;
          Number of CRs, $k = n(\mathrm{CR})$;
    return $k$
```

in Fig. 1, the SDN creates CRs by considering the direction and velocity on which the AV moves.

A. CR Maintenance

The stability of the created CRs is comparatively high, since the MiCR estimates and creates the CRs with AVs possessing similar direction and velocity. However, the dynamic nature of AVs induces the maintenance of the CRs, when the GDB$_{AV}$ of SDN is updated.

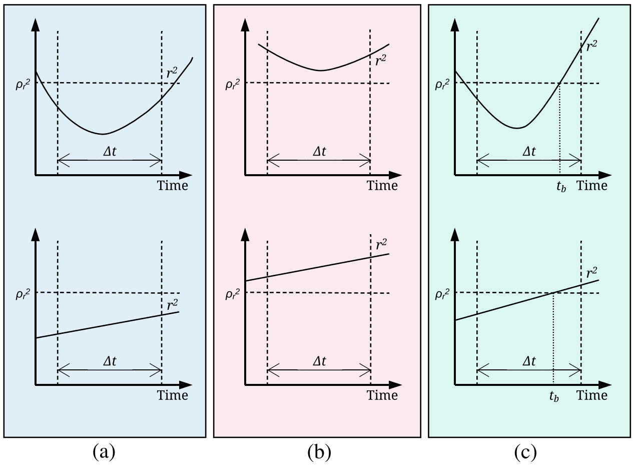


Fig. 2. Relationship between r^2 and ρ_r^2 at time period Δt. (a) $\rho_r^2 > r^2$, (b) $r^2 > \rho_r^2$, and (c) $r^2 > \rho_r^2$ at t_b.

1) Breaking the CR: A formed CR breaks for two significant cases.

a) Case 1: When the CR beats its compactness, the formed CR breaks. That is, when the radius $r_{(\mathrm{CR}_k)}$ of each CR exceeds the radius threshold value, the CR breaks. The radius threshold (ρ_r) is

$$\rho_r = \frac{1}{4}\sqrt{S_{\mathrm{CR}}} \tag{15}$$

where S_{CR} is the average size of the CR following uniform distribution, which is computed as

$$S_{\mathrm{CR}} = \frac{Area}{\left(\frac{n}{g}\right)} \tag{16}$$

where $Area$ is the total area covered by a gNB, n is the total number of AVs that belong to the gNB, and g is the default CR capacity, which is the proportion of four-tenth of the total number of AVs. The following equation gives the expression of g for estimating the average default capacity of a CR:

$$g = \frac{1}{10}\,(4 * n)\,. \tag{17}$$

The comparison of the radius of CR $(r_{(\mathrm{CR}_k)})$ with the radius threshold (ρ_r) analytically predicts the time at which the CR breaks. Fig. 2 shows the three possible types of relationship between r^2 and ρ_r^2 for a specific time period Δt, where Δt is the change of time from one instant t_x to another instant t_y. In Fig. 2(a), the square of the radius r^2 is persistently inferior to the radius threshold ρ_r^2, implying that CR_k will not break during the time limit Δt. In Fig. 2(b), r^2 outstrips ρ_r^2, which implies a faster break of the CR. In Fig. 2(c), r^2 outstrips ρ_r^2 at a specific break time (say t_b), which implies the break of CR at t_b.

b) Case 2: When the direction of any AV in a CR does not match with the direction of the CR or when the radius

Algorithm 2: BreakCR.

```
Input: Number of AVs $n$, Number of CRs $k$, Velocity of the AVs $v_{i,t_i}$, Direction of the AVs $\lambda$
Output: Updated number of CRs $k$
procedure BreakCR
  while CR is not null do
    if radius exceeds $\rho_r$ or time reaches $t_b$ then
      Store the AVs in a temporary set AV$_{\text{temp}}$
      for $\forall$ AVs in AV$_{\text{temp}}$ do
        ConstructCR$(n(\text{AV}_{\text{temp}}), v_{\text{temp},t_b}, \lambda)$
    if any AVs (AV$_x$) changes its direction then
      for $\forall$AV$_x$ do
        if SED for AV$_x$ is less than $sd_\Omega$ then
          if direction and velocity matches then
            UpdateCR$(\text{AV}_i, k, v_{i,t_i}, \lambda)$
          else
          for $\forall$AV$_{\text{ID}}$ do
            ConstructCR$(n, v_{i,t_i}, \lambda)$
    Update number of CRs $k$
return $k$
endprocedure
```

Algorithm 3: CombineCR.

