

Vehicular Software-Defined Networking and Fog Computing: Integration and Design Principles

Jéferson Nobre^{a,*}, Allan M. de Souza^{b,e}, Denis Rosário^c, Cristiano Both^d,
Leandro A. Villas^b, Eduardo Cerqueira^c, Torsten Braun^e, Mario Gerla^f

^aUniversity of Vale do Rio dos Sinos, São Leopoldo – Brazil

^bInstitute of Computing, University of Campinas (UNICAMP), Campinas – Brazil

^cFederal University of Pará, Belém – Brazil

^dFederal University of Health Sciences of Porto Alegre, Porto Alegre – Brazil

^eInstitute of Computer Science, University of Bern, Bern – Switzerland

^fUniversity of California Los Angeles (UCLA), Los Angeles – USA

Abstract

Vehicular Networks (VN) enable the collaboration among vehicles and infrastructure to deliver network services, where usually value-added services are provided by cloud computing. In this context, fog computing can be deployed closer to the users to meet their needs with minimum help from the Internet infrastructure. Software Defined Networking (SDN) might support the use of large-scale fog-enabled VN services. However, the current management of each wireless network that composes the VN has restricted the exploration of fog-enabled VN services. Therefore, the design principles for a VN architecture is still an open issue, mainly because it is necessary to address the diversity of VN fog applications. In this article, we investigate the design principles for fog-enabled Vehicular Software Defined Networking (VSDN) focusing on the perspectives of the systems, networking, and services. We evaluated these design principles in a use case of a traffic management system for a fast traffic accident rescue, using real traffic accident data. Finally, potential research challenges and opportunities for integrated use fog-enabled VSDN are discussed.

*Corresponding author

Email addresses: jcnobre@unisinos.br (Jéferson Nobre), allan@lrc.ic.unicamp.br (Allan M. de Souza), denis@ufpa.br (Denis Rosário), cbboth@ufcspa.edu.br (Cristiano Both), leandro@ic.unicamp.br (Leandro A. Villas), cerqueira@ufpa.br (Eduardo Cerqueira), braun@inf.unibe.ch (Torsten Braun), gerla@cs.ucla.edu (Mario Gerla)

Keywords:

Vehicular Network, Fog computing, Software Defined Networking

1. Introduction

Vehicular Networks (VN) enable the collaboration among vehicles and infrastructure to deliver value-added services, ranging from accident reduction and route recommendation to entertainment [1]. Several research efforts have
5 been devoted to investigating different VN characteristics, such as traffic management, road safety, social patterns, computing and networking demands [2]. In this context, VNs are about to evolve with emerging paradigms since vehicles directly seek contents regardless of their providers.

Fog computing provides cloud systems deployed closer to the users to meet
10 their needs regarding processing and delay with minimum help from the Internet infrastructure [3]. In this context, a vehicle could support a fog node that downloads the global traffic conditions from the cloud, and uploads local traffic conditions using the network infrastructure (vehicle-to-infrastructure - V2I and vehicle-to-vehicle - V2V) [1]. Fog nodes might be placed at different levels,
15 ranging from dedicated servers in the radio access network, or core network to the vehicles [4, 5]. For instance, vehicles will produce and consume plenty of data having the property of local relevance (either regarding time or space) [6]. However, the cooperation between fog and cloud computing for VN services must be seamless to bring significant benefits for both users and network/content
20 providers, considering heterogeneous VNs with different access technologies.

The fog/cloud integration improves user experiences without increasing the burden on V2V communications [7]. For example, a fog node enables better traffic light control to decrease the waiting time at intersections, and also rapid accident rescue to improve emergency responsiveness. Despite the recent intro-
25 duction of new flexible and efficient ways to configure and manage the network such as Software Defined Networking (SDN) [8], the current ossified networking infrastructure and the individual management of each wireless network that

composes VN has still restricted the exploration of fog infrastructures for scalable VN services [9]. Therefore, the design principles for a VN architecture is
30 still an open issue, mainly due to the diversity of VN fog applications.

In this article, we investigate a VN architecture based on SDN, called VSDN. We explore the design principles of SDN architecture, building on top of vehicular fog computing. In the VSDN architecture, the cloud orchestrates/controls the fog nodes in a centralized fashion. Moreover, the design principles for a
35 VSDN architecture should enable content dissemination to efficiently accommodate a large number of vehicular users with any kinds of communication technologies and devices. The main contributions of this article are: (i) design principles for a flexible VSDN architecture focused on the perspectives of the system, networking, and services considering SDN to improve the deployment
40 of fog nodes; (ii) a use case about a traffic management system for a fast traffic accident rescue for emergency vehicles, using real traffic accident-related data in the city of São Paulo, Brazil; (iii) an analysis of how the integration of VSDN and fog computing can minimize the arrival time of emergency vehicles at the accident location; and (iv) identification of potential research challenges and
45 opportunities for integrated use of SDN and fog computing in VN environments composed of different wireless technologies.

The remainder of this article is organized as follows. Section 2 presents a concise background on architectures for fog-enabled VSDN. Section 3 introduces the design principles for integrating VSDN and fog computing. Section 4 shows
50 a fast traffic accident rescue system for emergency vehicles. Section 5 discusses key research challenges. Finally, Section 6 presents the concluding remarks.

2. Reference Architecture for Fog-enabled VSDN

IP Wireless Access in Vehicular Environments (IPWAVE)¹ from the Internet Engineering Task Force (IETF) and also the 5GINFIRE project² introduce

¹<https://datatracker.ietf.org/doc/draft-ietf-ipwave-vehicular-networking/>

²<https://5ginfire.eu/it-av-automotive-testbed/>

55 fundamental architectural concepts on a real VN scenario. The literature also presents some reference architectures on fog computing [5, 10, 11, 12]. Furthermore, several reference architectures were proposed for VSDN [13, 14, 15]. However, the state-of-the-art of reference architectures that efficiently provide the integration between fog and VSDN is in an initial stage.

60 VSDN architectures consider SDN as a flexible approach to control and orchestrate the VN [14]. This is because SDN provides flexibility and programmability by separating the network infrastructure into planes together with the centralization of control functions, where each plane can be programmed to meet particular application requirements [16]. The SDN controller is responsible for
65 all control and management functions in a centralized way [17].

