C-V2X network slicing framework for 5G-enabled vehicle platooning applications

Lekidis, A. & Bouali, F.

Original citation & hyperlink:
Lekidis, A & Bouali, F 2021, C-V2X network slicing framework for 5G-enabled vehicle platooning applications. in VTC2021-Spring Workshops. IEEE, pp. 1-7, 93rd Vehicular Technology Conference, 25/04/21.
https://dx.doi.org/10.1109/VTC2021-Spring51267.2021.9448769

DOI 10.1109/VTC2021-Spring51267.2021.9448769
ISBN 978-1-7281-8964-2

Publisher: IEEE

C-V2X network slicing framework for 5G-enabled vehicle platooning applications

Alexios Lekidis
Intracom Telecom
Athens, Greece
alekidis@intracom-telecom.com

Faouzi Bouali
5G and 6G Innovation Centres (5GIC/6GIC)
University of Surrey
Guildford, United Kingdom
f.bouali@surrey.ac.uk

***Abstract*—The fifth generation (5G) of mobile networks is steering the technological evolution of many application domains, including many demanding automotive scenarios, such as platooning. Such evolution combines softwarization enablers (e.g., Network Function Virtualization (NFV) and Software-Defined Networking (SDN)) with Cellular Vehicle-to-Everything (C-V2X) communication to form end-to-end network slices with 5G infrastructure resources. Unlike the currently used 802.11p standard, the adoption of these technologies would support ultra-reliable low-latency communication (URLLC) between the platoon members and would offer better network coverage, which in turn would allow faster remote monitoring and operation of the entire platoons. In this paper, we present a novel C-V2X network slicing framework for platooning applications. The proposed framework includes a library of C-V2X Virtual Network Functions (VNFs) that can be used to customize network slices spanning over the entire 5G infrastructure. The implementation also includes edge computing functions to provide further latency and performance improvements as well as network security functions to provide cyber-resilience for the platoon. We have conducted implementation tests on a real 5G infrastructure forming a truck platoon in an automotive campus. The results demonstrate the support of URLLC on the C-V2X connectivity between trucks, which opens new horizons towards the adoption of C-V2X network slices in future platooning applications.**

***Index Terms*—Platooning, 5G, Network slicing, C-V2X, VNF, network security monitoring**

I. Introduction

Platooning is the process of linking of two or more vehicles in a convoy, using automated driving support systems and Vehicle-to-Vehicle (V2V) connectivity. Furthermore, it allows vehicles to keep close distance between each other by being connected for certain parts of a journey, while decreasing fuel consumption on motorways and easing the driving experience. In platooning applications, the vehicle at the head of the platoon acts as the leader, with the vehicles behind reacting and adapting to changes (that are communicated) in its movement without any further action from the drivers.

Platooning applications are currently based on the 802.11p standard [1], which is referred to in Europe as ITS-G5. Radio access is using a dedicated channel operating on a 10 MHz dedicated band at 5.9 GHz. However, since it relies on 802.11a, it also inherits features that are not well-suited for V2V connectivity, such as the use of Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) which does not perform well in congested scenarios [2]. Hence 802.11p is not scalable to large-scale V2V scenarios. Additionally, the high frequency of V2V transmissions increases substantially the probability of packet collisions. The other connectivity option that has recently emerged is the Cellular Vehicle-to-Everything (C-V2X) standard defined by 3GPP [3]. In addition to its infrastructure communication mode, the C-V2X option does include a direct communication mode through the PC5 interface [4]. This ensures a more efficient radio resource allocation, which provides more scalability and reliability for the platoons.

Up to the third-generation partnership project (3GPP) release 15 [5], C-V2X uses the 5G Non-Standalone (NSA) option with LTE as an underlying technology. However, release 16 has defined specifications for the use of the 5G Standalone (SA) option with the 5G New Radio (NR) and 5G Core (5GC) infrastructure. Moreover, the Network Function Virtualization (NFV) technology has started to become available for V2V scenarios to customize network slices that include mobile core, transport, Radio Access Network (RAN) as well as edge computing components in the form of Virtual Network Functions (VNFs). These VNFs allow the co-existence and sharing of infrastructure resources that belong to multiple operators. Since platooning are critical applications the formed network slices belong to the Ultra-Reliable Low-Latency Communication (URLLC) category.

The optimal utilization and resource sharing approaches of NFV come with severe security risks, such as the interception of user data from tenant network slices or Denial of Service (DoS) attacks using control commands against the 5G infrastructure [6]. Most of these risks lie on the broadcast and unsecured nature of V2V data exchange, which can be easily eavesdropped. To overcome such limits, a network security library should be developed to ensure the early detection and response against both known and zero-day (i.e., unknown) threats.

