Design and Evaluation of a Lightweight SDN Controller for Integrated Road and Rail Networks

Dingyang Liu[1], Dereje Mechal Molla[1][0000−0003−4515−679X], Leo Mendiboure[2][0000−0001−6643−9567], Sassi Maaloul[2,3][0000−0002−7190−9989], Marion Berbineau[1][0000−0003−3807−9669], and Hakim Badis[4][0000−0002−1792−6198]

[1] Univ. Gustave Eiffel, COSYS-LEOST, F-59650 Villeneuve d'Ascq, France
[2] Univ. Gustave Eiffel, COSYS-ERENA, F-33067 Pessac, France
[3] JUNIA Engineering School, 59000 Lille, France
[4] Univ. Gustave Eiffel, LIGM, 77454 Marne La Vallée, France

Abstract. Implementing reliable communications is a major challenge for rail and road communications networks. Current V2X (Vehicle-To-Everything) technologies come up against major limitations. These include the existence of white zones and incomplete coverage by cellular networks (4G/5G). This implies the joint use of multiple access technologies at the level of each vehicle and a real-time and transparent transition between these radio access technologies. The definition of optimal handover mechanisms based on Artificial Intelligence (AI) tools is part of the answer to this problem. However, the definition of a high-performance network manager, guaranteeing low latencies, is also an important challenge. Therefore, this paper proposes a novel Software-Defined Network (SDN) controller designed to optimise dynamic handovers between wireless interfaces based on real-time network conditions. It also considers Machine Learning (ML) to determine performance parameter requirements. This approach improves the reliability, flexibility and adaptability of in-vehicle communication systems, helping to increase the efficiency of Cooperative Intelligent Transportation Systems.

Keywords: SDN Controller · Rail and Road · V2X Connectivity technologies · Machine Learning · Radio Interfaces Management.

1 Introduction

Demand for reliable and efficient communication in vehicular networks grows because of the emergence of critical applications with important Quality of Service (QoS) requirements [1]: remote/tele-operated driving, automated vehicles management, platooning, etc. What's more, these services are increasingly tending

CORRESPONDING AUTHOR: Leo Mendiboure (e-mail: leo.mendiboure@univ-eiffel.fr).

to be multi-modal (vehicles that can operate on both rail and road), as demonstrated by the many current projects aimed at developing such systems, notably the FerroMobile and Flexy projects in France [2, 3].

Different V2X communication technologies such as ITS-G5 and C-V2X [4, 5] could be seen as a way to achieve the expected level of performance for multi-modal vehicular services. ITS-G5 provides low-latency, short-range communications and is suitable for safety-critical applications, while C-V2X offers better integration with existing cellular networks and wider coverage. However, both technologies have limitations, including limited network coverage, unstable performance at high speeds, difficulties with channel resource allocation, and security issues. Moreover, these technologies need more standardization, and seamless interoperability poses significant challenges in mixed rail and road networks [6].

Previous studies have proposed some methods to address these challenges, but there are still limitations. For example, [7] used the M/G/1 queueing model to analyze network connectivity, showing that proper base station numbers and transmission range can achieve desired connectivity. However, this costly solution seems unrealistic. Alternatively, another idea that has emerged is that the vehicle can draw on all the access technologies available by modifying the behaviour of its Radio interface as required [8]. Software Defined Networking (SDN), decoupling network control plane and data plane, is the technology that has been most widely considered for such an implementation. Indeed, it enables dynamic, software-based management of radio interfaces. However, work to date has been based solely on existing controllers (e.g. Ryu, ONOS [9]) not designed for high-mobility environments such as the vehicle, and therefore leading to significant delays that are too high for critical applications [10]. Furthermore, as highlighted in the related works, studies aimed at optimising these systems has not yet focused simultaneously on the ideas of mobility management, real-time interface management and taking account of multiple access technologies.