In the VSDN architecture, cloud computing might be responsible for the control functions of the VN nodes in a centralized fashion. It is convenient to implement SDN on cloud-based network applications because of its natural character of a centralized control mechanism [5]. For instance, such VSDN
70 architecture must consider a broad geographical distribution of fog nodes with different characteristics regarding processing, storage, and reachability. Additionally, the VSDN architecture must manage and orchestrate the V2V or V2I communications, which are supported by numerous wireless technologies, such as IEEE 802.11p, LTE, and others [18].

75 Several design challenges are imposed for a fog-enabled VSDN architecture to provide fog services for VN applications. The SDN controller must be responsible for the management of geographically distributed fog nodes. The controller must manage/orchestrate the communication among different wireless access technologies for V2V and V2I communication. Furthermore, fog nodes have
80 local knowledge about the VN and could provide service for the vehicles inside their range. In this sense, the VSDN architecture must give efficient management to deploy, migrate, and orchestrate the VN services in such fog nodes. A VN is composed of moving vehicles, and thus the VSDN architecture must deal with vehicle mobility by migrating fog services.

3. Design Principles

In this section, we discuss several challenges imposed for the design principles for future development of fog-enabled VSDN considering three perspectives: System, Networking, and Service. Figure 1 depicts a VN together with these perspectives, *i.e.*, system, networking, and services, for fog-enabled VSDN. From a networking perspective, the VN is composed of different wireless technologies to provide V2V and V2I communication and address intermittent connectivity caused by vehicle mobility. From the system perspective, fog nodes have separate storage and processing characteristics and are widely distributed across the environment. In this sense, fog computing aims to move processing and storage resources to the network edge, and the VSDN architecture should support this movement flexibly and efficiently. From the service perspective, the fog nodes offer many services for the VN, which require high-level forms of guidance, information, and commands (*i.e.*, policies) to meet specific applications or user requirements. Furthermore, these services have some security requirements.

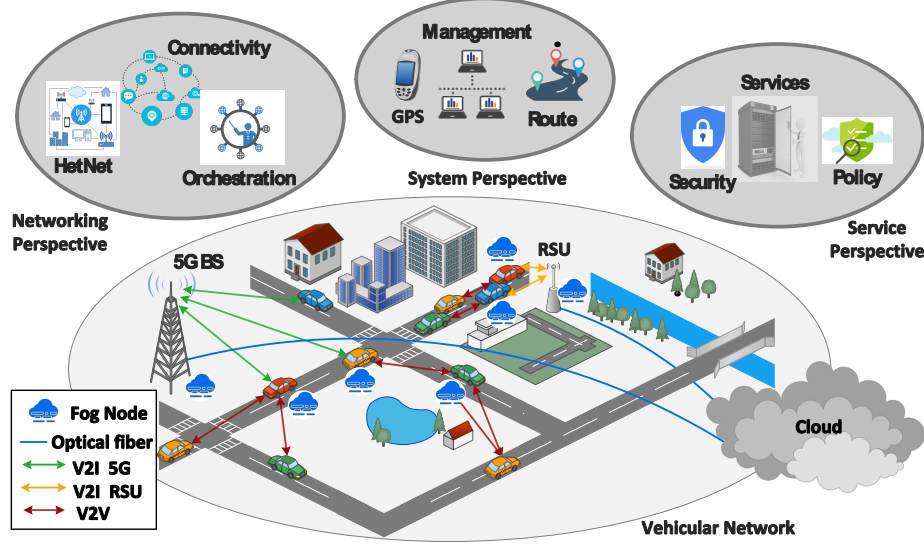


Figure 1: VN Scenario Considering System, Networking, and Service Perspectives

100 3.1. System Perspective

The design principles considering the system perspective can be highlighted in the context of (i) fog node localization, (ii) fog node characteristics, and (iii) fog management. These principles are presented in the following.

Fog Node Localization: Fog-enabled VSDN consists of fog nodes geo-
105 graphically distributed in the VN coverage area ranging from dedicated servers in the radio access or core network to the vehicles. These fog nodes provide computation, networking, and storage resources organized hierarchically between the cloud on the top and the vehicle at the bottom [19]. Each fog node offers services to match the topology and distributed workload properties of the VN
110 applications. In this sense, the scope of the gathered and processed data is restricted to the location of such nodes [20]. For instance, in a vehicular Traffic Management System (TMS) [21], a fog node might be responsible for a local traffic view, which stores an updated knowledge about traffic conditions of a particular area in the city.

Fog Node Characteristics: In a TMS application, each fog node might
115 receive route planning requests and recommend alternative routes for vehicles based on the current position and destination within its knowledge area (*i.e.*, local traffic view). Fog nodes might also collect and process video data from a given accident, which helps to understand how critical is the crash. These
120 characteristics should be extensible to handle the various vehicle content generated from dynamically changing road environments. Moreover, in some cases, each vehicle might play the content provider and consumer roles simultaneously, since the vehicles can also be fog nodes. Therefore, it is necessary to introduce a discovery process, where vehicles actively solicit specific services. The VSDN
125 architecture must help the discovery process through enabling selective and intelligent requests.

Fog management: This context can be conducted considering the fundamental characteristics of an SDN management plane [22]. Therefore, the VSDN controller and orchestrator should manage V2I communication, *i.e.*, not
130 only from Base Stations (BSs) but also from Road Side Unit (RSUs). In this

way, the VSDN controller can build a global connectivity graph of VSDN nodes and other necessary knowledge for fog computing services. Besides that, it is desirable to manage services distributed in different fog nodes, since they process some services without global knowledge as in a central VSDN controller or
135 orchestrator.

3.2. Networking Perspective

The design principles considering the network perspective can be highlighted in the context of (i) fog orchestration, (ii) heterogeneity management, and (iii) intermittent connectivity. These principles are presented in the following.

140 **Fog Orchestration:** The software component is required to tailor a VSDN architecture to assure the benefits of integrating SDN characteristics and fog computing into a VN. For example, the RSUs and BSs can be coordinated by a fog orchestrator based on the MANO architecture [12]. In this way, BSs and RSUs need an orchestration mechanism to disseminate data-forwarding rules
145 and information on services hosting [23]. The fog orchestrator must consider different details, such as Quality of Service (QoS), Quality of Experience (QoE), topology, network technology, operator, etc., to decide where and when to deploy fog nodes that run VN services on fog nodes. Finally, the fog orchestrator can either incorporate the VSDN controller or share a control plane with it.