In this paper, we propose a solution to these challenges by introducing a library that is implementing C-V2X connectivity for platooning applications. The library defines C-V2X network functions that can be used to customize 5G network slices, which include core, transport, RAN as well as edge VNFs. Moreover, a network security monitoring VNF is included to allow the protection of both the internal vehicle architecture and the C-V2X connections. Specifically, this paper has the following concrete contributions:

- Construction of a novel C-V2X network slicing framework for platooning applications. The constructed framework includes a library of C-V2X VNFs for platooning applications.
- Network slice extension at the edge for fast delivery of configurations and road emergency messages to the platoons and task offloading from the Mobile Core whenever needed.
- Network security monitoring VNF for cyber-resilience against C-V2X threats from abnormal or suspicious activities during the platoon operation.
- Application of the proposed methodology to an illustra-

tive truck platooning use case on a 5G SA deployment with edge computing support. Preliminary performance measurements, including slice instantiation/termination times, reliability and latency, are presented and discussed.

The rest of the article is organized as follows. Section II provides background information on platooning applications and the C-V2X connectivity layers. Section III presents the library that was developed for covering all the connectivity layers of truck platooning applications. The library is tested in Section IV for an illustrative truck platooning use case on a real 5G SA infrastructure and C-V2X performance measurements are presented. Finally, Section V provides conclusions and some perspectives for future work.

II. Background information

A. Cooperative adaptive cruise control and platooning

When looking at V2V applications, vehicle platooning belongs to the category of Cooperative Adaptive Cruise Control (CACC). In contrast to the normal Cruise Control, Adaptive Cruise Control (ACC) can adjust the vehicle speed based on the distance to the other vehicles lying in front of it, i.e., headway distances. The distance detection relies on radar sensors mounted at the front of the vehicle. Additionally, due to the latency from the moment when a front vehicle brakes to the moment when an ACC-enabled vehicle reacts to the decreased headway, safe following distance is still quite high in ACC systems. However, in CACC (e.g., platooning), when vehicles communicate their intended acceleration/deceleration, which helps to further reduce the safe following distances, also leading to a decrease in fuel usage. This is also the main reason why message reception reliability, rate and latency are very important requirements in such scenarios.

Since platooning applications need to satisfy critical real-time requirements, they are using safety-oriented V2V communication, which is illustrated in Fig. 1. The safety-oriented messages included in this figure are described in Section II-B and are based on [7]. Non-safety-oriented messages can be also exchanged in platooning, but they are rare and usually follow the Transmission Control Protocol (TCP) and IPv4/IPv6 protocols.

B. C-V2X connectivity layers

Since C-V2X is based on the cellular PC5 interface for radio access, the network and transport layer as well as the payload inside the packets is defined by the GeoNetworking protocol. This protocol is standardized by European Telecommunications Standards Institute (ETSI) EN 302 636-4-1 [8]. This standard specifies the Geographically scoped information dissemination and packet routing mechanisms using geographical positions. Moreover, its implementation is considered mandatory throughout Europe as a part of the Intelligent Transport System (ITS) infrastructure. GeoNetworking technologies are typically used to complement IPv6 technologies for ensuring the availability of the wireless medium in the presence of many vehicles. GeoNetworking also includes IPv4 addressing in certain scenarios. GeoNetworking packets contain different fields that are encoded using the Abstract Syntax Notation (ASN.1).

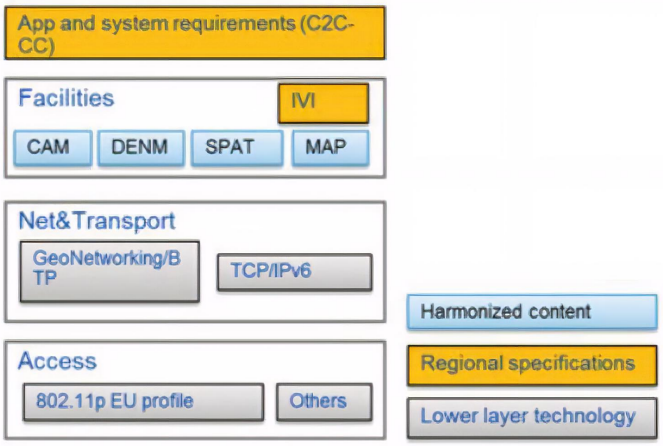

Fig. 1. Platooning connectivity

GeoNetworking includes four dedicated transport schemes that are presented in Fig. 2. Each transport scheme has a different usage and is defined to enable the dynamicity and reconfigurable architecture of V2V and Vehicle to Infrastructure (V2I) communication.

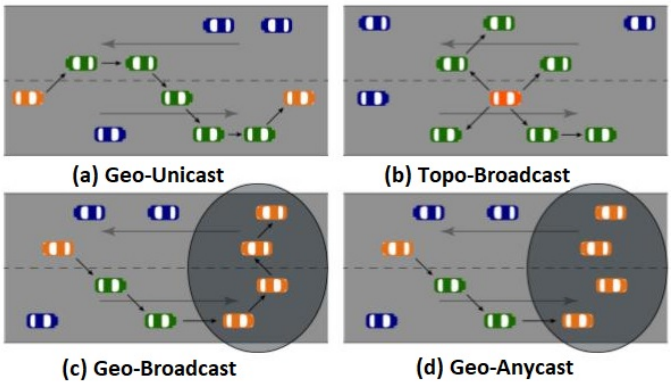

Fig. 2. GeoNetworking transport schemes

Fig. 2 defines four main schemes. Initially, the Geo-Unicast defines packet delivery to a given node in a certain geographic location. Then, the Topo-Broadcast defines packet delivery to all nodes located up to a certain distance. The Geo-Broadcast is used for packet delivery to all nodes within a certain geographic area and finally the Geo-Anycast for packet delivery to at least one node within a certain geographic area.