This study proposes a new framework for road and rail environments based on a novel SDN controller that ensures fast response and stable and reliable communication in highly dynamic environments. The SDN controller issues flow rules for interface switching based on the communication data received from the SDN enabled nodes. This aims at improving the reliability, scalability and efficiency of the on-board communication system. Both the operation, an implementation and an initial evaluation of the article are presented here. This work could represent a first step towards the definition of an SDN controller specifically adapted to the vehicular environment.

The rest of this paper is structured as follows: Section II reviews related work. Section III presents the network architecture and SDN integration. Section IV covers implementation details, and Section V evaluates system performance. Finally, Section VI concludes the paper and outlines future directions.

Table 1. COMPARISON OF RELATED WORK (Sorted by Similarity to Research)

References	Research Content	Innovation Points	SDN-Based Technology	Real-Time Dynamic Management	Multi-Network Integration	Adaptive Interface Selection
[11]	Proposes redundant architecture to improve autonomous vehicle safety	Introduces hardware redundancy and communication fault tolerance	×	×	✓	×
[12]	Proposes SDN-based architecture for managing sensor networks in hazardous environments	Introduces fault recovery and resource management for static networks	✓	×	×	×
[13]	Discusses cellular V2X challenges like network congestion and resource allocation	Proposes solutions to enhance V2X performance	×	×	✓	×
[14]	Proposes SDN framework for Li-Fi and Wi-Fi integration	Cross-layer optimization between Li-Fi and Wi-Fi	✓	×	✓	×
[15]	Proposes cluster-based radio resource management in dynamic vehicular networks	Introduces a vehicle matching algorithm and dynamic clustering for link stability	×	✓	×	✓
[16]	Compares SDN controllers in safety-critical vehicular applications, focusing on delay and throughput	Benchmarks SDN controllers for vehicular networks	✓	✓	×	✓
[17]	Studies SDN support for IEEE 802.11 and C-V2X in V2X communication	Proposes SDN framework to manage both standards	✓	✓	✓	×
[18]	Proposes hybrid DSRC and C-V2X routing in cluster-based networks	Optimizes communication for dense vehicular networks	✓	✓	✓	×
[19]	Focuses on improving handover management in 5G SDN-based V2X communication	Reduces handover latency using SDN, especially with X2 interface	✓	✓	✓	✓

2 Related Work

By focusing on work aimed at integrating SDN technology into a vehicular environment, we have been able to identify three main types of study:

- Focus on network robustness: [11], [12], [13] mainly focus on enhancing the robustness of the network through redundancy and fault-tolerant architectures. Their core lies in ensuring the system's reliability in the event of a fault or dangerous scenario. These projects involve anticipating breakdowns, white zones and the management of critical services. However, most of these studies aim at static or predictable network environments and fail to adequately address the needs of dynamic mobile networks;
- Focus on performance optimization: [14], [15], [16] propose various SDN-based network optimization methods (for example base station selection, resource reservation), with a particular focus on resource management and security-critical applications. While these studies propose effective performance optimization frameworks, they mainly focus on specific technologies or some static features of SDN and fail to cover the dynamic management of heterogeneous networks comprehensively;
- Focus on multi-radio acess technologies management: [17], [18], [19] have significantly addressed the challenges of V2X communications, especially in terms of multi-communication technology integration and handover management. However, these methods still need a complete solution for real-time adaptive interface selection, which is critical for communication stability in high-speed mobile scenarios.

This brief comparison, supplemented by a more detailed analysis of the environment in Table 1, highlights the shortcomings of the work proposed to date. The main objectives of this article are 1) to propose an SDN-based solution that takes advantage of the potential of this technology, and 2) to offer real-time management of network interfaces by considering different access networks and by being able to switch dynamically between different interfaces. To the authors knowledge, no work has yet been done to address these various elements. Consequently, the aim of this article is to fill this gap by proposing a solution adapted to the constraints of vehicular networks.