```
Input: Number of AVs $n$, Number of CRs $k$, Velocity of the AVs $v_{i,t_i}$, Direction of the AVs $\lambda$
Output: Updated number of CRs $k$
procedure CombineCR
  while CR is not null do
    if direction and velocity of two CRs matches then
      if SED of two CRs is less than $sd_\Omega$ then
        New CR, CR$_z$ = CR$_i$ $\cup$ CR$_j$;
        Centroid $C_Z \leftarrow$ Random$(k)$
        repeat
          Find SED between AVs and $C_Z$;
          Consign AVs to CR$_z$
          Update $C_Z$
        until no updation in $C_Z$
        return $C_Z$
        Add $C_Z$ to the set of CR CR $\leftarrow$ CR $\cup$ $C_Z$;
        Number of CRs $k = n(\text{CR})$;
        else
        UpdateCR$(\text{AV}_i, k, v_{i,t_i}, \lambda)$
      else
      ConstructCR
    return $k$
endprocedure
```

TABLE I
SIMULATION PARAMETERS

Parameter	Value
MAC protocol	SC-FDMA
SCM packet size	512 bytes
Simulated area	2500 x 500 m
Minimum speed of AVs	20 km/h
Maximum speed of AVs	150 km/h
Maximum transmission range	400 m
Data rate	$>$ 100 Mbps
Spectrum band	5.9 GHz
Environment	Highways
5G Base Stations (gNBs)	15
SDN Controller	1
SDN Protocol	OpenFlow
Simulation time	6000 s

threshold exceeds, the proposed BreakCR algorithm checks for the radius of CR$_i$. If the radius exceeds the radius threshold ρ_r, then it segregates the AVs from the centroid position to the end of the CR as AV$_{\text{temp}}$. The algorithm then creates a new CR for the AVs in AV$_{\text{temp}}$. Similarly, the algorithm checks if any AVs that belong to CR$_i$ change their direction. In this case, the BreakCR algorithm tries to insert them into the closest CR (say CR$_j$) by matching their direction and velocity and by estimating the SED between AV and the centroid C_j. When the estimated SED is less than the SED threshold (sd_Ω) and the direction and velocity matches, BreakCR inserts the AV into CR$_j$ via the UpdateCR algorithm. If none of the AVs match the nearby CR's direction and velocity, the BreakCR algorithm creates new CRs based on their directions and velocity. The pseudocode for the BreakCR algorithm is given in Algorithm 2.

2) Combining the CRs: When two differently directed CRs are ahead in the same direction, then SDN induces the CombineCR algorithm to merge the CRs. The CombineCR algorithm considers three crucial factors for combining the CRs. The first factor is the distance between moving CRs (CR$_i$ and CR$_j$), the second is the velocity of AVs in the CRs, and the third is the direction of the CRs. Initially, the CombineCR algorithm checks whether any two different CRs moving in the same direction have low values of SED. If the SED among them is less than the SED threshold sd_Ω, the algorithm checks for the velocity matching among the AVs in both CRs. The velocity matching constraints fall into two significant cases.

Case 1: All of the AVs have similar velocities. In this case, the CombineCR algorithm merges the two CRs (CR$_i$ and CR$_j$) and forms a new CR (CR$_z$).

Case 2: Some of the AVs may have different velocities. In this case, the CombineCR algorithm adds the AVs to the CRs having similar speeds with lesser SED than the SED threshold. If none of the CRs have the same velocity as the AVs, it creates a separate new CR for those AVs. The pseudocode for the CombineCR algorithm is given in Algorithm 3.