The C-V2X facilities layer includes two main messages:

1) Cooperative Awareness Message (CAM), which is providing vehicle presence information as for example the collision avoidance messages and has been standardized by ETSI EN 302 637-2 [9] and
2) Decentralized Environmental Notification Message (DENM), which is providing environmental hazards notifications as for example traffic conditions, notification of a accidents on the road and has been standardized by ETSI EN 302 637-3 [10].

CAM and DENM messages define data encoding according to the types and formats specified in the Common data dictionary (ETSI TS 102 894-2 [7]). The data structure inside the messages are called as containers, which are also specified through the ASN.1 notation.

Specifically, the CAM containers are organized into three main categories: 1) the high frequency container for rapidly changing vehicle state data, 2) the low frequency container for slowly changing (static) vehicle state data and 3) the Special Vehicle Container for additional vehicle-specific state information.

Likewise, the DENM containers are organized into four main categories: 1) the management container with essential situation information, 2) the situation container with information on situation type and event position, 3) the location container with information for moving situations and finally 4) the Alacarte Container with use-case specific information.

III. C-V2X SLICING FRAMEWORK FOR VEHICLE PLATOONING

In this section, we present the architecture of the proposed slicing framework for platooning applications. Specifically, it includes a library of VNFs together with OnBoard Units (OBUs) that are placed inside vehicles (e.g., trucks) and a RoadSide Unit (RSU) equipped with edge computing support and connected to Mobile Core/Cloud. The OBUs communicate between them and with the RSU over C-V2V and C-V2I links, respectively. The latter link allows the platoons to receive configurations and road emergency messages. Furthermore, the RSU receives control and data commands from the 5G Core that resides on a Cloud infrastructure. Similarly, the RSU facilitates fleet management by sending diagnostics to the Cloud, which allows to identify issues on individual vehicles (e.g., trucks) or the entire platoon. Moreover, we have placed network probes in the RSU and the Mobile Core/Cloud for monitoring message exchange, performing data analytics and measuring the 5G network Key Performance Indicators (KPIs). The architectural deployment is presented in Fig. 3.

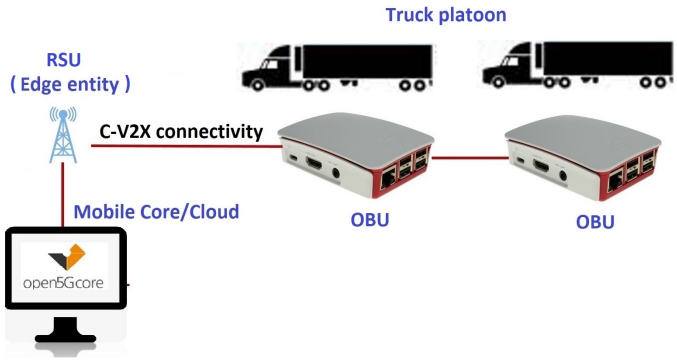

Fig. 3. Architectural deployment for the considered platooning application

A. C-V2X protocol stack

The following part presents the protocol stack implementation to support platooning communication. The protocol stack implementation is based on 3GPP Release 15 C-V2X [5], but is easily extensible to support the improvements brought by subsequent releases. The main focus of this section is on code snippets and detailed representation of the messages that are exchanged between the vehicles. This allows a comprehensive view of the connectivity interfaces as well as network message encoding for platooning applications.

Initially, as GeoNetworking is based on the Ethernet link type *8947*, the proposed implementation is based on the use of Linux raw sockets. These raw sockets are used for exchanging packets in lower layer protocols, such as the Medium Access Control (MAC). In the following code snippet, we illustrate a simple implementation (in C) to emulate a GeoNetworking node that uses the native Ethernet of a Linux machine for packet transmission:

```
1  unsigned char geonetworking[40] = {0x01,0x00,0xF1,0x01,
2  0x00,0x10,0x00,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0xE2,
3  0xDC,0x99,0xFA,0x00,0x7B,0x9E,0xE9,0x14,0x00,0x1C,0x6D,
4  0xD2,0x44,0x03,0xE5,0x68,0x22,0x00,0x00,0x00,0x00};
5
6  /* Get interface name */
7  if (argc > 1)
8          strcpy(ifName, argv[1]);
9  else
10         strcpy(ifName, DEFAULT_IF);
11
12 /* Open Linux raw socket to send on */
13 if ((sockfd = socket(AF_PACKET, SOCK_RAW, IPPROTO_RAW))
14         == -1) {
15         perror("socket");
16 }
17 strncpy(if_mac.ifr_name, ifName, IFNAMSIZ-1);
18 if (ioctl(sockfd, SIOCGIFHWADDR, &if_mac) != -1) {
19
20         eh->ether_type = 0x4789;
21         /* Send packet */
22         if (sendto(sockfd, sendbuf, tx_len, 0,
23         (struct sockaddr*)&socket_address,
24         sizeof(struct sockaddr_ll)) < 0)
25                 printf("Send failed\n");
26 }
```

Listing 1. GeoNetworking node example

In lines 1-4, the entire payload of the GeoNetworking packet is configured. Then, the program gets the name of the provided interface that is usually provided through the command line (lines 7-10), and subsequently opens a Linux raw socket. The socket is bound to the interface name and then the Ethernet type of the message is properly set (line 20). As final step, the socket is opened to transmit the packet (lines 22-25).