3 Considered Network Architecture

Our aim in this work is to design an architecture managing multiple radio access technologies (cellular, WiFi, Longe Range, and potentially other technologies) and the associated access points and interfaces thanks to an SDN-based approach. As shown in Fig. 1, such an architecture will be composed of two main elements: the Control Center and the network. The system presented in this section, in contrast to conventional SDN solutions, aims to integrate lightweight components designed to be seamlessly integrated into the vehicle architecture.

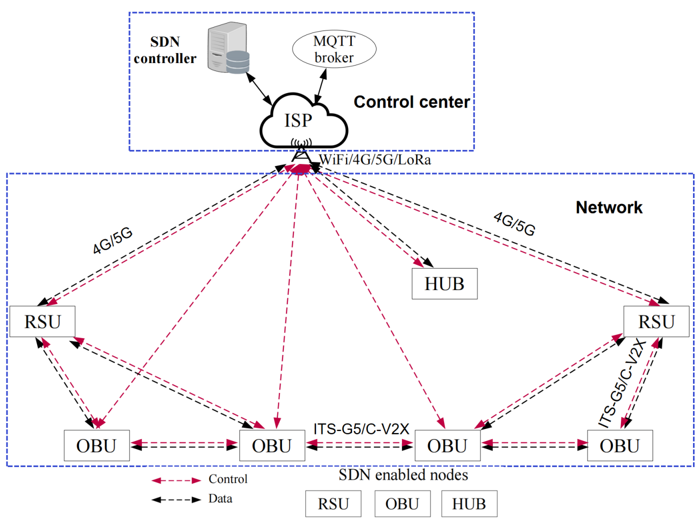

Fig. 1. SDN Architecture for Rail-Road Vehicles

3.1 Control Center

The control center is the core component that implements the SDN function. It consists of an SDN controller (described in Section 4) and a Message Queuing

Telemetry Transport (MQTT) agent [20]. The SDN controller centrally controls and dynamically manages network communication resources, optimizing routing, allocating spectrum, and managing wireless interface switching to ensure communication quality and connection reliability.

This controller connects with the entire network architecture through an Internet Service Provider (ISP). The ISP infrastructure mainly includes eNodeB/gNodeB (eNB/gNB) base stations and LoRaWAN Gateway, which provides 4G/5G and LoRa Radio Access Network (RAN) coverage for the system. The SDN controller communicates with network elements via the southbound interface in the SDN context. A MQTT protocol, which has a low bandwidth and efficient data transmission characteristics [20], is used to establish the southbound interface between the SDN controller and the SDN nodes.

3.2 Network

The network comprises multiple SDN-enabled nodes, such as Roadside Units (RSUs), Onboard Units (OBUs), and train stations (HUBs), managed by an SDN controller to form the overall communication architecture. These nodes are interconnected via various wireless technologies, including ITS-G5, C-V2X, 4G/5G, WiFi, and Long Range (LoRa), and support both single-hop and multi-hop communications, adapting to diverse traffic environments. Each node exchanges real-time data through ITS-G5 and C-V2X, and connects to the control center via 4G/5G and/or LoRa networks, enabling global monitoring and data transmission. In addition, LoRa can be used to exchange configuration information between the control center and SDN enabled nodes. This architecture forms a flexible multi-layer network supporting Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I), and Vehicle to Network (V2N) communications, ensuring efficient and stable services.

4 Lightweight SDN Controller Implementation

After this quick introduction of the main components of the architecture (control center, network), this section will focus on the mechanisms implemented at SDN controller level to enable flow rules management and at SDN-enabled nodes to enable data transmission and collaboration between them.

4.1 Internal architecture of SDN-based vehicular networks

As shown in Fig. 2, the SDN controller and the SDN nodes (*e.g.,* OBUs, RSUs, HUBs) are the two key components of this experimental system. The node ensures real-time communication with the SDN controller by continuously sending status updates (such as position, speed, and QoS). It can also receive flow rules from the controller and use the switching manager to switch communication interfaces. This collaboration effectively meets the high reliability and efficiency requirements of V2X communication. Therefore, components need to be implemented at both levels to reproduce an SDN-based operation.

