

Software-Defined Networking (SDN) based VANET Architecture: Mitigation of Traffic Congestion

Tesfanesh Adbeb¹
School of Computer Science and
Technology
Dalian University of Technology
Dalian, China

Wu Di²
School of Computer Science and
Technology
Dalian University of Technology
Dalian, China

Muhammad Ibrar³
School of Software
Dalian University of Technology
Dalian, China

Abstract—In VANETs (Vehicular Ad-hoc Networks), the number of vehicles increased continuously, leading to significant traffic problems like traffic congestion, a feasible path, and associated events like accidents. Though, the Intelligent Transportation System (ITS) providing excellent services, such as safety applications and emergency warnings. However, ITS has limitations regarding traffic management tasks, scalability, and flexibility because of the enormous number of vehicles. Therefore, extending the traditional VANET architecture is indeed a must. Thus, in the recent period, the design of the SD-VANETs (Software-Defined Networking defined VANETs) has gained significant interest and made VANET more intelligent. The SD-VANET architecture can handle the aforesaid VANET challenges. The centralized (logically) SDN architecture is programmable and also has global information about the VANET architecture. Therefore, it can effortlessly handle scalability, traffic management, and traffic congestion issues. The traffic congestion problem leads to longer trip times, decreases the vehicles' speed, and prolong average end-to-end delay. Though, somewhere, some routes in the network are available with capacity, which can minimize the congestion problem and its characteristics. Therefore, we proposed heuristic algorithms called Congestion-Free Path (CFP) and Optimize CFP (OCFP), in SD-VANET architecture. The proposed algorithms address the traffic congestion issue and also provide a feasible path (less end-to-end delay) for a vehicle in VANET. We used the NS-3 simulator to evaluate the performance of the proposed algorithms, and for generating a real scenario of VANET traffic; we use the SUMO module. The results show that the proposed algorithms decrease road traffic congestion drastically compared to exiting approaches.

Keywords—Software-Defined Networking; VANET; congestion; feasible path; NS3; SUMO

I. INTRODUCTION

In the current communication era, VANETs have received the significant attention of the researchers because of its unique and critical characteristics like frequent changes in topology, link failure, network stability, efficient traffic management, safety, congestion, and reliability [1], [2]. The characteristics, as mentioned earlier, lead to network instability because of high vehicle mobility. Thus, high vehicle mobility yields overall network efficiency, create road-side safety, and security issues. Therefore, ITS deployment is required to handle the enormous traffic efficiently, avoid congestion, reliability, and also provide the services to the passengers (like safety applications, emergency warnings, video streaming, lane change warning, and entertainment). These types of services, as mentioned before, need efficient and improved Packet Delivery

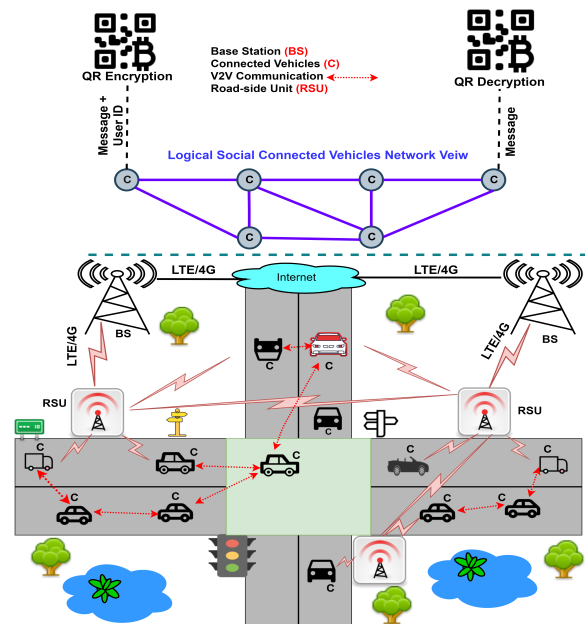


Fig. 1. V2X Architecture

Ratio (PDR), need high-quality communication, congestion-free path, and average end-to-end delay.

ITS is a vital next-generation transportation system [3], [4], and it is a combination of communication technologies used in VANET management (i.e., efficiency, safety, and sustainability) and leading-edge information. In VANET, vehicles are like mobile nodes. They collect and disseminate information about their speed, current position, destination [5], [6], [7], [8]. In some emergency conditions, such as health issues, road accidents, and congestion, the VANET architecture (ITS) ensure the driving safety, alternative routes, and timely report. Therefore, through V2X (Vehicle-to-Everything) [9] architecture, it is possible to inform the nearby vehicles in a specific area to avoid congestion, emergency conditions, and provide the alternate reliable route, as shown in Fig. 1. Noticeably, under some traffic conditions, the shortest route can lead to the congestion problem. The congestion problem leads to longer trip times, decreases the vehicle's speed, and prolong end-to-end delay [10]. Although, somewhere, some routes in the network are available with capacity, which can minimize the congestion problem and its characteristics. The

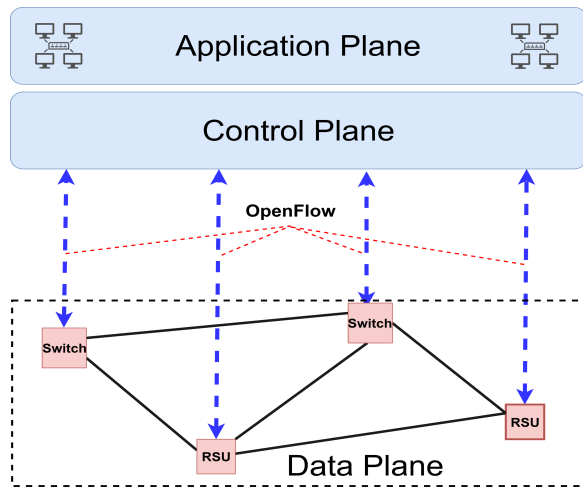


Fig. 2. SDN-based Architecture

goal of traffic engineering is to make sure that traffic is managed such that community ability is utilized efficiently and in a balanced manner. There are several techniques to handle Traffic Engineering (TE) problems like congestion and delay in VANETs.