In its current version, the library does not provide a systematic mechanism to switch between different GeoNetworking modes. It can be however used to analyse packet encoding inside the GeoNetworking messages as well as the overall message latency. Fig. 4 illustrates an example of a GeoNetworking packet that is responsible for announcing the presence of a new vehicle to the network. To properly decode this V2X message using the Wireshark packet analyser[1], an ITS plugin has been developed, which is complaint with the ETSI EN 302 637-2 [9] and ETSI EN 302 637-3 standards [10]. The plugin allows to dissect V2V messages encoded with the ASN.1 notation.

[1]https://www.wireshark.org/

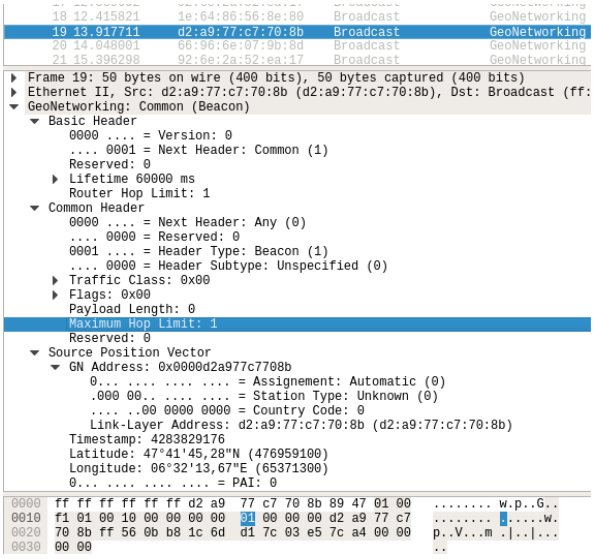

Fig. 4. V2V GeoNetworking message in Wireshark packet analyzer

Although the encoding and transmission of networking and transport layer messages (as with GeoNetworking) can be done through the use of common interfaces as the Linux raw sockets, the same doesn't apply for the messages of the facilities layer shown in Fig. 1. Since the latter layer is the most vital for the configuration of important characteristics for the functionality of the platoon (i.e., vehicle position, speed and acceleration), we have developed libraries for safety-oriented V2V/V2I communication to support this layer.

To provide data configuration for the platooning scenario, we exploited the ETSI mapping between the CAM and DENM containers information and the in-vehicle data (e.g. speed, acceleration, engine control) [7] [11]. To this end, we have analysed how the in-vehicle pieces of information are mapped to containers of the CAM and DENM messages, a process that resulted in deriving the initial mapping presented in Table I.

In-vehicle information	CAM Container	DENM Container
Speed	High Frequency Container	-
Acceleration Control	High Frequency Container	-
Steering Wheel Angle	High Frequency Container	-
Exterior lights	Low Frequency Container	-
Event Speed	-	Location Container
Stationary Cause	-	A la carte Container
Vehicle Identification	-	A la carte Container

TABLE I
IN-VEHICLE INFORMATION TO V2V MAPPING

From Table I, we can infer that changes on safety-related in-vehicle data are propagated to nearby vehicles to introduce awareness (e.g., Speed, Exterior lights and Vehicle Identification) or inform about emergency situations (e.g., Event Speed and Stationary Cause). These pieces of information were used to configure the platooning application with data originating from a real in-vehicle architecture that has been integrated into our laboratory facility. Alternatively, the configuration data can be logged through the OnBoard Diagnostics (OBD-II) port [12].

Our experiments have focused on analysing the data that are monitored through network probes placed in the RSU and the Mobile Core/Cloud. Specifically, we have analysed the CAM and DENM messages in terms of the encoding of the information that originated from the in-vehicle architecture, as well as the parameters that are specific to the platooning communication scenario. The latter have focused only on the containers that are necessary for the construction of such messages.

The platooning setup is using DENM messages to signal emergency conditions that are interrupting the heading of the platoon (e.g. stationary vehicle conditions). These messages originate from the nearby RSUs and contain the structure of Fig. 5. Specifically, they include management information about the action that happened (*actionID*) and the station which is reporting it (*originatingStationID*). In this specific scenario, the station that is reporting the incident is the RSU (*originatingStationID = 15*). Additionally, incident information, such as its position (eventPosition) and station type (stationType), are included. In this message, the incident is reported by a truck vehicle inside the platoon, hence according to ETSI EN 302 637-3 standard [10], it is equal to *lightTruck*.