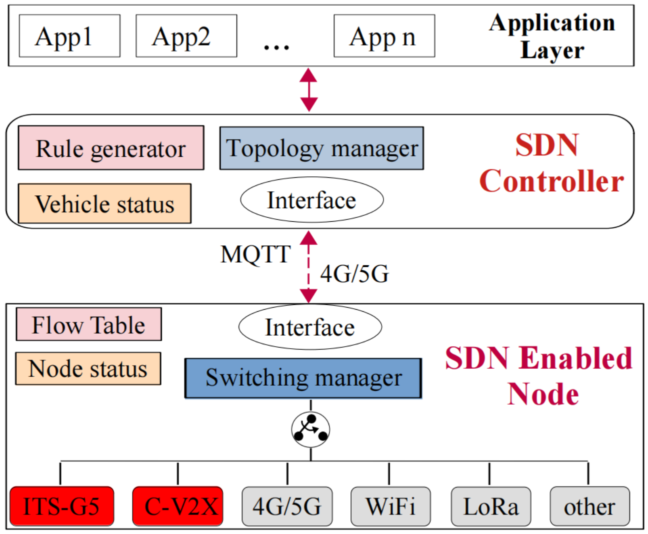

Fig. 2. Architecture of SDN-Based Vehicular Networks

4.2 Flow Rule Generator

At the SDN controller, an essential element is the flow rule generator. Indeed, the flow rule generator in the SDN controller defines its core mechanism for issuing flow rules (routing rules formulation for SDN nodes). The structure of a flow rule is mainly divided into three parts, as shown in Fig. 3:

1. Match - what characterises the flow?
2. Action - what do we do with this flow?
3. Statistics - What details are available about implementing this flow?

Implementation details of each of these elements is provided below. The "Num" is mainly used for an SDN-enabled node to distinguish between disabled flow tables of the same "Command Type" and "OBU-ID".

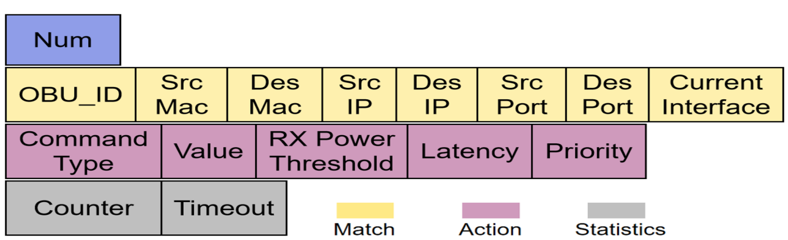

Fig. 3. Structure of a Flow Rule

Match The flow rule section is a prerequisite for executing actions. It defines the characteristics of data packets corresponding to the flow rule (e.g., Source/Destination MAC, IP, port, Interface, *etc.*), determining how to process the packet [10]. This is essential for enabling the SDN architecture to function.

Action This part defines execution thresholds and priorities based on communication data and perform the corresponding action when conditions are met. Each SDN enabled node receives "Command Type", "Value", "RX Power Threshold", and "Latency" from the SDN controller if the values in the Match field matches with their identification. The Command Type could be switching radio interfaces, or forwarding to next node. The SDN enabled node performs the action defined in the Value field based on comparison of the real time value of RX Power and/or Latency with their threshold. At this level, our idea was to offer a multi-layer solution, as opposed to traditional SDN architectures which do not necessarily integrate information from the low ISO layers.

That is why "Latency" and "RX Power Threshold", common parameters for ITS-G5 and C-V2X, are used as execution conditions in the flow rules (see Fig 3). If Latency or RX Power exceeds or is below the values set by the SDN controller (based on applications' requirement which could be redefined by a network administrator or based on Machine Learning - ML), the condition is marked as high priority, while the other is set as low priority. Defining these conditions based on ML approach is described below.