3) Updating CR: We propose an UpdateCR algorithm to insert or remove an AV in or from a CR, respectively. The threshold sd_Ω defines the utmost potential of the SED distance, which prevents the CRs from frequently changing, and it is illustrated as

$$sd_\Omega = 2\sqrt{S_{\text{CR}}} \tag{18}$$

where S_{CR} is the average size of the CR, the idea behind the definition of sd_Ω is that if the distance between two AVs AV$_x$

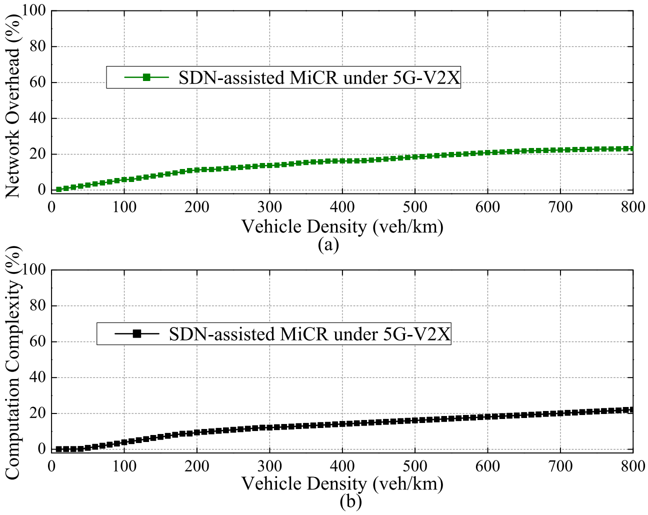


Fig. 3. (a) Evaluation of MiCR network overhead. (b) Evaluation of MiCR computational complexity in SCM dissemination with respect to the density of AVs.

Algorithm 4: UpdateCR.

```
Input: AV to be inserted AV$_i$, Number of CRs $k$, Velocity of the AVs $v_{i,t_i}$, Direction of the AVs $\lambda$
Output: Updated number of CRs $k$
procedure UpdateCR
  while CR is not null do
    if SED of AV$_i$ exceeds $sd_\Omega$ of CR$_i$ then
      Remove AV$_i$ from CR$_i$ and update CR
    return $k$
    if SED and velocity of AV$_i$ matches with CR$_j$ then
      Add AV$_i$ to CR$_j$ and update CR
      else
      Create new CR, CR$_{\text{new}} \leftarrow$ ConstructCR
      Update set of CRs, CR $\leftarrow$ CR $\cup$ CR$_{\text{new}}$;
    Number of CRs, $k = n(\text{CR})$;
    return $k$
endprocedure
```

and AV$_y$ is greater than twice the diameter of the CR, then the AVs most possibly belong to two different CRs CR$_i$ and CR$_j$. Thus, sd_Ω assists the UpdateCR algorithm for straightforwardly inserting the AV into the CR, which in turn reduces the computational overhead. Also, when the SED between the AV and CR exceeds sd_Ω, the UpdateCR algorithm removes the AV from the respective CR and inserts it into the CR, which matches its direction and has lesser sd_Ω. If both the cases fail, the UpdateCR algorithm creates a new CR with the removed AV as a candidate. Algorithm 4 displays the outline of the UpdateCR algorithm.

B. Transferring MiCR Information to Next gNB

When the CRs move out of coverage of one gNB (gNB$_i$) and enters into another (gNB$_j$), the feature set of the particular CR will be transferred from the gNB$_i$ to the next gNB gNB$_j$. Thus, MiCR gradually minimizes the network overhead in creating new CRs, when the AVs of CR$_i$ move to the coverage of another gNB.

V. Performance Evaluation

The performance of MiCR is analyzed by executing a set of experiments on a real-time map, which further ensures the performance of vehicular CEI. The outcomes of the experiments are compared with the existing approaches: ABGR [16], TB-EM [8], BBPSO [18], and DCRB [14].

A. Scenarios and Simulation Parameters

We utilize the widely used Simulation of Urban Mobility to simulate AVs' mobility on the real-time map obtained from OpenStreetMap. The OpenStreetMap delivers global map records with excellent quality in a unified tagging schema. For simulating the network and communication control of SDN among the AVs, we utilize the OpenFlow module of Network Simulator (NS, version 3.28) by utilizing the MAC protocol single-carrier frequency-division multiple access (SC-FDMA).[1] The NS3 has a special feature that it can act as both a simulator and an emulator; in addition, the results obtained will be 90% suitable for the real-world scenario. To evaluate the performance of MiCR, we simulate the real highway environment along the roads of three cities: Washington, New York, and Los Angeles. On each map, we simulate a maximum of 800 vehicles. The density of the AV directly depends on the capacity of the road. Table I displays the parameters used for simulating MiCR.