To overcome the congestion problem in VANETs and its characteristics (i.e., prolong end-to-end delay, longer trip times, an emergency condition), we look at controlled (logically) SDN architecture. The programmable SDN architecture provides a flexible way to manage and control the traditional VANET architecture systematically. The main objective of programmable and logically centralized SDN architecture is to decouple the control plane from the data plane [11], [12], [13]. In SDN architecture, the unified controller (control plane) is responsible for monitoring, controlling, and managing the network resources efficiently. The purpose of the SDN controller is to improve and optimize the overall network performance like path selection, traffic control, congestion control, and efficient communication. The data plane is a networking infrastructure, forwarding devices (switch, router, Access Point (AP), and Roads-side Unit (RSU)) used for the data forwarding process. These forwarding devices connect with wired or wireless channels.

The SDN controller uses the OpenFlow protocol [11] for communication with data forwarding devices, as shown in Fig. 2. The proposed SD-VANET architecture consists of the following components. (a) Control Plane: the logically unified SDN controller that provides control functionalities about the entire network. The controller communicates with network devices in the data plane, and with the application. (b) Data Plane: The data plane consists of the vehicles (i.e., SDN-enabled wireless mobile nodes) and RSU (i.e., SDN-enabled stationary nodes) that receive the control message from the control plane. Each SDN-enabled wireless node contains a local agent, called SDN-agent. The SDN-agent is used to communicate with the controller.

The current approaches focused on the performance of SD-VANET routing, link stability, heterogeneity, offloading, and mobility. However, these approaches do not distribute the vehicles based on vehicles' density proactively; therefore, which causes congestion and prolongs the end-to-end delay. Congestion also leads to traffic accidents [1], [2], [13], [5]

and also decreases the packet delivery ratio and QoS (Quality-of-Services). Motivated by the congestion problem and its characteristics in VANET, as mentioned earlier, we proposed new heuristic algorithms called Congestion Free Path (CFP) and Optimized CFP (OCFP) in the SD-VANET architecture. More explicitly, in this paper, our main contributions are summarized as follows.

- Prominently, the centralized (logically) SDN controller can manage, control, and provide flexible communication between V2I (Vehicle-to-Infrastructure) and V2V (Vehicle-to-Vehicle). More specifically, in SD-VANET architecture, from the global (abstract) view of the VANET, the centralized controller compute the congestion-free and optimal path for a vehicle.
- In this paper, we proposed heuristic algorithms called CFP and OCFP in the SD-VANET architecture. More specifically, these algorithms proactively distribute the vehicles on roads according to vehicles' density and compute the congestion-free path and its characteristics such as long queuing delay, longer trip times, decrease the vehicle's speed, and safety.
- The results of the proposed algorithms show that it decrease the traffic congestion ratio significantly.
- To evaluate the proposed algorithms, in this paper, we use NS-3.29 and SUMO. The SUMO is used to generate realistic road scenarios.

The remainder of the paper is systematized as follows. Section II categorized the related work into two subcategories, such as traditional schemes for road traffic congestion mitigation and SD-VANET architecture. The importance of the problem statement is shown in III. We present the proposed solution in Section IV to compute the feasible path (the congestion-free path with minimum end-to-end delay) for a vehicle in the SD-VANET architecture. Section V presents the experimental setup and simulation results of the proposed algorithms. In last, Section VI shows the conclusion.

II. RELATED WORK

In this section, the related work is categorized into two categories, as follows.

A. Scheme for Congestion Mitigation and Detection

Recently one of the foremost research topics in VANETs is solving the problem of traffic congestion. The approaches proposed focused on V2V or V2I communications pose significant limitations below.

The authors proposed a scheme named TrafficView [10]. The TrafficView scheme considers the aggregated dissemination of traffic information such as broadcast time, average speed, and position of the specific vehicle on the road. This, in turn, a vehicle can learn about other on-road vehicles. However, this scheme does not consider the congestion and road traffic safety problem. In SOTIS (Self Organizing Traffic Information System) [14], the process of information exchange would be like [10], in which vehicles regularly relay data about themselves and other vehicles that they sense. In [15], the

proposed scheme applies to the prediction and simulation of traffic congestion algorithms in distributed V2V architecture. In this scheme, if the travel time of a vehicle surpasses the consistent travel time in a free-flow condition, then the route is considered as a congested route. Additionally, for experimental purposes, the authors in the proposed work, monitor each road lane for a day. In the proposed scheme, the centralized entity (server) is responsible for getting the traversal times of the vehicles. From the traversal time, the centralized entity shows the results in different VANET scenarios. The proposed work also does not compute to the traffic congestion traffic on the road.

Ahmed et al. proposed a novel scheme called IVCD (infrastructure-based Vehicle Congestion Detection) to support vehicle congestion detection and speed estimation [5]. By using iterative COC (Context-Oriented Communication) information, the proposed IVCD extracts the protection period (time-headway) between vehicles. The IVCD mechanism detects traffic congestion between vehicles over time and disseminates the identification results locally as well as globally. Other well-known schemes are [6], [16], [17] proposed for traffic congestion detection. The purposed schemes studied the relationship between different traffic parameters like traffic density/congestion and vehicle's speed and showed how these parameters could affect each other. Additionally, the relation between congestion/density and speed is the most critical parameter in the purposed schemes. In [6], the authors suggested the following linear relation of speed-density, as follows:

$$\mu = \mu_{fs} - \left(\frac{\mu_{fs}}{D_j} \right) * D, \quad (1)$$

where free-speed signified with μ_{fs} , D shows the traffic density, μ shows mean speed at D , and jam-density represented with D_j , as shown in Eq. (1). The authors in [17], proposed a logarithmic speed-density relationship, as shown in Eq. (2), where μ_m represent the speed at maximum flows.

$$\mu = \mu_m \ln \left(\frac{D_j}{D} \right) \quad (2)$$

In [7], the authors proposed a probabilistic scheme to gather traffic data based on the V2I architecture. In the proposed scheme, two new techniques are used to detect accident events in the road traffic scenario automatically. During the accident event, a vehicle interacting with the RSU on the specified channel at regular intervals, which is available on the roadside. An approximation of the instantaneous congestion scheme of the vehicle is also proposed in [8], based on data obtained by the RSU and vehicle. This scheme calculates the congestion level from the RSU messages, and RSU collects the beacon messages from both architecture, such as V2V and V2I.