150 **Heterogeneity Management:** Modern VNs heavily rely on heterogeneous wireless access technologies, such as IEEE 802.11p, LTE, and others [24]. In a VSDN architecture, the control plane can integrate the network and routing protocols with the requirements of the application [25]. In the fog computing context, the VSDN control plane must also control the fog node communication.
155 Besides that, the VSDN orchestrator must deal with vehicles moving between different locations and networks in heterogeneous VN scenarios.

Intermittent Connectivity: Vehicle mobility causes intermittent connectivity, which can threaten the availability of fog resources and cause failures to collaborate. In this context, some creative VN applications have strict time
160 constraints, such as data for emergency systems. The controller must main-

tain the up-to-date network topology of each vehicle, which can be achieved by periodically broadcasting beacon messages along with geographical data of the vehicle, such as route map, position, and speed [23]. Since fog nodes are distributed across the city to ensure adequate coverage of the region, they can also
165 correctly manage dynamically changing connectivity to minimize the impact on the provisioning service.

3.3. Service Perspective

The design principles for the service perspective can be highlighted in the context of (i) fog services, (ii) policy, and (iii) security mechanism. These
170 principles are presented in the following.

Fog Services: A set of pre-programmed services might be available by fog nodes, such as assisted driving, autonomous driving, collision avoidance, accident detection, emergency messages dissemination, TMSs, remote video analysis, and other VN services. For instance, TMSs have strict requirements re-
175 garding low latency communications and real-time responsiveness to perform freeway-traffic-flow management, individualized vehicle path planning, vehicle localization, and other services [21]. In this sense, a fog node can be accountable for having information about each local traffic view as well as to be responsible for managing the traffic mobility within its area of knowledge. In other words,
180 each fog node is responsible for receiving route planning requests and recommending alternative routes for vehicles with current and destination positions within its area of knowledge. Cloud might also be responsible for storing the global traffic view and for managing the traffic mobility beyond the fog node's knowledge.

Policy: The services provided in a fog-enabled VSDN must achieve exten-
185 sibility and flexibility on the city level. Several levels of policy abstraction are required. The VSDN control plane communication is responsible for disseminating policy rules. Therefore, the VSDN controller and orchestrator will send general policy rules in which specific behavior will be decided by fog nodes
190 depending on their local context. In this context, intent-based languages, *e.g.*,

Network MOdeling (NEMO) language³ would be the best choice as they present high-level abstractions. In any case, intent-based languages have to be translated to lower level policies. These policies should have the power to define the content processing by fog nodes as well as the operation of the VN. For example, the control plane could be steered considering fog applications. Therefore, if required by such application, the process of control channels (either dedicated or common/broadcast) can be modeled to support specific QoS/QoE.

Security Mechanisms: An essential aspect of a VN is security mechanisms. For example, only valid vehicles should be allowed to participate in such networks. Thus, some form of authentication process (*e.g.*, in the way of an authentication server in the traffic control center) must be in place, using vehicle identifiers and certificates to authenticate vehicle/user pairs. Privacy is of paramount importance since the information about and produced by each vehicle now traverses not only the cloud but also fog nodes. Monitoring such information can reveal drivers' patterns, regarding mobility and applications use. Therefore, some form of anonymization must be employed, *e.g.*, using temporary and randomized identifiers schemes. Besides the cloud platform itself, the most critical availability issue would be threats targeting fog nodes. In this context, VSDN can improve vehicular communication security properties since the elements that implement the control plane (*i.e.*, controller and orchestrator) can be used to host security mechanisms. Finally, such mechanisms can directly fit the network using VSDN features. For example, an anti-DDoS system could interact with the controller to redirect malicious traffic.

4. Use Case: Fast Traffic Accident Rescue

In this section, we present a study about a traffic management system for a fast traffic accident rescue for emergency vehicles. First, we show the benefits of VSDN and fog integration for such goal. Next, we address a full scenario and

³Network MOdeling (NEMO) language - <http://nemo-project.net/>

traffic accident dataset description. Furthermore, we discuss the performance analysis of the about the TMS for a fast traffic accident rescue for emergency
220 vehicles.

4.1. The Fog-enabled VSDN

Traffic accidents are notably grave concerns for big cities around the world since their negative consequences range from physical damages for the injured to death. According to the commission for global road safety, traffic accidents
225 kill at least 1.3 million people each year and injure more than 50 million [26]. The so-called *golden hour* philosophy indicates that accident victims have much poorer survival rates if they are not delivered to the hospital and definitive care within one hour, including the time taken for call-out, traveling to the accident location, the rescue of the injured and transport to the hospital [27]. However, in
230 an urban environment, bottlenecks in transportation infrastructure and traffic congestion can delay such relief. In practice, when a traffic accident blocks the traffic flow in an metropolitan area, the emergency vehicles will not be able to get at the emergency location suitably, due to the lack of prioritized emergency traffic and rescue lanes. In summary, traffic accidents cause traffic congestion
235 and delay the emergency arrival, increasing the rescue time, while reducing the survivability chances of injured people.

Fog-enabled VSDN provides a potential opportunity to deal with traffic accident rescue properly. The low-latency granted by the VSDN together with the fast responsiveness of fog nodes can improve the rescue approach by performing
240 the following tasks:

- **Remote video analysis and initial assessment:** The traditional rescue approach sends an emergency vehicle whenever an aid request is received without any first assessment of the accident scene. The emergency vehicles try to arrive at the accident point within the shortest time. However, traditional rescue approaches do not consider any analysis and advice to understand how critical is the accident, which does not avoid more
245

damages and collisions, and also for removing the injured persons from the vehicle properly. It is reasonable to utilize sensors available in the vehicle to gather accident-related information and forward it to the nearest rescue center. Furthermore, a real-time video can be retrieved and disseminate to the rescue center, which can have accurate first-hand information to evaluate the accident based on visual information. In this way, it is possible to prepare an efficient rescue, and also notifying suitable doctors and devices to further treatment.

- **Accident notification and congestion avoidance:** Since nearby vehicles do not know about the traffic accident, the traffic congestion may arise at the accident location. The increase of congestion is because the vehicles will enter into a blocked road without knowing it, and consequently get stuck in the traffic jam. These potentially blocks the access of the emergency vehicles to the accident location. However, a fog node close to the accident location can define a congestion avoidance area, and disseminate it to warn vehicles going towards the accident location. Once a vehicle enters into a congestion avoidance area, it is notified to avoid the accident spot to prevent traffic congestion and delaying the rescue activity.
- **Fast rescue route and emergency traffic prioritization:** A policy to prioritize emergency traffic can be defined to avoid delay in the rescue activity caused by traffic congestion. In this sense, a rescue route is set, and the vehicles monitoring such policy will pave the way for the emergency vehicles. Moreover, each fog node that covers the rescue route can alert vehicles about the rescue activity, reducing its time.