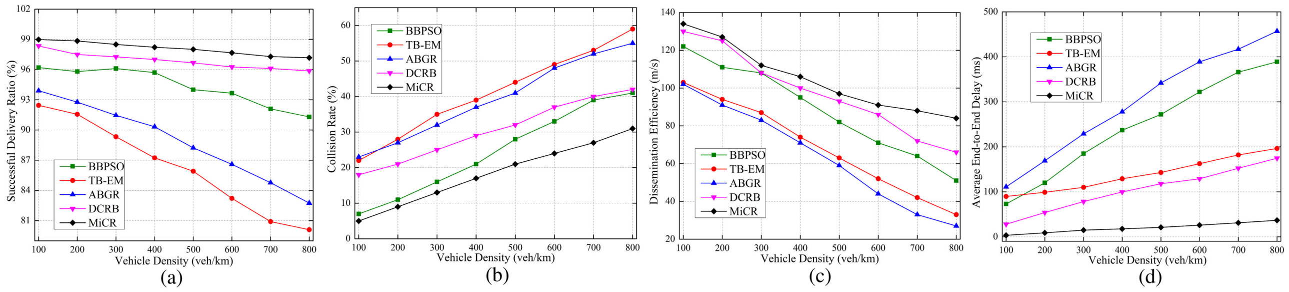
B. Results and Discussions

We evaluate the performance of MiCR by varying the AV density and velocity. Thus, MiCR is evaluated on its effectiveness and the efficiency based on the following performance metrics.

1) Complexity Analysis of Federated-K-Means-Based ConstructCR Algorithm: The ConstructCR algorithm begins with a single-step operation for initializing the parameters, and the time complexity is $O(1)$. Creating the CRs based on the server and client procedures takes the complexity of $O(n \log n)$. Thus, the ConstructCR algorithm has the best-case complexity of $O(1)$ and the worst-case complexity of $O(n \log n)$.

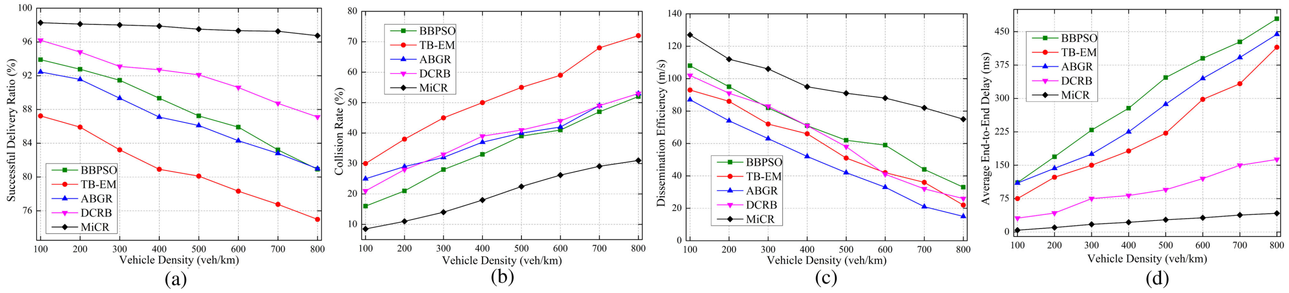
2) Network Overhead and Computational Complexity: By network overhead, we refer to the redundant demand on network resources such as time, bandwidth, energy, and memory imposed by our framework. Computational complexity is an additional measure of cost, which is used in the evaluation. Both the metrics have been evaluated for our proposed MiCR approach across a range of vehicle densities, and the results are plotted in Fig. 3. The MiCR maintains the created CRs perpetually and disseminates the SCMs via the parallel multipoint transmission approach. When the AVs move toward the next gNB, the MiCR transfers the features of CRs to the next gNB. Initially, when n =10, MiCR has a network overhead of 0.5% and computational complexity of 0.1%. When the density of AVs reaches a medium peak of 400 to 500, MiCR reaches the network overhead

[1] 3GPP, "Architecture enhancements for V2X services," Third Generation Partnership Project, Tech. Rep., May 2017



(a) (b) (c) (d)

Fig. 4. Performance comparison of MiCR framework with existing approaches. (a) Successful delivery ratio. (b) Collision rate. (c) Dissemination efficiency. (d) Average end-to-end delay in SCM dissemination with respect to the density AVs.