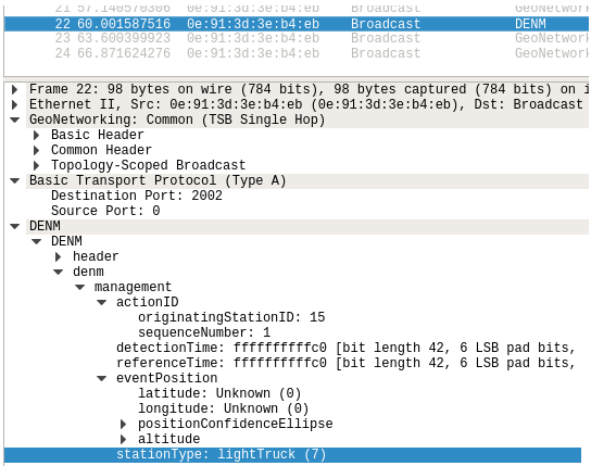

Fig. 5. V2V DENM message from Wireshark packet analyzer

B. 5G network slicing with edge computing resources

The establishment of a network slice for platooning applications includes the 5G core network functions [5] augmented by a TCP connectivity interface for the transport layer as well as the radio access layers. Additionally, it is also extended by edge entities which manage the lifecycle of edge resources. These resources provide latency and performance improvements for the platoon, but also offload processing and message exchange functions to edge Points of Presence (PoPs). Each edge PoP interacts with an instance of an edge Virtual Infrastructure Manager (VIM), that is referred to as Mobile Edge Platform (MEP). The MEP instance receives instructions for configuring the edge resources and including them into a network slice from the NFV Management and Orchestration (MANO) framework [13], responsible for the orchestration of the Mobile Core/Cloud resources. The interconnection of

the Mobile Core/Cloud and each Edge PoP is handled by two individual micro-services (i.e., Cloud service in Mobile Core/Cloud and Edge service in Edge PoP) that include an Application Programming Interface (API) to provide connectivity.

Through the interconnection, the VNFs running on the MANO framework are transferred to the Edge PoP. Consequently, they are transformed into edge functions, such as the RAN, C-V2X, a local User Plane Function (UPF) developed for packet processing and the network monitoring security module presented in Section III-C (i.e., SEC-MONITOR VNF). Specifically, the edge entity receives instructions through the API in the form of Network Slice Templates (NSTs) that MANO creates. These NSTs contain slice-specific inputs and attributes and are translated into Application Descriptor (AppD) instructions, which guide the edge entity on how to configure, instantiate and connect the VNFs using virtual links. Upon successful operation, the edge entity informs the MANO, which activates the slice.

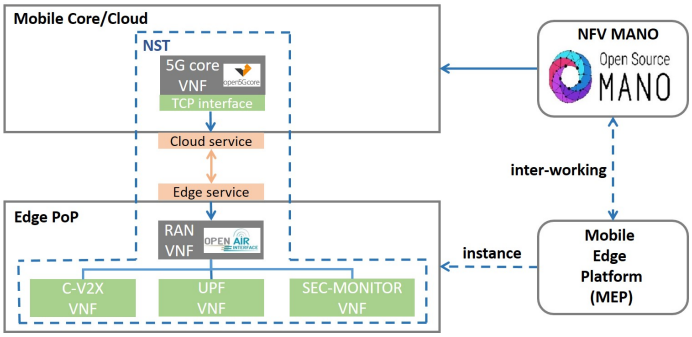

Fig. 6. Network slice establishment with edge resources

C. Network security monitoring

Another key aspect to be considered for platooning applications is the underlying security risks from the exchange of V2V data. To handle this aspect, we have implemented a Network Intrusion Detection System (NIDS) that monitors data and control commands that are exchanged between the entities depicted in Fig. 3. The NIDS is implemented as a network monitoring security VNF (i.e., SEC-MONITOR VNF) and more specifically as a dockerized Linux systemd service. Moreover, it can be deployed inside (1) the *5G Mobile Core/Cloud*, (2) the *RSU (edge entity)* or (3) the *OBUs*.

All the above options allow the detection of C-V2X, ITS-G5 as well as non-safety message threats (i.e. WiFi-based through TCP). However, in contrast to the first two options, only the OBU deployment is able to detect in-vehicle threats.

Fig. 7 describes the OBU deployment of the proposed platooning NIDS. Initially, it uses parser modules to interpret the exchanged network commands and data as well as to learn the normal operation of the platooning application. Then, whenever an abnormal or suspicious event is spotted on the network, it signals a warning or alert to the MANO framework residing in the Mobile Core/Cloud or the RSU edge entity. The warning or alert is following the Common Event Format (CEF) standard[2]. Based on the alert severity, appropriate countermeasures are taken to ensure the reliable and continuous platooning operation.