To analyze the performance of the tasks described above, we define a realistic scenario for a fast traffic accident rescue. The metropolitan scenario was designed by integrating two simulation environments, as it can be shown in the following.

275 4.2. Accident Rescue Scenario

We consider the OMNeT++ 4.3 [28] network simulator combined with the Simulation of Urban Mobility (SUMO) [29] version 0.25 as the simulation platform to evaluate the fast traffic accident rescue. Specifically, SUMO manages the mobility of the vehicle, while the framework Veins 4.3 [30] implements the features of a VN, such as IEEE 802.11p and the signal attenuation model that
280 considers the effects of obstacles, *e.g.*, buildings in urban areas.

To obtain a realistic simulation scenario, we used a fragment of $\approx 25 \text{ km}^2$ of São Paulo city, Brazil, exported from OpenStreetMap (OSM) [31]. The traffic conditions impact the rescue time, and thus we define two different traffic mobility models using the TrafficModeler tool. The first one relies on a typical
285 traffic situation, where the travel time of the majority of vehicles is close to the free-flow travel time. The second one relies on a rush-hour case, *i.e.*, heavy traffic congestion, and, thus, the travel time of the majority of vehicles is significantly higher than the free-flow travel time. In this way, we generate the traffic
290 at a constant rate by deploying one vehicle each second in the simulation with a route path from one side to another side of the simulation scenario. By default, the shortest travel time paths are automatically calculated and assigned to each vehicle at the beginning of the simulation based on the road speed limits.

We deployed a set of nine RSUs in a homogeneous fashion to cover the whole area of the scenario. We define each RSU's position based on its communication
295 range. These RSUs work as fog nodes, and thus each fog node is responsible for managing vehicles within its coverage area based on services available in that area. We also define the principal server and an SDN controller, which can communicate with all fog nodes. Table 1 shows additional parameters used in the simulation. Regarding the traffic accidents, we used a real accident report
300 provided by the traffic engineering company of São Paulo ⁴. This report contains the location of each traffic accident that occurred in the São Paulo city during one year. Therefore, we randomly selected a set of 100 accidents from all traffic

⁴<http://www.cetsp.com.br/sobre-a-cet/relatorios-corporativos.aspx>

accidents within the scenario area in the year 2016.

Table 1: Simulation parameters

Parameters	Values
OSM bound box	$-46.6677, -23.5740; -46.6009, -23.5220$
Normal traffic density	≈ 250 vehicles/km ²
Rush-hour traffic density	≈ 1000 vehicles/km ²
Channel frequency	5.890e9 Hz
Propagation model	<i>Two ray</i>
Vehicle communication range	300 m
RSU communication range	1000 m
Antenna model	Omnidirecional
Bit rate	18 Mbit/s
PHY model	IEEE 802.11p
MAC model	EDCA

305 After the simulation gets stable (*i.e.*, after a warm-up period defined as 500 seconds), an accident is induced at the location previously obtained using real traffic accident-related data in the city of S  o Paulo, Brazil. Afterwards, an emergency vehicle is added to the simulation to provide the rescue. Naturally, the rescue route is defined based on the accident location and the nearest emergency center, which can be obtained from the information available in the OSM exported file.

For the sake of clarity, Figure 2 depicts the scenario description. As it can be seen, while an accident occurs, the RSU receives a video from the crash and sends it to the nearest rescue center. Meanwhile, a virtual congestion avoidance area 315 is defined (represented by the blue square) based on the accident’s location to warn vehicles within it about the crash and also the ones who are going towards such area. Finally, the rescue route is defined (represented by the green line starting on the hospital and ending at the accident location), and the vehicles that are respecting the policy of emergency rescue are alerted to keep the route 320 clean for the emergency vehicles.

4.3. Performance Analysis

We analyze two scenarios for evaluating how VSDN and fog integration can improve the rescue activity. We compare the performance of a Fast Traffic Acci-

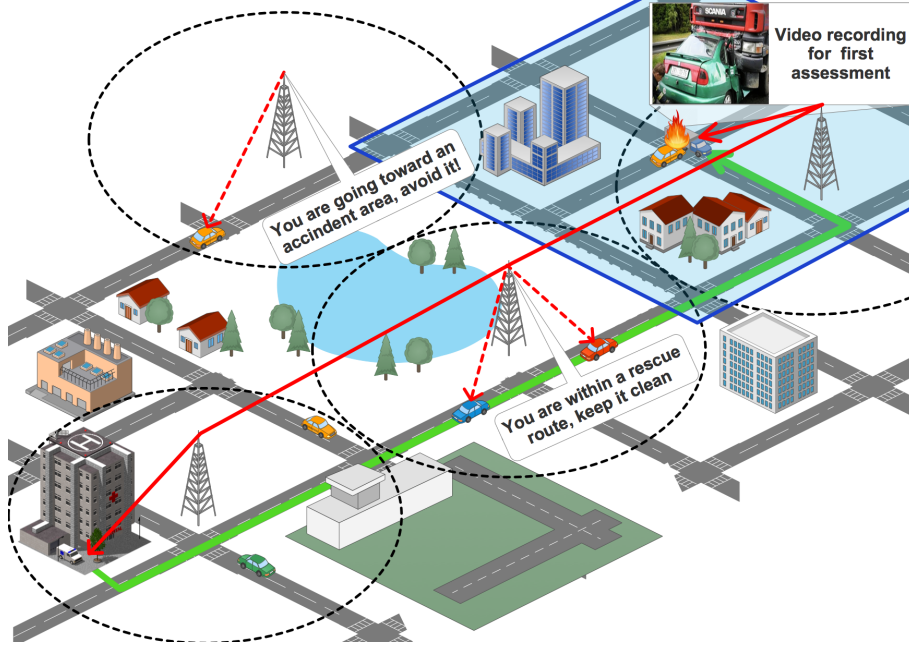


Figure 2: Fast Traffic Accident Rescue Scenario for Emergency Vehicle Considering VSDN and Fog computing