(a) (b) (c) (d)

Fig. 5. Performance comparison of MiCR framework with existing approaches at intersections and roundabouts. (a) Successful delivery ratio. (b) Collision rate. (c) Dissemination efficiency. (d) Average end-to-end delay in SCM dissemination with respect to the density AVs.

TABLE II
COMPARATIVE ANALYSIS OF PACKET DROP RATIO

AV Density (AV/km)	BBPSO (%)	TB-EM (%)	ABGR (%)	DCRB (%)	Mi-CR (%)
100	7	22	23	18	5
200	11	28	27	21	9
300	16	35	32	25	13
400	21	39	37	29	17
500	28	44	41	32	21
600	33	49	48	37	24
700	39	53	52	40	27
800	41	59	55	42	31

of 10.25% and 14.02% computational complexity. When the number of AVs hits a maximum density of 800, MiCR utilizes the network resources up to 21%, and the complexity is of 19.8%.

3) Comparative Analysis of MiCR With Existing Approaches: The MiCR and the existing approaches are simulated and analyzed for different numbers of AVs, starting from 100 to 800. Table II displays the various packet drop ratios for MiCR and the existing approaches. From the table, it is inferred that, due to the extensive performance and low complexity in creating and maintaining the CRs, the MiCR possesses a low packet drop ratio.

Figs. 4 and 5 show the comparative analysis of the MiCR approach with the existing techniques at straight road structures and extreme cases such as roundabouts and intersections. Figs. 4(a) and 5(a) show the comparison of the successful delivery ratio of the SCMs against the density of the AVs. Compared to MiCR, TB-EM has a 7.07% and 10.5% lower rate of successful delivery. However, as the number of AVs increases, all the existing approaches except DCRB lessen profoundly in the successful delivery ratio by an average of 1.8% and 3.2% compared to MiCR. At a greater number of AVs ($n = 800$), the delivery ratio of MiCR exceeds TB-EM by 17.53% and 18.25%. Thus, TB-EM has the lowest successful delivery ratio than MiCR concerning the number of AVs, and MiCR outperforms the existing approaches by 12.71% and in roundabouts by 11.82%. Among the existing approaches, ABGR possesses a better delivery ratio as like MiCR. This is because both approaches utilize SDN-assisted cellular-based communications. However, the delivery ratio of ABGR lags when GPS location-sharing services do not exist.

Figs. 4(b) and 5(b) show the comparison of the collision rate of MiCR with the existing approaches. When compared with the existing approaches, MiCR has a difference of 54% initially, and when $n = 200$, MiCR outstrips them by an average of 49%. The performance difference changes to 47% for $n = 300$, which further changes to 44%, 35.3%, 35.1%, 29.7%, and 32.56% for the vehicle density of 400, 500, 600, 700, and 800 respectively. On average, MiCR transcends the existing approaches by 40.83%, as the SAVs alone transmit the SCMs to the MAVs. It is also observed that the ABGR has the next lower packet collision ratio; this is because it has a CSMA/CA mechanism that senses the idle wireless channel and books a mini-slot from it. Among the remaining techniques, TB-EM has a higher collision ratio as it has an initial delay in selecting the CH for the clusters.

Figs. 4(c) and 5(c) show the comparison and analysis of the dissemination efficiency of various approaches. The figures show that when $n = 100$, MiCR has a dissemination efficiency of about 134 m/s, which changes to 106 m/s for $n = 400$ and to

84 m/s for $n = 800$. Compared with the ABGR and TB-EM, the dissemination efficiency of MiCR differs by about 29.78% when $n = 100$ and by 66.6% when $n = 800$. Thus, the dissemination efficiency of MiCR surpasses the existing approaches by an average of 48.76%.