In this paper, we use SD-VANET architecture, unlike traditional VANET architecture and as well as all the works mentioned above, which distributes the traffic on different road proactively to mitigate the traffic congestion.

B. SD-VANET Architecture

Recently, SDN-based architecture has been proposed for VANET to solve network problems. This section explains the related work to SD-VANET architecture as follows:

In [11], the authors proposed an SD-VANET architecture that provides scalability and flexibility in different operational modes. In case of connectivity failure with SDN controller or Base Station (BS), the proposed work installs local SDN agents in every vehicle to optimize the performance of the network. Additionally, the results show that centralized architecture (i.e., SD-VANET) performs better than the traditional distributed architecture. However, the proposed work does not discuss the about traffic congestion management and feasible path. The dynamic nature of VANET, wireless links are vulnerable because of the high mobility of vehicles, which leads to packets loss. In dynamic VANET, link stability plays a vital role in increasing the packet delivery ratio. In [12], the authors proposed a novel routing scheme in SDVN (Software-Defined Vehicular Network) to forward the packets on multiple shortest paths.

In HetNets, to enable the communication between vehicles, the authors proposed an SDVN architecture [18]. The authors also explain the challenges in the SD-VANET architecture and highlight the opportunities of this integrated architecture. The proposed work focus on the heterogeneity problem in the VANET. Additionally, the proposed SDVN minimize the frequency of status update of the vehicles by using the trajectory predictions. They use POX as an SDN controller with an NS3 simulator to validate their proposed architecture performance. In [19], the authors using the SDN controller, to collect all the information about the vehicles in the network, like vehicle speed, direction, geographical position, and neighboring RSUs' ID using 802.11p. Based on the information, the proposed scheme takes the offloading and handover decision. The proposed scheme is called OHD-SDN (Offloading with Handover Decision based on SDN).

B. Dong et al. proposed on-demand routing in the SD-VANET architecture called SDAO [20]. The proposed SDAO architecture consists of two levels, centralized local level and distributed global level. The centralized local level computes a route for every vehicle, and for global routing distributed global level is responsible for reducing the route computation overhead in VANET. In [21], the authors proposed SD-VANET architecture to support the next generation (5G) communication. The proposed architecture also provides efficient resource utilization, flexible control, and network management. They also used fog computing architecture to minimize the delay, control overhead on the SDN controller, and also maximize the throughput. In [22], [23], [24], the authors also proposed energy-efficient routing for VANET architecture using SDN and fog computing architecture to decrease the control overhead on the SDN controller, and also maximize the throughput.

The above related work shows that mostly SD-VANET focused on the minimize the SDN controller overhead, maximize the throughput, minimize the delay, and high resource utilization. However, the traffic congestion problem is missing in related work. Additionally, traffic congestion leads to the vehicle's accident, increases the trip time, and prolong the delay. Thus, in the proposed SD-VANET architecture, we try to minimize the traffic congestion and provide a feasible alternative subject to the less end-to-end delay.

III. THEORETICAL PROBLEM EXPLANATION

The VANET (i.e., ITS) attracted considerable attention because it is now a phenomenon and provisioning a variety of new services like traffic safety, avoid traffic congestion, and enhance traffic flow. The ITS provides traffic alerts, mobile cloud services, route planning, and roadside safety [1], [2], [5]. However, some critical issues in the traditional VANET architecture in urban areas are still unresolved like traffic congestion, which leads to long queuing delay, longer trip times, and decrease the vehicle's speed [6], [7], [8]. The exponential increment in connected vehicles leads to the traffic congestion problem. Traffic congestion is one of the severe problems which can paralyze the complete VANET architecture. The VANET architecture, the devices not only represent the connection among the vehicles but also include communication among infrastructure, pedestrians' collaboration, and the roads. Based on the forecast, over 300 million vehicles are emerging into the VANET market in the coming years [25]. To reducing the traffic congestion problem as mentioned earlier, it would be imperative to inform the vehicles timely to select the feasible alternative route toward destination.

To clarify the above problem statement, we consider a network, as shown in Fig. 3. In this scenario, we consider two different destination points (A and B); for type-A vehicles, the destination point is A, and for type-B vehicles, the destination is B. There are multiple paths for both destinations (A and B) in the given Fig. 1, but for the sake of simplicity, we consider four (4) paths for type-A vehicles, and type-B vehicles, we assume three (3) paths (see Fig. 3). For example, all the vehicles (type-A) select the path based on delay (i.e., path-1). This produces congestion on the path-1 and prolong the queuing delay and decrease vehicle speed. The same case with type-B vehicles, if they select the path based on delay (i.e., Path-1 or Path-2), this produces traffic congestion. The traditional VANET routing protocol provides the shortest path to the vehicles (i.e., ITS), which may lead to the congestion problem. As mentioned earlier in the problem statement that traffic congestion not only prolong the queuing delay, longer trip times, and decrease vehicle speed [5], [6] and paralyze the traffic system, but also waste the time of travelers. Therefore, it is essential to detect traffic congestion and rapid action accordingly.

To handle the traffic congestion problem in VANET, we should compute the route or divert the traffic based on the number of vehicles (vehicles' density) on route to minimize the congestion problem and minimize the delay. For this, we need a global view of the network to handle the congestion problem in VANET. Therefore, SD-VANET architecture possibly plays a vital role in reducing road traffic congestion because of the congestion problem prolong trip time, increase queuing delay, and also decrease the vehicle's speed. As mentioned earlier, the logical centralized SDN architecture decouples the control plane (controller) for data plane devices (forwarding devices) like routers, switches, APs, and RSU. Thus, SDN makes the forwarding devices (i.e., routers/switches/RSU) programmable [26]. These forwarding devices send information to the control plane, also known as the SDN controller. This, in turn, makes it easy to manage and to control the VANET network.