325 dent Rescue (FTAR) approach with a conventional Traffic Management System (TMS). Specifically, the FTAR approach employs the mechanisms for a better rescue activity, including a first analysis based on video data, accident alerting, congestion avoidance, and emergency traffic prioritization. On the other hand, TMS [32, 33, 20, 34, 35] is a literature-based approach, which employs dynamic routing planning mechanisms for improving the overall traffic mobility, which is
 330 based on a routing interval, *e.g.*, a predefined period which all vehicles are re-routed. TMS computes the fastest route for each vehicle considering the current traffic conditions on roads. For the sake of clarity, the TMS used for comparison in this paper is described in [33], which shows substantial improvements compared to literature solutions introduced in [36] and are based on cellular in-
 335 frastructures and smartphone applications for dealing with the traffic efficiency problem. In this context, we consider the following metrics to analyze using both traffic situations, *i.e.*, normal and hush traffic situations:

- **Relative rescue time:** It represents the ratio between the travel time of the emergency vehicles using either FTAR or TMS approaches and the travel time of the traditional baseline. This baseline means without any mechanism for improving the rescue time or minimizing traffic congestion within the rescue route. This metric summarizes the reduction in the time taken by the emergency vehicle between departing from its origin and arriving at the accident location.
- **Relative time loss:** It means the ratio between the time loss of the emergency vehicle route using either FTAR or TMS approaches and the time loss of the emergency vehicles using the traditional baseline. This metric summarizes the reduction of the time loss related to mobility bottlenecks, *i.e.*, poor signal phase timing, traffic congestion, etc., during the emergency route rescue. The time loss is computed using the difference between the total travel time and the free-flow time of its route.

Comparison between FTAR and TMS approaches

Figures 3 and 4 show the relative rescue time and relative time loss of FTAR and TMS in both normal and rush hour traffic conditions, represented by a Cumulative Distribution Function (CDF). As it can be seen, the TMS approach can improve the time taken until the emergency vehicle arrives at the accident location in 40% of the accidents during normal traffic conditions, as depicted Figure 3(a). In other words, the emergency vehicle came faster than in the traditional baseline case in 40% of the accidents by considering TMS, *i.e.*, relative rescue time lower than 1. On the other hand, during rush hours it can improve the time taken in up to 60% of the accidents, as shown in Figure 4(a). However, it also increases the time taken to arrive at the accident location in more than 40% and 20% of the accidents during normal traffic and rush hour situations, respectively, *i.e.*, relative rescue time is higher than 1. This behavior is because TMS focuses on improving the overall traffic condition. Therefore,

some vehicles can increase their travel time to provide a better traffic balance.

Naturally, as the TMS approach does not prioritize the emergency vehicle traffic, and thus in some cases the emergency vehicle can be punished. In particular, in some cases, the time taken to reach the accident location by the emergency vehicle is increased in up to 3 times more than the time taken in the traditional baseline, as it can be seen in Figure 4(a). TMS also increases the time loss along the rescue route in 40% of the accidents during normal traffic conditions and in 30% of the accident during the rush hour, as shown in Figures 3(b) and 4(b), respectively. Unfortunately, such results show that the TMS approach is not suitable for this scenario since during rush hours the time taken to arrive at the accident location of the traditional baseline is already pretty high. Furthermore, increasing this time will delay even more the rescue of the injured person, *i.e.*, dramatically decreasing their survivability chances.

FTAR reduces the time taken to arrive at the accident location in up to 80% of the cases in both traffic situations, *i.e.*, regular traffic and rush hour. This behavior is because FTAR provides fast responsiveness for the services deployed over the fog nodes in its management area. The vehicles that are monitoring the emergency policy will be promptly notified about the rescue route and the congestion avoidance area. This management avoids traffic jams within the route planned for the emergency vehicle, thus, not delaying the rescue time. Hence, differently, from the TMS approach, the worst case for FTAR is at most the same (time that it takes to arrive at the accident location and time loss during the rescue route) compared to the traditional baseline. Moreover, it is important to mention that FTAR can minimize the time loss of an emergency vehicle by more than 50% during the typical traffic conditions, and in approximately 70% of the accidents during the rush hour. This behavior is a desirable feature to improve the survivability chances of injured.