Figs. 4(d) and 5(d) show the comparison of MiCR with the existing approaches in terms of average end-to-end latency. Initially, MiCR has a latency of 134 ms when the vehicle density increases to 500; MiCR possesses a latency of 311 ms, which reaches a maximum of 467 ms for $n = 800$. Compared with the existing approaches, MiCR debacles with an average of 24.8% when $n = 100$ and by 46.7% when $n = 800$. However, DCRB is the next approach that possesses low latency in SCM dissemination. This is because DCRB utilizes the frequently available public transportation system in urban areas to relay the SCMs. On the contrary, ABGR has very high latency, as the approach follows a multihop fashion for the transmission of SCMs.

The above performance analysis justifies that MiCR increases delivery ratio and dissemination efficiency by reducing the latency and collision rate of the SCMs. The MiCR framework amends the existing approaches' performance gaps and is the most suitable for disseminating SCMs to the AVs. Thus, MiCR proved its efficiency in delivering SCMs and ensured the vehicular CEI with high safety.

VI. Conclusion

In this article, the significance of SCM dissemination for vehicular CEI, especially to the fast-moving AVs, was detailed with ConstructCR, BreakCR, CombineCR, and UpdateCR algorithms in the proposed MiCR framework. This envisioned framework uses moving clusters and a range of CR algorithms to improve the dynamic AV environment significantly. The results obtained in the simulations showed that MiCR outperforms the current methods by 18.3% concerning the high successful delivery ratio of SCMs with very low latency even under extreme circumstances. Thus, the MiCR framework delivers the SCMs to the AVs efficiently, thereby enhancing the safety of the vehicular CEI. The future issues of AVs are the extensive vehicular applications that provide personalized services, which are associated with various challenges such as computation, communication, and storage. The federated learning approach can be incorporated in AV networks to address these challenges and provide extraordinary intelligent services.

References

[1] S. H. Bouk, S. H. Ahmed, Y. Eun, and K.-J. Park, “Multimodal named data discovery with interest broadcast suppression for vehicular CPS,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 5, pp. 1877–1891, May 2021.

[2] A. Ali *et al.*, “Priority-based cloud computing architecture for multimedia-enabled heterogeneous vehicular users,” *J. Adv. Transp.*, vol. 2018, pp. 1–12, 2018.

[3] J. Qiu, D. Grace, G. Ding, M. D. Zakaria, and Q. Wu, “Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms,” *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 140–148, Dec. 2019.

[4] G. Raja, A. Ganapathisubramaniyan, S. Anbalagan, S. B. M. Baskaran, K. Raja, and A. K. Bashir, “Intelligent reward-based data offloading in next-generation vehicular networks,” *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3747–3758, May 2020.

[5] G. Manogaran *et al.*, “A response-aware traffic offloading scheme us-ing regression machine learning for user-centric large-scale Internet of Things,” *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3360–3368, Mar. 2021.

[6] S. Anbalagan, D. Kumar, M. F. J., G. Raja, W. Ejaz, and A. K. Bashir, “SDN-assisted efficient LTE-WiFi aggregation in next genera-tion IoT networks,” *Future Gener. Comput. Syst.*, vol. 107, pp. 898–908, 2020.

[7] S. Li, Y. Liu, and J. Wang, “An eefficient broadcast scheme for safety-related services in distributed TDMA-based VANETs,” *IEEE Commun. Lett.*, vol. 23, no. 8, pp. 1432–1436, Aug. 2019.

[8] S. S. Shah, A. W. Malik, A. U. Rahman, S. Iqbal, and S. U. Khan, “Time barrier-based emergency message dissemination in vehicular ad-hoc networks,” *IEEE Access*, vol. 7, pp. 16 494–16503, 2019.

[9] X. Xu *et al.*, “Service offloading with deep Q-network for digital twinning empowered internet of vehicles in edge computing,” *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2020.3040180.

[10] Z. Zhou, F. Xiong, C. Xu, Y. He, and S. Mumtaz, “Energy-efficient vehicular heterogeneous networks for green cities,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1522–1531, Apr. 2018.