Thresholds are determined using Machine Learning Models (high priority) and residual control (low priority). A linear regression model predicts Latency (L) or Power (P) based on features like Received Signal Strength Indicator (RSSI), Channel Busy Rate (CBR), Packets Per Second (PPS), Channel Busy Percentage (CBP) and payload. With the near-linear relationship between features and target variables, linear regression is chosen for its simplicity, continuous variable efficiency, and suitability for real-time vehicular communications.

$$\hat{y} = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + b \tag{1}$$

where $\hat{y}$ is the predicted value, which could be latency $\hat{L}$ or power $\hat{P}$; $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ represents the input feature vector which is depending on the current QoS; $w_1, w_2, \ldots, w_n$ are the learned weights, and b is the bias term.

The model is trained by minimizing the Mean Squared Error (MSE) to optimize the weights $\mathbf{w}$ and bias b:

$$L(\mathbf{w}, b) = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \tag{2}$$

where N is the number of samples, $\hat{y}_i$ is the prediction for the i-th sample, and y_i is the true value.

The model predicts latency or power thresholds based on input characteristics and uses these predictions as real-time system criteria. By minimizing the error between predicted and actual values during training, it accurately captures network changes and adjusts thresholds dynamically in real-time.

When the high priority threshold is triggered, the low priority threshold is dynamically adjusted based on the residual between the high priority's predicted and actual values. Let $V = \{L, P\}$ represent the variable for either latency (L) or power (P). The residual for V is calculated as:

$$e_V = |V_{\text{actual}} - \hat{V}| \tag{3}$$

The low priority threshold is adjusted as:

$$V_{\text{adjusted}} = \hat{V} - k_V \cdot e_V \tag{4}$$

Here, k_V is the adjustment factor for V, where V_{actual} represents the actual latency or power, and $\hat{V}$ is the predicted threshold from the machine learning model. The residual e_V represents the deviation between expected and actual values, with the low priority threshold adjusted by reducing $\hat{V}$ in proportion to the residual. The adjustment factor k_V determines the impact of this residual.

By calculating the residual for high priority conditions, the system adjusts the low priority threshold based on actual network performance, ensuring coordinated handling of high and low priorities under varying network conditions.

Statistics The Statistic part includes Timeout and Counter, as depicted in Fig 3, elements relating to the communication flow. Timeout represents the duration of the flow rules remaining active before expiring, while Counter tracks how often the user device applies the flow rules. These are the usual components of SDN architectures, but in theory they are based on an arbitrary definition of these elements. The idea here is to link them to the mobile nature of the nodes and the availability of the communication networks. In another words, a node need to be able to switch its radio interfaces while within the coverage range of the current active radio interface to maintain connectivity. Thus, the timeout varies in relation to the speed, direction and position of the nodes (vehicles). Consequently, the algorithm for the Timeout mechanism is defined as follows:

If the direction is towards the boundary (D=1):

$$time\_to\_boundary = \frac{C - P}{S} \tag{5}$$

If the direction is away from the boundary (D=-1):

$$time\_to\_boundary = \frac{P}{S} \tag{6}$$

Here, S represents the device speed (in $meters/second$), P is the device's current position relative to the boundary, D indicates movement direction (1 for towards the boundary, -1 away from the boundary), and C is the device's maximum coverage range (in $meters$).

4.3 Switching Manager

As shown in Fig. 2, this component is part of the SDN enabled node, designed to connect with various network technologies such as ITS-G5, C-V2X, 4G/5G, WiFi, and LoRa. It is therefore a component that manages interfaces and is integrated at node level. Its purpose is to enable real-time switching. An implementation of such a switching manager based on the use of Software-Defined Radio (SDR) device is presented in section 5.