The SDN controller continuously collects information from the OpenFlow enabled RSUs about the entire network to

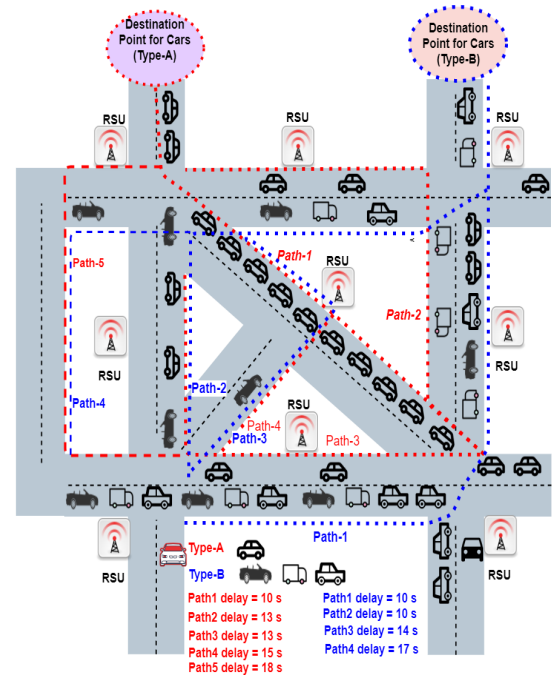


Fig. 3. Congestion scenario in VANET Architecture (shortest Path Selection).

make intelligent decisions about traffic control. In SD-VANET architecture, in wireless medium exchange beacon messages (a standard message in the VANET architecture, to learn information about the neighbor's vehicles) periodically to collect the information the network. The logical centralized controller uses this information to create a network graph. Additionally, using this information, the SDN controller can handle the congestion problem proactively distribute the vehicles according to vehicles' density on the roads and provide a feasible path and less congested path in the network. OpenFlow protocol is used to collect the information from the OpenFlow-enabled devices. When the SDN controller receives a new request for the vehicle, then search for less congestion path subject to delay parameter. After the path computing, the SDN controller informs the vehicle about the less congested path. For the sake of simplicity, consider Type-A vehicles. Four (4) paths are available toward destination-A. If the SDN controller distributes the vehicles on different paths, noticeable, it can decrease the congestion problem and minimize the queuing delay.

IV. PROBLEM FORMULATION

This section articulates the articulate the problem statement (as describe in Section III) to handle the challenging task to compute the best path (i.e., a congestion-free path and average minimum end-to-end) from the source vehicle to the chosen destination. Our proposed heuristic algorithms Congestion Free Path (CFP) and OCFP proactively disturbed the vehicles in the SD-VANET architecture to minimize traffic congestion and also decrease the average delay. For quick reference, first, we summarize the significant notations in Table I.

SDN framework primarily designed for wired networks, but now it is widely used in wireless and mobile networks [11], [18]. The SDN framework offered centralized control to optimize the resources in the wireless network like channel allocation, congestion avoidance, interference avoidance.

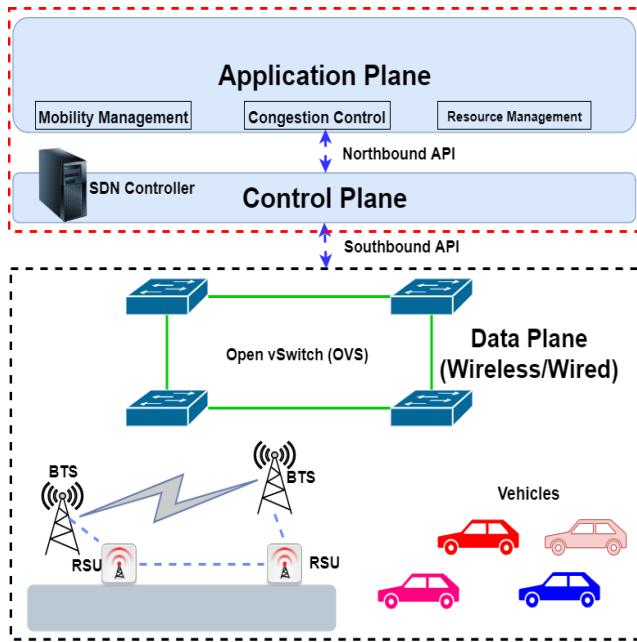


Fig. 4. SD-VANET Framework.

Packet forwarding in different scenarios (multi-path, multi-hop), efficiently handle the mobility issues, manage the heterogeneity and dynamic VANET environment. In SD-VANET, the control plane (controller) is responsible for possesses all the logical functionalities and takes actions on behalf of data plane devices, like routing, topology control, congestion avoidance, and mobility management, as shown in Fig. 4. The data plane consists of different types of devices or nodes like RSU, vehicles, OBUs (On-Board Units), and OpenFlow switches. These data plane devices execute actions specified by the flow rules which are supplied by the controller (control plane) [11], [12], [18]. Usually, for communication, the data plane devices such as OBU and RSU use wireless interfaces such as DSRC (Dedicated Short-Range Communication) and LTE.

To avoid traffic congestion in SD-VANET, the controller periodically collect the network information from OpenFlow enabled RUS [11], [12]. The controller uses this information to build a global network (G) to make intelligent decisions, such as routing, mitigation of congestion, and mobility management. The traditional VANET finds the path based on the collective functioning of devices; however, in SD-VANET, the path computation relies on the logically centralized SDN controller. The fact, the SDN controller, maintains a global network view and it simplified the congestion management in SD-VANET.