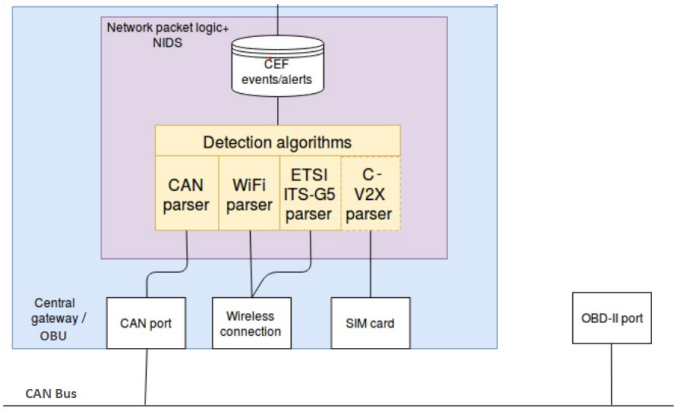

Fig. 7. NIDS for vehicle platooning applications

The main V2X network security monitoring modules shown in Fig. 7 are:

- ETSI ITS-G5 detector (safety-oriented): It consists of the ETSI ITS-G5 parser to interpret the safety-oriented V2X messages (complying to the ETSI EN 302 637-2/3 standards [9] [10]) and detection algorithms for the abnormal messages among them.
- C-V2X detector: It includes the C-V2X parser and detection algorithms for detecting abnormal C-V2X messages that are exchanged in V2X connectivity scenarios.
- ETSI ITS-G5 detector (non safety-oriented): It is composed of the WiFi parser to interpret the non-safety-oriented V2X packets using TCP transport and detection algorithms to detect the abnormal messages among them.

The in-vehicle intrusion detection module, connected to the vehicle OBD-II port, monitors the CAN Bus using a CAN message parser [12]. The interested reader is referred to [14] for more details about this module.

The additional security mechanisms needed to ensure end-to-end security for platooning applications include a Public Key Infrastructure (PKI)-based security mechanism for the authentication of platooning members [15], followed by the set-up of symmetric encryption for the exchange of the platooning messages during the process of 'associating' the vehicles to the platoon.

IV. 5G TESTBED EVALUATION

The proposed methodology has been applied to an illustrative truck platooning use case on a real 5G infrastructure.

The testbed that is used for our experiments (Fig. 8) is based on the 5G field laboratory of the automotive campus in Helmond, Netherlands. Furthermore, the field laboratory is also is extended to include an RSU supported with edge computing facilities. A campus building hosts the 5G core that is using the Open5GCore toolkit of Fraunhofer FOKUS[3].

[2]https://ldapwiki.com/wiki/Common%20Event%20Format
[3]https://www.open5gcore.org/

The core is installed in a desktop computer with 2.9 GHz processing power (Intel Core i5), 16 GB DDR3 RAM and 120 GB SSD root disk. For the RAN, we used the open-source OpenAirInterface implementation, which supports 5G NR access [16]. Moreover, the RSU is also a desktop computer with 1 CPU, 4 GB RAM, 28 GB disk, an antenna and a SIM card slot for C-V2X connectivity. The OBUs are based on a Raspberry Pi Model 3 with a cellular adapter and are connected to the OBD-II port of the trucks for the reception of in-vehicle messages. The exchange of CAM messages was periodic (i.e., every 0.1 s) and DENM asynchronous based on occurred incidents.

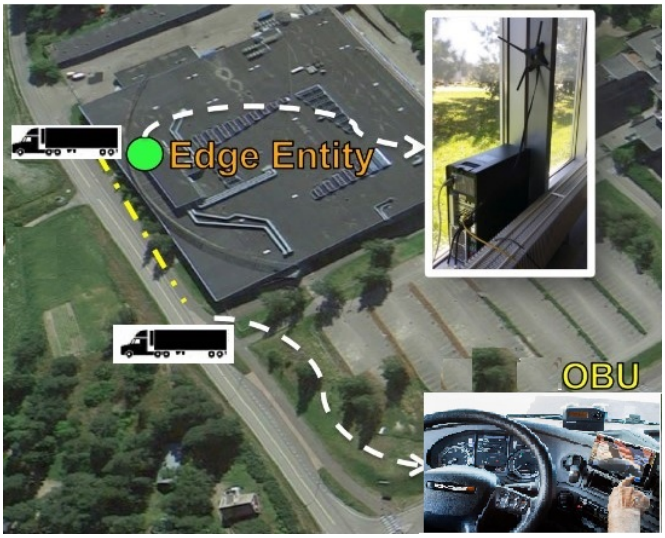

Fig. 8. 5G vehicle platooning testbed

The creation of the URLLC slice (Fig. 6) is based on NSTs of an ETSI Open-Source MANO (OSM)[4] installation that is also used to manage and orchestrate the 5G network resources (e.g. VNFs and Virtual Links). In total, the truck platooning application used six VNFs, namely two for the C-V2X connectivity in the OBUs and one for the UPF, 5G core, RAN and network security module, respectively. By deploying the platooning application on the Fig. 8 testbed, we are able to test the relevant KPIs, including: (1) Slice Instantiation/Termination times, (2) End-to-end latency and (3) Reliability (i.e., probability of successful reception at a given latency target).