4.4 Dynamic Adjustment of Data Sending Frequency

To prevent the SDN controller from making incorrect decisions, SDN enabled nodes should dynamically adjust their data transmission frequency based on current conditions. In the flow rules, high latency, low power, and short timeout values signal the need to increase the transmission frequency to ensure accurate decisions by the SDN controller. The specific formula is as follows:

Let $V = \{\text{latency}, \text{power}, \text{timeout}\}$ represent the variable for latency, power, or timeout. The weight for each parameter is calculated as:

$$V_{\text{weight}} = \begin{cases} \frac{\text{latency} - L_{\min}}{L_{\max} - L_{\min}} & \text{for latency,} \\ \\ \frac{V_{\max} - V}{V_{\max} - V_{\min}} & \text{for power and timeout.} \end{cases} \tag{7}$$

$$\begin{aligned} &combined\_weight = w_L \times latency\_weight + w_T \\ &\times timeout\_weight + w_P \times power\_weight \end{aligned} \tag{8}$$

$$\begin{aligned} adjusted\_frequency = f_{\min} + (1 - combined\_weight) \\ \times (f_{\max} - f_{\min}) \end{aligned} \tag{9}$$

Where most of the variables in Eq. 7, 8, and 9 are self explanatory. The weighted coefficients (w_L, w_T and w_P) adjust the importance of each factor, allowing the algorithm to compute results for real-time applications.

The various components identified as being necessary for the operation of our SDN architecture have been defined in this section. In the next section, we demonstrate the deployment of SDN controller.

5 First deployment of the SDN Rail-Road Controller

In this section, we present the environment we have considered for an initial deployment and evaluation of the SDN controller we have proposed. For this demonstration, we have considered two OBUs to form a two-hop network infrastructure through two wireless technologies, our custom SDN controller, southbound interface through MQTT, and an ISP based on 4G/5G RAN. The detailed network components and their implementation is described below.

5.1 Network components selection and implementation

A simple schematic of deployment components is shown in Fig. 4.

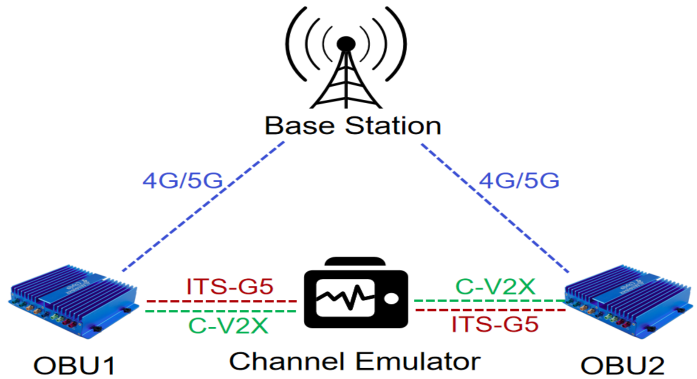

Fig. 4. Network and Channel Emulator

Technical choices for implementing the architecture Cellular networks are widely used in vehicle communication systems for their high-speed, low-latency connections, essential for real-time data transmission [21]. For this reason, in this first experiment, a 4G/5G RAN is considered to enable communication between SDN controller and the SDN enabled nodes. The equipment (OBUs) are therefore connected to a base station via cellular communication. It is assumed here that it is capable of guaranteeing relatively reliable communications between the SDN controller and the SDN-enabled nodes (OBUs). Network selection takes place at the level of direct communications between OBUs. At this link, two radio access technologies are available (ITS-G5/C-V2X) and the system will aim to determine in real time which is the best technology. The variation in performance of these communication links is reproduced using a channel emulator. To sum up, our objective is to evaluate an initial real-time management system for multiple radio access interfaces.

Base Station and User Equipment emulation In this study, the base station consists of an eNB, a key component of the 4G LTE network, responsible for wireless communication with User Equipment (UE) and forwarding data to the core network. The eNB is implemented using an open-source RAN platform as presented in [22]. It uses a SDR platform based on a general purpose processor (Linux based workstation) and USRP B210 SDR board for flexible signal transmission. The Evolved Packet Core (EPC) handles packet routing, connection control, and network interface management, which is deployed on the same workstation as the eNB. All functions of the eNB and EPC are implemented using srsRAN 4G open-source RAN software [23]. The SDN enabled nodes act as the User Equipment in the 4G/5G terminology where each node is equipped with a SIM card module. With each component deployed and configured, we have a system that enables the implementation of an end-to-end 4G network and guarantees the communication between the control center (SDN controller) and the network elements (SDN enabled nodes).