In the proposed model, the road network signified as a directed graph $G(V, R)$. In G , V denotes a set of nodes, such as points of interest or vertices, where R is the set of roads (i.e., links). In G , each road $r \in R$ associated road capacity $r_c > 0$ and a time $r_t > 0$ spent to traverse the uncongested/unloaded road. For computing the congestion level of all roads in a network, the proposed CFP and OCFP algorithms use a cost function, signified as λ_r is calculated as the traffic load on the road r_l subjected to the delay parameter r_q . Consequently, $n \in N$ is the set source and destination pairs (i.e., (s_n, d_n)), where $N \subseteq V \times V$. Each source node s_n has associated with traffic demand χ_n towards the destination node d_n . In the road network, each node $v_i \in V$ have a set

TABLE I. NOTATION DEFINITION

Notation	Definition
G	road network signified as a directed graph
V	set of nodes, such as points of interest or vertices, where $v_i \in V$
R	set of roads (i.e., links), where $r \in R$
r_c	road capacity (the maximum rate at which vehicles can travel on the road during a given time)
r_t	time spent to traverse a road (seconds (s))
r_l	the traffic load on the road (number of vehicles)
r_q	delay parameter (s)
λ_r	the cost function, calculated as the traffic load on the road r_l subjected to the delay parameter r_q
N	$n \in N$, where $N \subseteq V \times V$
(s_n, d_n)	pair of the source node and destination node
χ_n	traffic demand associated with n
$In(v_i)$	incoming roads
$Out(v_i)$	outgoing roads
$T\chi_N$	total traffic demand between source and destination pairs (i.e., (s_n, d_n))
ψ	maximum utilization of a road
r_μ	average end-to-end delay (s)

of incoming roads (i.e., links) and a set of outgoing roads, signified by $In(v_i)$ and $Out(v_i)$ respectively. The total traffic demand χ_N can be $T\chi_N = \sum_{n \in N} \chi_n$.

Objective Function

The foremost objective of our proposed algorithms is to minimize the traffic congestion problem and the average delay in VANET architecture. More precisely, to minimize the maximum utilization of the roads and provide an average end-to-end delay, as shown in Eq. (3) and Eq. (4). The objective function Eq. (3) distribute the traffic (minimize the maximum utilization) among all roads. For an average end-to-end delay, the cost function for a road $r \in R$ is $\lambda_r = r_l r_t \left[1 + \left(\frac{r_l}{r_c} \right)^{r_q} \right]$. In real-world VANETs, road delay r_q is commonly defined by nonlinear function subject to the congestion parameter. Here, we assume that the cost function λ_r is an increasing function. In Eq. (4), we normalized the average end-to-end delay r_μ by dividing $T\chi_N$. Furthermore, it shows the end-to-end delay for the traffic on each road is minimized. The constraints of Eq. (3) and Eq. (4) are explained as follows. The maximum utilization ψ of a road, as shown in constraint Eq. (5). The total traffic χ_n of all source nodes on a road $r \in R$, is shown in the constraint Eq. (6) with the decision variable ∂_r^n , where ∂_r^n is the proportion of commodity (i.e., (s_n, d_n)) $n \in N$ on a road r . The traffic conservation is shown in constraint Eq. (7), and constraint Eq. (8) defines the domain of decision variables. The objective function provides vital statistics about the traffic and distributes the traffic efficiently if $\psi < 1$ (i.e., without beyond the road capacity level $\left(\frac{r_l}{r_c} \leq \psi \right)$).

Decision Variables

- maximum road utilization = ψ
- traffic load on a road = r_l
- total traffic ((proportion of commodity), (i.e., (s_n, d_n)))
- $n \in N$ on the road $r = \partial_r^n$

$$\text{Objective1 } \min \psi \quad (3)$$

$$\text{Objective2 } \min r_\mu = \sum_{r \in R} r_l r_t \left[1 + \left(\frac{r_l}{r_c} \right)^{r_q} \right] / T_{\chi_N} \quad (4)$$

$$\frac{r_l}{r_c} \leq \psi, \quad \forall r \in R \quad (5)$$

$$r_l = \sum_{n \in N} \chi_n \partial_r^n, \quad \forall r \in R \quad (6)$$

$$\sum_{r \in In(v_i)} \partial_r^n - \sum_{r \in Ou(v_i)} \partial_r^n = \begin{cases} 1, & \text{if } v_i = d_n \\ -1, & \text{if } v_i = s_n \\ 0, & \text{otherwise} \end{cases} \quad \forall r \in R, n \in N \quad (7)$$

$$\partial_r^n \in [0, 1], r_l \geq 0, \forall r \in R, \forall n \in N, \psi \leq 1 \quad (8)$$

The SDN controller continuously collects information about the entire network to make intelligent decisions about traffic control. The proposed SD-VANET framework, in wireless medium exchange beacon messages (a standard message in the VANET framework, to lean information about the neighbor's vehicles) periodically to collect the information the network [11]. The SDN controller uses this information to create a network connectivity graph G . Additionally, using this information, the SDN controller can handle the congestion problem and provide a feasible path in the network. When the SDN controller receives a new request, then search for less congestion path subject to delay parameter, based on Eq. (3) and Eq. (4) subject to the mentioned constraints. After the path computing, the SDN controller informs the vehicle about the less congested path, as shown in Algorithm 1. When the SDN controller receives a path request from a new vehicle, if the requested path is congestion-free (i.e., $\psi \leq 0.5$), then the SDN controller returns the path (see Step 2). Otherwise, the SDN controller checks the alternative path for the vehicle (see Step 3). In Algorithm 1, the SDN controller searches all the alternative routes and return the optimum route based on Eq. (3) and Eq. (4) subject to additional constraints. However, searching for all routes increases the computational time and also NP-hard. Thus, to minimize the computational time, we proposed, the proposed Optimized heuristic CFP algorithm (OCFP) (see Algorithm 2), the controller only computes the best path among the "k" (i.e., "k" = 15) alternative routes. This is an optimization problem; therefore, to minimize the computational time, the proposed OCFP algorithm only searches for the "k" path based on Eq. (3) and Eq. (4) subject to mentioned constraints.