[11] X. Wang, Z. Ning, and L. Wang, “Offloading in Internet of vehicles: A fog-enabled real-time traffic management system,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4568–4578, Oct. 2018.

[12] S. Kim and B.-J. Kim, “Crash risk-based prioritization of basic safety message in DSRC,” *IEEE Access*, vol. 8, pp. 211961–211972, 2020.

[13] W. Liu and Y. Shoji, “DeepVM: RNN-Based vehicle mobility prediction to support intelligent vehicle applications,” *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 3997–4006, Jun. 2020.

[14] J. Wang, K. Liu, K. Xiao, X. Wang, Q. Han, and V. C. S. Lee, “Delay-constrained routing via heterogeneous vehicular communications in software defined BusNet,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5957–5970, Jun. 2019.

[15] M. Zhang, G. M. N. Ali, P. H. J. Chong, B.-C. Seet, and A. Kumar, “A novel hybrid MAC protocol for basic safety message broadcasting in vehicular networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4269–4282, Oct. 2020.

[16] W. Li, W. Song, Q. Lu, and C. Yue, “Reliable congestion control mechanism for safety applications in urban VANETs,” *Ad Hoc Netw.*, vol. 98, 2020, Art. no. 102033.

[17] Y. Wang, J. Shi, L. Chen, B. Lu, and Q. Yang, “A novel capture-aware TDMA-based MAC protocol for safety messages broadcast in vehicular ad hoc networks,” *IEEE Access*, vol. 7, pp. 116542–116554, 2019.

[18] W. Yanbin, W. Zhuofei, Z. Jing, L. Zhijuan, and M. Xiaomin, “Analysis and adaptive optimization of vehicular safety message communications at intersections,” *Ad Hoc Netw.*, vol. 107, 2020, Art. no. 102241.

[19] S. A. Alghamdi, “Cellular V2X with D2D communications for emergency message dissemination and QoS assured routing in 5G environment,” *IEEE Access*, vol. 9, pp. 56049–56065, 2021.

[20] Z. Ning, X. Wang, J. J. Rodrigues, and F. Xia, “Joint computation offloading, power allocation, and channel assignment for 5G-enabled traffic management systems,” *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3058–3067, May 2019.

[21] M. U. Ghazi, M. A. K. Khattak, B. Shabir, A. W. Malik, and M. Sher Ramzan, “Emergency message dissemination in vehicular networks: A review,” *IEEE Access*, vol. 8, pp. 38 606–38621, 2020.

[22] L. Nkenyereye, L. Nkenyereye, S. M. R. Islam, C. A. Kerrache, M. Abdullah-Al-Wadud, and A. Alamri, “Software defined network-based multi-access edge framework for vehicular networks,” *IEEE Access*, vol. 8, pp. 4220–4234, 2020.

[23] RBRi, “Self-driving cars and policy,” *Robot. Bus. Rev.*, Nov. 2017. [Online]. Available: https://www.roboticsbusinessreview.com/wp-content/uploads/2018/03/RBRi_SelfDrivingCars_WP-Report.pdf

[24] X. Duan, Y. Liu, and X. Wang, “SDN enabled 5G-VANET: Adaptive vehicle clustering and beamformed transmission for aggregated traffic,” *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 120–127, Jul. 2017.

[25] B. Xiong, Z. Zhang, J. Zhang, H. Jiang, J. Dang, and L. Wu, “Novel multi-mobility V2X channel model in the presence of randomly moving clusters,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 5, pp. 3180–3195, May 2021.

[26] B. Peng, K. Guan, A. Kuter, S. Rey, M. Patzold, and T. Kuerner, “Channel modeling and system concepts for future terahertz communications: Getting ready for advances beyond 5G,” *IEEE Veh. Technol. Mag.*, vol. 15, no. 2, pp. 136–143, Jun. 2020.

[27] Y. Liu, Z. Ma, Z. Yan, Z. Wang, X. Liu, and J. Ma, “Privacy-preserving federated K-means for proactive caching in next generation cellular networks,” *Inf. Sci.*, vol. 521, pp. 14–31, 2020.