Impact of accepting the emergency policy

Since many vehicles may not accept the emergency policy, we analyzed the

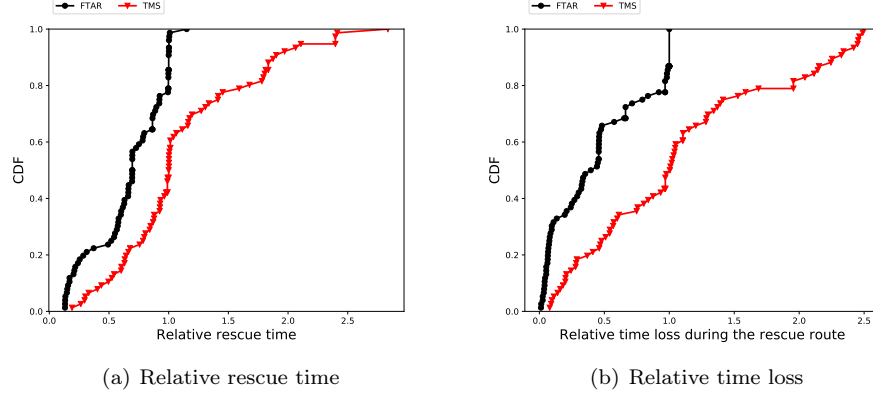


Figure 3: Results for FTAR and TMS in a normal traffic situation

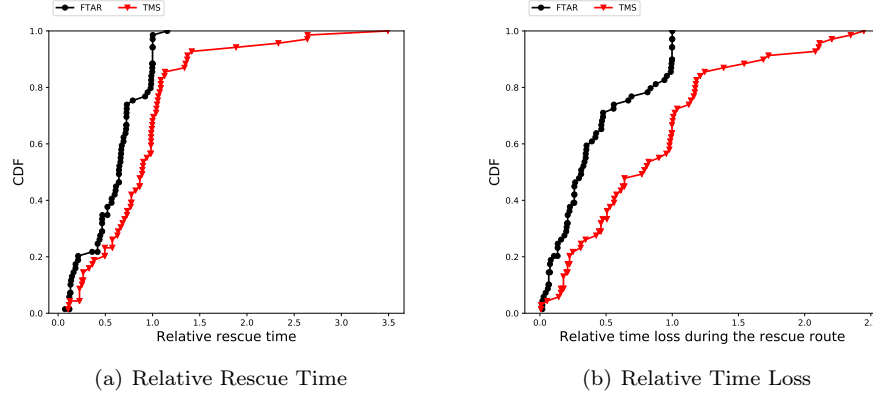


Figure 4: Results for FTAR and TMS in a Rush Hour Situation

395 impact of the policy acceptance. In this way, we conducted simulations with
 FTAR assuming that a set of vehicles will accept the emergency policy. Specif-
 ically, we define four configurations of vehicles accepting the policy, *i.e.*, 25%,
 50%, 75%, and 100% of the entire set of vehicles randomly chosen for each acci-
 dent. Figures 5 and 6 depict the results of the acceptance ratio for normal and
 400 rush hour traffic situations, respectively. As expected, we have better perfor-
 mance as soon as more vehicles accept the policy to avoid the rescue area. This
 performance is because more vehicles will be respecting the emergency policy,

and will pave the way for the emergency vehicle within the rescue route and also will avoid the congestion avoidance area.

405 FTAR reduces the time taken to arrive at the accident location in 40%, 60%, and 70% of the accidents for 25%, 50%, and 75% of vehicles accepting the policy, for both traffic situations respectively. Furthermore, FTAR also decreases the time loss in the same proportion for both cases. Let us assume an acceptance ratio of 100% to describe how the acceptance ratio impacts the

410 rescue activity efficiency. In this scenario, the time for emergency vehicle arrives at the accident location is minimized in at least 80% for 40% of accidents. Moreover, considering only 25% of vehicles accepting the emergency policy, it is still possible to decrease the time to arrive at the accident location in more than 40% of the accidents in both traffic situations, consequently, increasing the

415 survivability chances of the injured persons. These results show the efficiency of the fog-enabled VSDN approach for such scenario. In the next section, we discuss the main key research challenges to design and to deploy a Fog-enabled VSDN.

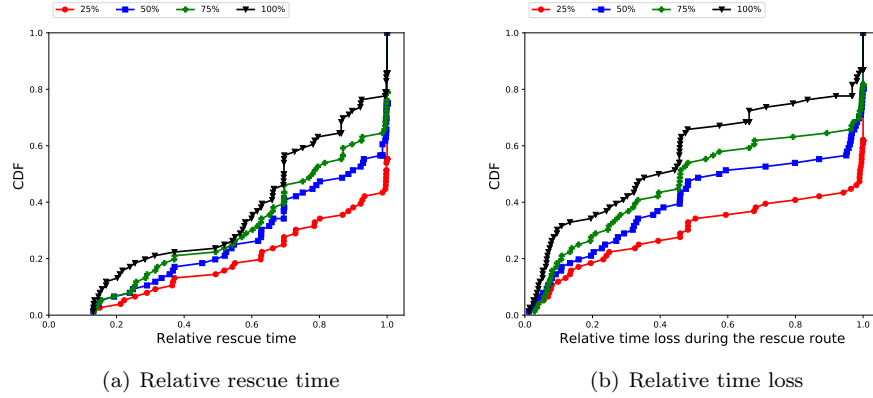


Figure 5: Results With Different Accepting Ratio in a Normal Traffic Situation

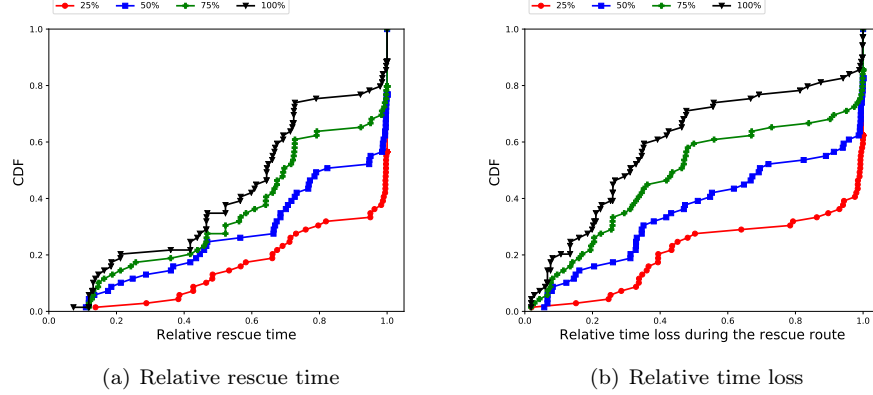


Figure 6: Results With Different Accepting Ratio in a Rush Hour Traffic Situation

5. Key Research Challenges

SDN has been emerging as a promising paradigm to control fog computing and VNs. Intelligent Transportation Systems (ITS), as well as new applications for entertainment and environment monitoring, have brought new challenges and led to fundamental and interesting research issues. We identify and discuss a non-exhaustive set of five research challenges coming from the proposed design principles to guide the use of fog-enabled VSDN, namely (i) Standardization, (ii) Fault Tolerance, (iii) Advanced Network Programmability, (iv) Incentive, and (v) Self-Driven Vehicles.

- **Standardization:** There are several initiatives of standard activities for VNs in standardization organizations. Some examples of such initiatives are IEEE Guide for WAVE (IEEE 1609) [37], ETSI ITS [38], and ISO Intelligent Transport System (IPv6 over Communications Access for Land Mobile - CALM) [39]. These initiatives define architectures and a set of services that collectively enable V2V and V2I wireless communications regarding several OSI layers. Additionally, the standards usually support IPv6-based additional network mechanisms. However, it is still necessary to provide a complete standardized framework. Besides that, as there is

no clear “winner” standard, an approach to integrate different standards is required.