V. SIMULATION SETUP

This section presents the simulation and configuration setups in various scenarios to validate the proposed algorithms in SD-VANET architecture. We use NS3¹ simulator for the model architecture, and for the urban road network, we used SUMO² to generate the real vehicle mobility traffic of Dalian city, China using OpenStreetMap (OSM). OpenStreetMap shows a layout of the road network, as shown in Fig. 5. Geographically, every road network contains the number of alternative paths, but traditional VANET routing protocols are distributed in nature and provide the shortest path. The shortest path causes the congestion problem and decreases the reliability of the

Algorithm 1: CFP in SD-VANET

input : Graph of road map $G(V, R)$
output: reliable congestion free path subject to the minimum utilization ψ and average minimum delay r_μ

- 1 SDN Controller \leftarrow path_request (new_vehicle)
- 2 if $\psi \leq 0.5$:
 the requested path is congestion-free
 return path_request
- 3 else :
 (i) check all alternative congestion-free paths
 (ii) select path based on ψ and r_μ , subject to the constraints
 (iii) return path_request (update the vehicle about the alternative path)
- 4 update Graph road map status ($G(V, R)$)
- 5 end

Algorithm 2: OCFP in SD-VANET

input : Graph of road map $G(V, R)$
output: reliable congestion free path subject to the minimum utilization ψ and average minimum delay r_μ

- 1 SDN Controller \leftarrow path_request (new_vehicle)
- 2 if $\psi \leq 0.5$:
 the requested path is congestion-free
 return path_request
- 3 else :
 (i) check "k" alternative congestion-free paths (i.e., "k" = 15)
 (ii) select path based on ψ and r_μ , subject to the constraints
 (iii) return path_request (update the vehicle about the alternative path)
- 4 update Graph road map status ($G(V, R)$)
- 5 end

network like prolong delay, increases trip time. The foremost objective of the proposed heuristic algorithms is to minimize traffic congestion and also provide a feasible path to a vehicle subject to the less average delay.

Thus, our goal is to minimize the problem of traffic congestion in urban areas; we generate a road traffic scenario of Dalian city, China, one square kilometer ($1km * 1km$) in scales. The scenario for road traffic consists of six-vertical and six-horizontal roads, with junction/intersection every 250 meters. Each road is one kilometer long and two meters wide with two (2) lanes in each direction. Subsequently, at junctions, vehicles can move straight or can turn left or right. We kept constant the total number of vehicles, that is to say, 500. In vehicles, car probability is 0.80, and bus probability is 0.20, while car length is 5m, and bus length is 10m. A vehicle's maximum speed is set at 40 km/h; however, the vehicle's speed changes with time because the SUMO also simulates the traffic lights as well. The vehicles grouped in 10 traffic flows. Each flow is taking a different path/route in a total distance. At junctions, these paths intersect to simulate a high number of vehicles on the road (i.e., more than road capacity).

¹NS3 Homepage, <https://www.nsnam.org/>

²SUMO Homepage, <http://sumo.sourceforge.net/>



Fig. 5. (a) Map of Dalian city obtain from OpenStreetMap. (b) SUMO Network View to OSM of Dalian road network

This would turn in traffic congestion, which our algorithms are intended to mitigate. The total simulation time is 500s. We simulate the proposed algorithms in NS-3 and measure its performance against SDN-based Shortest Path (SDN-based SP) and an existing Distributed Road Traffic Congestion (DRTC) [6].

A. Performance Metrics

To validate the performance of the proposed algorithms, we examine the following performance parameters are considered:

- Monitor the Vehicle's Speed: We monitor the vehicle's speed by varying the congestion parameter, i.e., $\psi \leq 0.5$, $\psi \leq 0.7$, and $\psi \geq 1$.
- Congestion ratio: Congestion ratio means the density of vehicles on each path. In the simulation, a road can be considered as a congested road, if $\psi \geq 1$.
- Underutilized Roads: In the simulation, we examine the utilization ratio of roads in the network. In the simulation, a road can be considered as an underutilized road, if $\psi < 0.4$.

B. Experimental Results and Analysis

In this section, we would like to show the performance of our proposed congestion mitigation algorithms, i.e., CFP, OCFP, compared to SDN based Shortest Path (SP), and an existing Distributed Road Traffic Congestion (DRTC) [6]. In particular, we evaluate the impact number of vehicles on the congestion level. Moreover, we compare the performance in terms of vehicle's speed by varying the congestion parameter, congestion level, and underutilized roads.

To evaluate the proposed scheme (i.e., CFP), we exam the vehicle's speed over different congestion levels, i.e., $\psi \leq 0.5$, $\psi \leq 0.7$, and $\psi \geq 1$. In the simulation, the maximum speed of a vehicle is 40km/h (11.11 m/s), but the actual speed is based on the traffic conditions like congestion and traffic lights. Therefore, we evaluate the vehicle speed by varying traffic congestion level. Consequently, we selected one vehicle and plotted its speed under different congestion levels, as shown in Fig. 6, Fig. 7, and Fig. 8. From these results, we exam

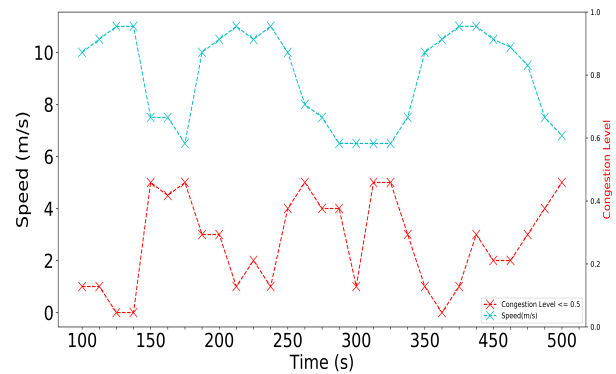


Fig. 6. Vehicle's Speed vs. Congestion Level ($\psi \leq 0.5$).

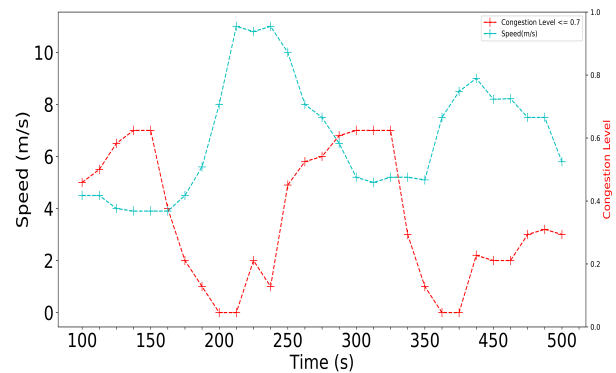


Fig. 7. Vehicle's Speed vs. Congestion Level ($\psi \leq 0.7$).

the congestion level significantly affect the vehicle's speed. In Fig. 6, the congestion level is $\psi \leq 0.5$; therefore, the vehicle's speed is affected less compared to Fig. 7 and Fig. 8. When the congestion level reaches the maximum level (i.e., $\psi \geq 1$), then it affects the vehicle's speed dramatically, as shown in Fig. 8.