The considered KPIs were monitored during the following two sets of experiments:

- *5G core only*: In this experiment, we used only OSM to instantiate the slice VNFs and virtual links to inform directly the truck OBUs.
- *5G core+edge*: In this experiment, we have instantiated the RAN, UPF and network security VNFs in the edge entity, which is co-located with the RSU that forwards the commands to the truck OBUs. With this deployment, we can test the edge computing improvements in terms of latency and performance for the truck platoon.

Fig. 9 illustrates the time duration for the instantiation of each VNF and the associated virtual links for both sets of experiments.

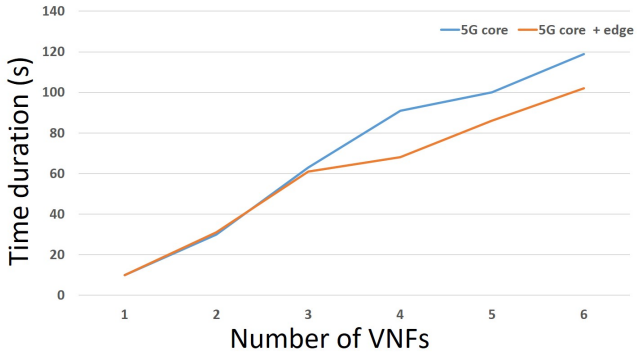

Fig. 9. 5G network slice instantiation time with edge-Core resources

As depicted in Fig. 9, the time for the instantiation of each VNF is stacked. We can also note that, in the second set of experiments (i.e., *5G core+edge*), the main difference lies after the instantiation of the third VNF, where a communication with the edge entity is taking place through the API mentioned in Section III-B. Then, the edge entity uses less time duration for instantiating the rest of the VNFs resulting overall in approximately 20 s latency gain for the URLLC slice creation. Additionally, we have conducted measurements of the slice termination time with an average time of 34 s without the edge entity and 26 s with it.

For the message latency of the formed URLLC slice, without edge support (i.e., *5G core only*), we have measured an average end-to-end time latency of 8.1 ms for GeoNetworking as well as 9.7 ms for the CAM and DENM messages as shown in Fig. 10(a). The presence of the edge entity (i.e., *5G core+edge*) in Fig 10(b) allows the reduction of the average latency to 4.4 ms for GeoNetworking and 5.2 ms for CAM and DENM messages. The performance for both scenarios is satisfying the targeted KPI of 10 ms set by 3GPP for platooning applications [17]. In our view, further latency improvements can be achieved by applying automation techniques, such as intent-based policies [18], in the communication between the Mobile Core and the edge entity, which is left for future consideration.

For the reliability assessment, we have set the latency target to 10 ms and evaluated the probability of successful reception at the network and transport layers (i.e. GeoNetworking messages) as well as the facilities layer (i.e. CAM and DENM messages). The achieved reliability was 99.988% for both GeoNetworking as well as CAM and DENM messages. This was caused by radio channel interference leading to some messages being lost. The observed performance was not impacted by the presence of the edge entity, however it already ensures high reliability for the truck platooning application. As a matter of fact, the achieved reliability almost hits the target

[4]https://osm.etsi.org/

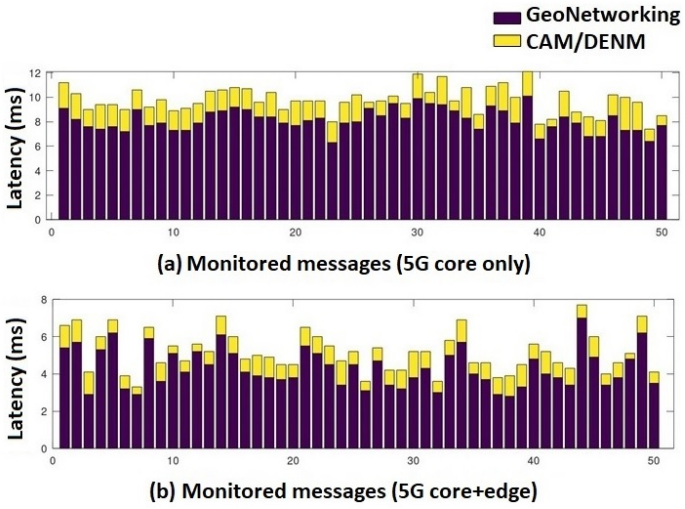

Fig. 10. Message latency results for the URLLC slice

of 99.99% set by 3GPP for the highest degree of automation [17]. We believe that the application of network slice isolation policies will minimize radio channel interference and further improve the achieved reliability.

V. Conclusions and Future Work

This paper constructs a novel Cellular Vehicle-to-Everything (C-V2X) network slicing framework for platooning applications. The constructed framework is compliant with the relevant European Telecommunications Standards Institute (ETSI) standards and uses a Mobile Core to enable 5G communication. It includes a library of virtual network functions (VNFs) that can be used to customize network slices spanning over the entire 5G infrastructure. In particular, a set of Edge Computing entities are deployed to enable fast delivery of configurations and road emergency messages to the platoons and task offloading from the Mobile Core whenever needed. The framework includes a set of Network security monitoring functions for real-time detection and prevention of cyber-threats from abnormal or suspicious activities during the platoon operation. The proposed methodology has been applied to an illustrative truck platooning use case on a real 5G Standalone (SA) testbed that is installed in an automotive campus. A set of experiments have been conducted and the most relevant Key Performance Indicators (KPIs), including the slice instantiation and termination times, communication reliability and end-to-end latency, have been monitored. The results demonstrate that the proposed methodology helps to meet the ultra-reliable low-latency communication (URLLC) requirements between the trucks, which opens new horizons towards the adoption of C-V2X network slices in future platooning applications.