- 440 • **Fault Tolerance:** Vehicles freely join and leave the wireless access to the vehicular fog as they move (churn rate). In this context, when a vehicle leaves the VSDN (and consequently, the fog), another vehicle (or set of vehicles) should be selected to substitute the services it has provided. However, fog services are prone to malicious and non-malicious faults, so that they can present failures to service consumers. Therefore, fault-tolerance features are needed to avoid unpredictable failures. The VSDN
445 control plane can aid such characteristics through the programmability of the data plane to cope with environmental dynamics.
- **Advanced Network Programmability:** Advanced network services for fog nodes can be supported in an environment with SDN. However,
450 SDN features beyond OpenFlow could further improve these fog nodes. Some examples of such features can be enabled by orchestration and Network Function Virtualization (NFV). The orchestration process (usually performed by an orchestrator) can consider information about QoE, QoS, topology, video content, operator, and others, for improved decision-making. Moreover, an orchestrator can deal with mobility and network
455 heterogeneity in lower layers in an integrated way. NFV promotes the migration from conventional network equipment to Virtual Network Functions (VNF). Such functions can be performed by software packages or in virtualized infrastructures. In this context, NFV complements SDN
460 features by introducing new network capabilities, *e.g.*, the migration of network services. Typical examples of VNFs are security-related ones, such as firewalls and Deep Packet Inspection (DPI) engines.
- **Incentives:** The fog-enable VSDN considers that vehicle could become a fog node, and thus these vehicles must be appropriately rewarded for the
465 services they provide [40]. Otherwise, fog nodes would use their valuable resources for their tasks. This reward is unaffected considering a cloud

because the incentive mechanism does not need to cope with unrestricted decentralization. Two critical aspects of the future research are (i) how to integrate the incentive with the features to program the network and (ii) how to deploy securely distributed incentive mechanisms. First, it is necessary to design mechanisms for offering incentives to encourage user participation considering how the network can be shaped to exploit such cooperation. Therefore, the motivation must consider locality issues. Since the distribution of fog nodes can be designed in a variety of ways, any monetary schemes or credit-based incentive mechanisms must take the available network services and resources in the fog region into account.

- **Self-Driving Vehicle Features:** The technological advances in automotive technology, such as self-driven vehicles, will lead the evolution in vehicular networking. This evolution is a challenge because vehicular automation processes must have access to on-line sensor measurements as well as to cloud and fog information to improve decision engines. In this context, decision-making schemes can have a significant advantage if they strike a balance between network provisioning and the use of fog resources. Vehicular fog services are produced by collaborative nodes, which can also benefit from inherent properties of self-driven vehicles. For example, the intelligence deployed to drive the vehicles can also be used to improve the orchestration capabilities.

6. Final Remarks

VNs received significant interest in the last years. Even with cloud and fog computing, technological advances are necessary to support vehicular applications with restricting QoS/QoE requirements. In this context, SDN features can provide excellent benefits over the current ossified networking infrastructure and the individual management of each VN. In this article, we advocate design principles for the integrated use of VSDN and fog computing. These principles ease the fog/cloud network integration to improve user experiences and the

efficiency of vehicular communications. Besides that, we introduced an experimental case study to analyze the feasibility of the design principles. We also presented the key research challenges regarding the fog-enabled VSDN, namely standardization, fault tolerance, advanced network programmability, incentives,
500 and self-driven vehicle features.

As future work, we intend to enhance the design principles regarding different network capabilities. For example, we plan to introduce NFV in the VSDN infrastructure. Furthermore, we are also looking at additional properties of the VSDN that could lead to significant effects. An example of such improvements
505 is the use of network programmability languages, such as P4. Additionally, we are looking for other operational scenarios regarding the computational and vehicular requirements.

References

- [1] M. Gerla, C. Wu, G. Pau, X. Zhu, Content Distribution in VANETs, Vehicular Communications 1 (1) (2014) 3–12.
510
- [2] S. Bitam, A. Mellouk, S. Zeadally, VANET-Cloud: a Generic Cloud Computing Model for Vehicular Ad Hoc Networks, IEEE Wireless Communications 22 (1) (2015) 96–102.
- [3] K. Bilal, O. Khalid, A. Erbad, S. U. Khan, Potentials, Trends, and Prospects in Edge Technologies: Fog, Cloudlet, Mobile Edge, and Micro
515 Data Centers, Computer Networks 130 (2018) 94–120.
- [4] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog Computing and Its Role in the Internet of Things, in: Proceedings of the Mobile Cloud Computing, ACM, 2012, pp. 13–16.
- 520 [5] D. Rosário, M. Schimunek, J. Camargo, J. Nobre, C. Both, J. Rochol, M. Gerla, Service Migration from Cloud to Multi-tier Fog Nodes for Multimedia Dissemination with QoE Support, Sensors 18 (2).

- [6] E. Lee, E.-K. Lee, M. Gerla, S. Y. Oh, Vehicular Cloud Networking: Architecture and Design Principles, *IEEE Communications Magazine* 52 (2) (2014) 148–155.
- [7] N. Kumar, S. Zeadally, J. J. Rodrigues, Vehicular Delay-Tolerant Networks for Smart Grid Data Management Using Mobile Edge Computing, *IEEE Communications Magazine* 54 (10) (2016) 60–66.
- [8] L. Tartarini, M. A. Marotta, E. Cerqueira, J. Rochol, C. B. Both, M. Gerla, P. Bellavista, Software-Defined Handover Decision Engine for Heterogeneous Cloud Radio Access Networks, *Computer Communications* 115 (2018) 21–34.
- [9] M. Shojafar, N. Cordeschi, E. Baccarelli, Energy-Efficient Adaptive Resource Management for Real-time Vehicular Cloud Services, *IEEE Transactions on Cloud Computing* PP (99) (2016) 1–14.
- [10] Y. Mao, C. You, J. Zhang, K. Huang, K. B. Letaief, A Survey on Mobile Edge Computing: The Communication Perspective, *IEEE Communications Surveys & Tutorials* 19 (4) (2017) 2322–2358.
- [11] R. Vilalta, V. Lopez, A. Giorgetti, S. Peng, V. Orsini, L. Velasco, R. Serral-Gracia, D. Morris, S. De Fina, F. Cugini, et al., TelcoFog: A Unified Flexible Fog and Cloud Computing Architecture for 5G networks, *IEEE Communications Magazine* 55 (8) (2017) 36–43.
- [12] F. van Lingen, M. Yannuzzi, A. Jain, R. Irons-Mclean, O. Lluch, D. Carrera, J. L. Perez, A. Gutierrez, D. Montero, J. Marti, et al., The Unavoidable Convergence of NFV, 5G, and Fog: A Model-Driven Approach to Bridge Cloud and Edge, *IEEE Communications Magazine* 55 (8) (2017) 28–35.
- [13] K. Zheng, L. Hou, H. Meng, Q. Zheng, N. Lu, L. Lei, Soft-Defined Heterogeneous Vehicular Network: Architecture and Challenges, *IEEE Network* 30 (4) (2016) 72–80.