The result shows that the congestion level increases in all schemes when the number of vehicles increases, as shown in Fig. 9. The result shows that in our proposed approaches (i.e., CFP and OCFP), the traffic congestion level is very less compared to SDN-based SP and DRTC approaches. The congestion

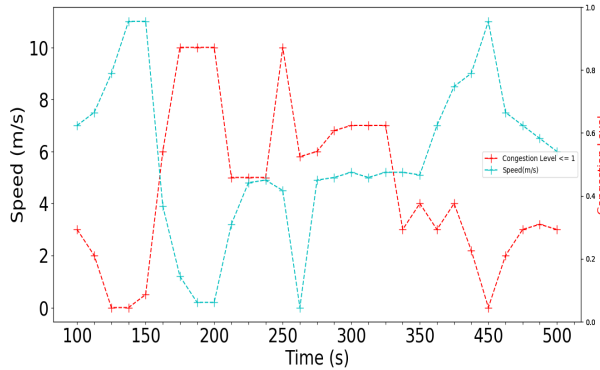


Fig. 8. Vehicle's Speed vs. Congestion Level ($\psi \leq 1$).

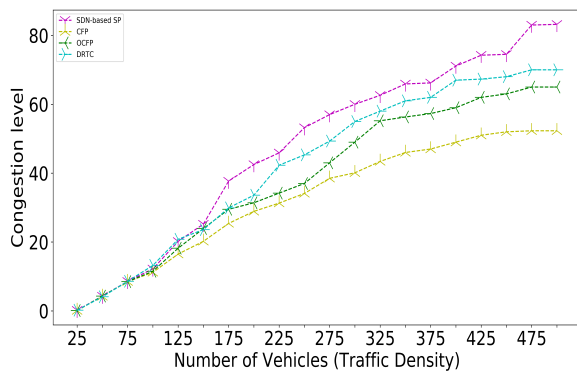


Fig. 9. Congestion Level Vs. Traffic Density.

level is almost the same in all approaches when the traffic density is between 25 to 125. However, when the traffic density increases, the proposed approaches perform better. The main reason is that the proposed approaches distribute the vehicles proactively on the alternative paths according to the vehicles' density on the road. The congestion level is very less in CFP approach because of this scheme's search for all the alternative paths for a vehicle. Additionally, the congestion level in OCFP is more than CFP because this scheme only searches for the "k" alternative path based on the objective function, unlike CFP. Subsequently, SDN-based SP finds the shortest path and does not consider the vehicles' density on the road, which leads to traffic congestion. Furthermore, the congestion level in DRTC is more compared to proposed algorithms because DRTC is an infrastructure-less distributed V2V scheme for congestion detection. The DRTC scheme enables each vehicle to detect the traffic congestion condition and then share the congestion information with vehicles through cooperation. The cooperation, however, decreases the number of broadcasting vehicles but lack of a centralized controller, the congestion level is more than the proposed algorithms. Additionally, the DRTC scheme does not distribute the vehicles proactively on different paths. The congestion level in CFP is almost 53%, 65% in OCFP, in DRTC, the congestion level is almost 70%, and in SDN-based SP, the congestion level is more than 80% when the traffic's density reached to 500.

Fig. 10 shows the result of the underutilized ratio of roads in VANET architecture. The underutilized ratio of roads decreases in all schemes when the number of vehicles increases.

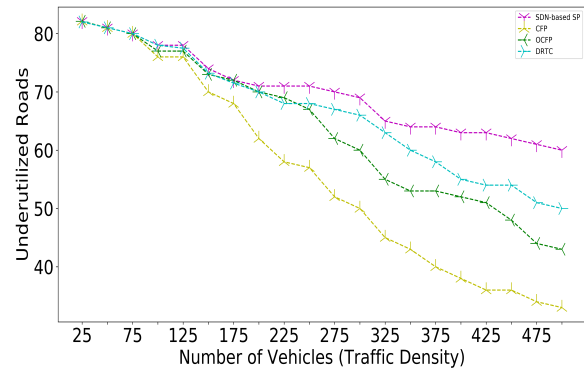


Fig. 10. Underutilized Roads Vs. Traffic Density.

However, the result shows that in our proposed approaches (i.e., CFP and OCFP), distribute the vehicles proactively on all paths and utilize the more roads compared to SDN-based SP and DRTC approaches. Therefore, the underutilized ratio of roads less in the proposed approach. The utilization ratio of roads in all approaches is almost the same when the traffic density is between 25 to 125. However, when the traffic density increases, the proposed approaches perform impressively because the proposed approaches distribute the vehicles based on traffic density on the roads. The underutilized ratio is very less in CFP approach because of this scheme's search for all the alternative paths for a vehicle, as explained in Algorithm 1. Additionally, the underutilized ratio in OCFP is more than CFP because this scheme only searches for the "k" alternative path based on the objective function, unlike CFP. The SDN-based SP and DRTC schemes do not distribute the vehicles on all roads; therefore, the underutilized ratio is more than the proposed approaches. The underutilized ratio in CFP is almost 32%, 43% in the OCFP scheme, in DRTC approach, the congestion level is almost 50%, and in SDN-based SP, the congestion level is more than 60% when the traffic's density reached to 500.

The results of all scenarios indicate that our proposed algorithms, CFP and OCFP, outperform compared to DRTC. The primary difference between our proposed algorithms and DRTC is that our algorithms use centralized SDN architecture, and DRTC uses distributed architecture. Secondly, DRTC uses Eq. (1) to compute the congestion level; however, our algorithm distributed the vehicles proactively on alternative paths. Thus, this is the main reason that the proposed algorithms have global information about the network and provide better services. The proposed algorithms decrease the congestion problem in VANET.