As a part of our future work, we plan to extend the interface between the the Mobile Core and Edge entity with intent-based policies that will provide automation on the slice creation, diagnostics and maintenance. With this increased level of automation, further performance improvements will be investigated. Finally, while the current methodology is based on ETSI Zero Touch Network and Service Management (ZSM) Working Group [18], further extensions will be based on isolation policies for network slices, such that the optimal sharing of the 5G infrastructure resources is achieved.

Acknowledgment

This work has been partially founded by the 5G-HEART project, that has received Grant Agreement No. 857034 by the European Commission Horizon 2020 Research and Innovation Framework Programme.

References

[1] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Vehicular Technology Conf., 2008. VTC Spring 2008. IEEE*. IEEE, 2008, pp. 2036–2040.

[2] V. Vukadinovic, K. Bakowski, P. Marsch, I. D. Garcia, H. Xu, M. Sybis, P. Sroka, K. Wesolowski, D. Lister, and I. Thibault, "3GPP C-V2X and IEEE 802.11p for Vehicle-to-Vehicle communications in highway platooning scenarios," *Ad Hoc Networks*, vol. 74, pp. 17 – 29, 2018.

[3] S. Husain, A. Kunz, A. Prasad, K. Samdanis, and J. Song, "An overview of standardization efforts for enabling vehicular-to-everything services," in *2017 IEEE Conf. on Standards for Comm. and Net. (CSCN)*, 2017, pp. 109–114.

[4] R. Molina-Masegosa and J. Gozalvez, "LTE-V for sidelink 5G V2X vehicular communications: A new 5G technology for short-range vehicle-to-everything communications," *IEEE Vehicular Technology Magazine*, vol. 12, no. 4, pp. 30–39, 2017.

[5] 3rd Generation Partnership Project, "Technical Specification Group Services and System Aspects; Release 15 Description; Summary of Rel-15 Work Items (Release 15)," 2019.

[6] ENISA, "ENISA Threat Landscape for 5G Networks Report: Updatedthreat assessment for the fifth generation of mobile telecommunications networks (5G)," Tech. Rep., December 2020.

[7] ETSI, "TS 102 894-2 V1. 3.1," *Intelligent Transport Systems (ITS): Users and applications requirements*, 2018.

[8] ETSI, "EN 302 636-4-1," *GeoNetworking Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications*, 2017.

[9] ETSI, "EN 302 637-2," *Intelligent Transport Systems (ITS) Vehicular Communications Basic Set of Applications Part 2: Specification of Cooperative Awareness Basic Service*, 2014.

[10] ETSI, "EN 302 637-3," *Intelligent Transport Systems (ITS) Vehicular Communications Basic Set of Applications Part 3: Specifications of Decentralized Environmental Notification Basic Service*, 2014.

[11] A. Lekidis and P. Katsaros, "Energy characterization of IoT systems through design aspect monitoring," *Software Tools for Technology Transfer (STTT)*, 2021.

[12] A. Lekidis and I. Barosan, "Model-based simulation and threat analysis of in-vehicle networks," in *2019 15th IEEE International Workshop on Factory Communication Systems (WFCS)*. IEEE, 2019, pp. 1–8.

[13] R. Mijumbi, J. Serrat, J.-L. Gorricho, S. Latré, M. Charalambides, and D. Lopez, "Management and orchestration challenges in network functions virtualization," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 98–105, 2016.

[14] G. Dupont, J. Den Hartog, S. Etalle, and A. Lekidis, "Evaluation Framework for Network Intrusion Detection Systems for In-Vehicle CAN," in *2019 IEEE International Conf. on Connected Vehicles and Expo (ICCVE)*. IEEE, 2019, pp. 1–6.

[15] P. Cincilla, O. Hicham, and B. Charles, "Vehicular PKI scalability-consistency trade-offs in large scale distributed scenarios," in *2016 IEEE Vehicular Networking Conf. (VNC)*. IEEE, 2016, pp. 1–8.

[16] F. Kaltenberger, G. de Souza, R. Knopp, and H. Wang, "The OpenAir-Interface 5G new radio implementation: Current status and roadmap," in *WSA 2019; 23rd International ITG Workshop on Smart Antennas*. VDE, 2019, pp. 1–5.

[17] 3rd Generation Partnership Project, "Technical Specification Group Services and System Aspects; Enhancement of 3GPP support for V2X scenarios; Stage 1 (Release 16), 3GPP TS 22.186," June 2019.

[18] ETSI, "Zero-touch network and Service Management (ZSM): Means of Automation," Tech. Rep., May 2020.