- [14] I. Ku, Y. Lu, M. Gerla, R. L. Gomes, F. Ongaro, E. Cerqueira, Towards Software-Defined VANET: Architecture and Services, in: proceedings of 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), IEEE, 2014, pp. 103–110.
- 555 [15] E. Kalogeiton, Z. Zhao, T. Braun, Is SDN the Solution for NDN-VANETs?, in: proceedings of the 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), IEEE, 2017, pp. 1–6.
- [16] I. Zacarias, L. P. Gaspary, A. Kohl, R. Q. Fernandes, J. M. Stocchero, E. P. de Freitas, Combining Software-Defined and Delay-Tolerant Approaches
560 in Last-Mile Tactical Edge Networking, IEEE Communications Magazine 55 (10) (2017) 22–29.
- [17] Z. Zhao, E. Schiller, E. Kalogeiton, T. Braun, B. Stiller, M. T. Garip, J. Joy, M. Gerla, N. Akhtar, I. Matta, Autonomic Communications in Software-Driven Networks, IEEE Journal on Selected Areas in Communi-
565 cations 35 (11) (2017) 2431–2445.
- [18] M. Wang, H. Shan, R. Lu, R. Zhang, X. Shen, F. Bai, Real-time Path Planning based on Hybrid-VANET-enhanced Transportation System, IEEE Transactions on Vehicular Technology 64 (5) (2015) 1664–1678.
- [19] C. C. Byers, Architectural Imperatives for Fog Computing: Use Cases, Re-
570 quirements, and Architectural Techniques for Fog-Enabled IoT Networks, IEEE Communications Magazine 55 (8) (2017) 14–20.
- [20] C. A. R. L. Brennand, F. D. da Cunha, G. Maia, E. Cerqueira, A. A. F. Loureiro, L. A. Villas, FOX: A Traffic Management System of Computer-based Vehicles FOG, in: proceedings of the IEEE Symposium on Computers and Communication (ISCC), IEEE, 2016, pp. 982–987.
575
- [21] A. M. de Souza, C. A. Brennand, R. S. Yokoyama, E. A. Donato, E. R. Madeira, L. A. Villas, Traffic Management Systems: A Classification, Re-

view, Challenges, and Future Perspectives, *International Journal of Distributed Sensor Networks* 13 (4).

- 580 [22] J. Wickboldt, W. De Jesus, P. Isolani, C. Both, J. Rochol, L. Granville, Software-Defined Networking: Management Requirements and Challenges, *Communications Magazine, IEEE* 53 (1) (2015) 278–285.
- [23] N. B. Truong, G. M. Lee, Y. Ghamri-Doudane, Software Defined Networking-based Vehicular Ad-hoc Network with Fog Computing, in: proceedings of the IFIP/IEEE International Symposium on Integrated Network Management (IM), IEEE, 2015, pp. 1202–1207.
- 585 [24] P. Salvo, I. Turcanu, F. Cuomo, A. Baiocchi, I. Rubin, LTE Floating Car Data Application Off-loading via VANET Driven Clustering Formation, in: proceedings of the 12th Annual Conference on Wireless On-demand Network Systems and Services (WONS), IEEE, 2016, pp. 1–8.
- 590 [25] J. Nobre, D. Rosario, C. Both, E. Cerqueira, M. Gerla, Toward Software-Defined Battlefield Networking, *IEEE Communications Magazine* 54 (10) (2016) 152–157.
- [26] I. T. S. Data, A. Group, Technical report: Road Safety Annual Report, in: OECD, 2017.
- 595 [27] Y. Zhang, M. Chen, D. Huang, D. Wu, Y. Li, idoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization, *Future Generation Computer Systems* 66 (2017) 30–35.
- [28] A. Varga, R. Hornig, An Overview of the OMNeT++ Simulation Environment, in: proceedings of the International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops (Simutools '08), ACM, 2008, pp. 1–10.
- 600 [29] M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz, SUMO - Simulation of Urban MObility: An Overview, in: proceedings of the International

- 605 Conference on Advances in System Simulation (SIMUL '11), ACM, 2011, pp. 63–68.
- [30] C. Sommer, R. German, F. Dressler, Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis, *IEEE Transactions on Mobile Computing* 10 (1) (2011) 3–15.
- 610 [31] M. M. Haklay, P. Weber, OpenStreetMap: User-Generated Street Maps, *IEEE Pervasive Computing* 7 (4) (2008) 12–18.
- [32] A. M. de Souza, R. Yokoyama, A. Boukerche, G. Maia, E. Cerqueira, A. A. Loureiro, L. A. Villas, ICARUS: Improvement of traffic Condition through an Alerting and Re-routing System, *Computer Networks* 110 (2016) 118–
- 615 132.
- [33] A. M. de Souza, R. S. Yokoyama, G. Maia, A. Loureiro, L. Villas, Real-time path planning to prevent traffic jam through an intelligent transportation system, in: 2016 IEEE Symposium on Computers and Communication (ISCC), 2016, pp. 726–731. doi:10.1109/ISCC.2016.7543822.
- 620 [34] R. Doolan, G. M. Muntean, EcoTrec: A Novel VANET-Based Approach to Reducing Vehicle Emissions, *IEEE Transactions on Intelligent Transportation Systems* 18 (3) (2017) 608–620.
- [35] A. M. de Souza, N. L. S. da Fonseca, L. Villas, A fully-distributed advanced traffic management system based on opportunistic content sharing, in: 2017
- 625 IEEE International Conference on Communications (ICC), 2017, pp. 1–6. doi:10.1109/ICC.2017.7997071.
- [36] J. Pan, I. S. Popa, K. Zeitouni, C. Borcea, Proactive vehicular traffic rerouting for lower travel time, *IEEE Transactions on Vehicular Technology* 62 (8) (2013) 3551–3568. doi:10.1109/TVT.2013.2260422.
- 630 [37] IEEE, 1609.0-2013 - IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture.

- [38] European Standard, ETSI Technical Committee Intelligent Transport Systems (ITS) (2017).
- [39] International Organization for Standardization, ISO 21210, Intelligent transport systems – Communications access for land mobiles (CALM) – IPv6 Networking (2010).
635
- [40] W. Lobato, D. Rosario, M. Gerla, L. A. Villas, Platoon-Based Driving Protocol Based on Game Theory for Multimedia Transmission over VANET, in: proceedings of the IEEE Global Communications Conference (GLOBE-
640 COM), 2017, pp. 1–6.