VI. CONCLUSION

The proposed algorithms try to provide the congestion-free path subject to the average minimum delay paths to the vehicles in the SD-VANET network. In VANET, the congestion problem leads to longer trip times, decreases the vehicle's speed, and prolong delay. The SDN controller in VANET collects the information about the vehicles on each road to calculate the traffic density on the road. If the road is going to dense, the SDN controller uses global network information and divert the traffic on another feasible path subject to minimum delay. The results indicate that the proposed algorithms

(i.e., CFP and OCFP) in SD-VANET architecture effectively switches the vehicle on an un-congested path subject to minimum delay and minimize the congestion characteristics. Our proposed approach is a heuristic approach and has linear time complexity. For future work, we will use some machine learning approaches to minimize the time complexity and provide better performance.

REFERENCES

- [1] A. Wahid, A. Rao, and D. Goel, "Server communication reduction for gps-based floating car data traffic congestion detection method," in *Integrated Intelligent Computing, Communication and Security*. Springer, 2019, pp. 415–425.
- [2] Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Dependable content distribution in d2d-based cooperative vehicular networks: A big data-integrated coalition game approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 953–964, 2018.
- [3] X. Ke, L. Shi, W. Guo, and D. Chen, "Multi-dimensional traffic congestion detection based on fusion of visual features and convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2157–2170, 2018.
- [4] C. Jayapal and S. S. Roy, "Road traffic congestion management using vanet," in *2016 International Conference on Advances in Human Machine Interaction (HMI)*. IEEE, 2016, pp. 1–7.
- [5] M. Ahmad, Q. Chen, Z. Khan, M. Ahmad, and F. Khurshid, "Infrastructure-based vehicular congestion detection scheme for v2i," *International Journal of Communication Systems*, vol. 32, no. 3, p. e3877, 2019.
- [6] M. Milojevic and V. Rakocevic, "Distributed road traffic congestion quantification using cooperative vanets," in *2014 13th annual Mediterranean Ad Hoc networking workshop (MED-HOC-NET)*. IEEE, 2014, pp. 203–210.
- [7] O. Popescu, S. Sha-Mohammad, H. Abdel-Wahab, D. C. Popescu, and S. El-Tawab, "Automatic incident detection in intelligent transportation systems using aggregation of traffic parameters collected through v2i communications," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 2, pp. 64–75, 2017.
- [8] J. Barrachina, P. Garrido, M. Fogue, F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, "A v2i-based real-time traffic density estimation system in urban scenarios," *Wireless Personal Communications*, vol. 83, no. 1, pp. 259–280, 2015.
- [9] M. Dixit, R. Kumar, and A. K. Sagar, "Vanet: Architectures, research issues, routing protocols, and its applications," in *2016 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2016, pp. 555–561.
- [10] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "Trafficview: A scalable traffic monitoring system," in *IEEE International Conference on Mobile Data Management, 2004. Proceedings. 2004*. IEEE, 2004, pp. 13–26.
- [11] I. Ku, Y. Lu, M. Gerla, R. L. Gomes, F. Ongaro, and E. Cerqueira, "Towards software-defined vanet: Architecture and services," in *2014 13th annual Mediterranean ad hoc networking workshop (MED-HOC-NET)*. IEEE, 2014, pp. 103–110.
- [12] K. L. K. Sudheera, M. Ma, and P. H. J. Chong, "Link stability based optimized routing framework for software defined vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2934–2945, 2019.
- [13] L. Zhao, W. Zhao, A. Al-Dubai, and G. Min, "A novel adaptive routing and switching scheme for software-defined vehicular networks," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.
- [14] L. Wischhof, A. Ebner, and H. Rohling, "Information dissemination in self-organizing intervehicle networks," *IEEE Transactions on intelligent transportation systems*, vol. 6, no. 1, pp. 90–101, 2005.
- [15] G. Marfia and M. Roccetti, "Vehicular congestion detection and short-term forecasting: a new model with results," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 2936–2948, 2011.
- [16] M. Kimura, Y. Taoda, Y. Kakuda, S. Inoue, and T. Dohi, "A novel method based on vanet for alleviating traffic congestion in urban transportations," in *2013 IEEE Eleventh International Symposium on Autonomous Decentralized Systems (ISADS)*. IEEE, 2013, pp. 1–7.
- [17] Z. He, J. Cao, and T. Li, "Mice: A real-time traffic estimation based vehicular path planning solution using vanets," in *2012 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, 2012, pp. 172–178.
- [18] Z. He, J. Cao, and X. Liu, "Sdvn: Enabling rapid network innovation for heterogeneous vehicular communication," *IEEE network*, vol. 30, no. 4, pp. 10–15, 2016.
- [19] C.-M. Huang, M.-S. Chiang, D.-T. Dao, H.-M. Pai, S. Xu, and H. Zhou, "Vehicle-to-infrastructure (v2i) offloading from cellular network to 802.11 p wi-fi network based on the software-defined network (sdn) architecture," *Vehicular Communications*, vol. 9, pp. 288–300, 2017.
- [20] B. Dong, W. Wu, Z. Yang, and J. Li, "Software defined networking based on-demand routing protocol in vehicle ad hoc networks," in *2016 12th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*. IEEE, 2016, pp. 207–213.
- [21] A. A. Khan, M. Abolhasan, and W. Ni, "5g next generation vanets using sdn and fog computing framework," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2018, pp. 1–6.
- [22] A. J. Kadhim and S. A. H. Seno, "Energy-efficient multicast routing protocol based on sdn and fog computing for vehicular networks," *Ad Hoc Networks*, vol. 84, pp. 68–81, 2019.
- [23] J. C. Nobre, A. M. de Souza, D. Rosário, C. Both, L. A. Villas, E. Cerqueira, T. Braun, and M. Gerla, "Vehicular software-defined networking and fog computing: Integration and design principles," *Ad Hoc Networks*, vol. 82, pp. 172–181, 2019.
- [24] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular adhoc network with fog computing," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 1202–1207.
- [25] K. Liu, X. Xu, M. Chen, B. Liu, L. Wu, and V. C. Lee, "A hierarchical architecture for the future internet of vehicles," *IEEE Communications Magazine*, vol. 57, no. 7, pp. 41–47, 2019.
- [26] J. Bhatia, R. Dave, H. Bhayani, S. Tanwar, and A. Nayyar, "Sdn-based real-time urban traffic analysis in vanet environment," *Computer Communications*, vol. 149, pp. 162–175, 2020.