Channel Emulator Deployment The channel emulator in this study, built with GNU Radio, includes USRP sources and sinks, a dynamic channel model, and a frequency analysis module. The USRP components simulate signal transmission and reception, while the dynamic channel model replicates real-world effects like multipath propagation, fading, Doppler, and noise. As an initial test, an Extended Vehicular A (EVA) [5] channel model was designed with GNU Radio. This setup enables accurate V2X communication performance testing under various channel conditions in a controlled lab environment.

5.2 Global view of the considered platform

Fig. 5 shows the platform that integrates the main components of the considered system and that was used for evaluating our new SDN controller:

- A workstation called SDN Controller Terminal in Fig. 5 that integrated both 1) the SDN controller that we developed, based on the components described in Section 3; 2) the channel emulator, required to reproduce varying channel conditions; and 3) the components required to run both 4G access and core network (srsRAN and Open5GS [24]);
- A laptop called SDN Enabled Nodes Terminal in Fig. 5 is used to access and configure the Linux based OBU boards that represent the SDN-enabled nodes, through serial/Ethernet interface. In this setup, an Ethernet interface is used to log-in to the OBU boards. The laptop is also used to display output results from each OBU board using separate terminal window;
- OBU-1 and OBU-2 represent the two SDN-enabled nodes used to form the network infrastructure, as shown in Fig. 4. They are based on Cohda Wireless MK6 OBU boards [25]. While ITS-G5 and C-V2X interfaces of the OBU are used to form the network between the two OBUs, cellular interface of the OBU is used to send vehicle status information to the SDN controller and receive flow rules from the SDN controller. A Global Navigation Satellite System (GNSS) antenna is also used to obtain vehicle location information, and for time synchronization of the two OBU boards.
- USRPs are used in eNB and channel emulator, and their main purpose is signal reception and transmission and Downlink(DL) and Uplink(UL) communication. Since the two OBUs are connected to a single ISP (in this case one eNB), signal splitters (shown as SPs in Fig. 5), is used to divide one input signal to multiple output signals, allowing the same signal coming from the eNB to be used by multiple devices (OBUs). Indeed, the signal power after the splitter can be affected slightly (around 3 dB by the SPs used in our

demonstrator). Similarly, other two splitters are used between the channel emulator and the two OBUs. The power degradation due to the splitter (*i.e.,* 3 dB value) is taken into account in our experimentation.

– The Switch is able to connect OBU-1 and OBU-2 to the laptop described above so that it helps to avoid using multiple laptops for each OBU.

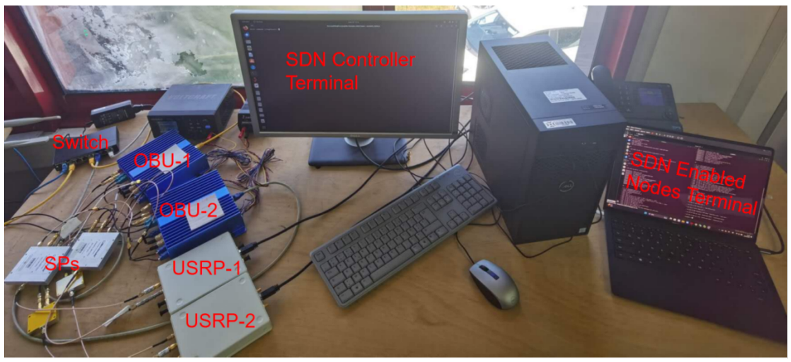

Fig. 5. General framework of the experimental platform

6 Time-consuming evaluation

Real-time and reliable data transfer is crucial for traffic safety and road efficiency in vehicular communication systems. As noted in [21], even slight delays can significantly impact system responsiveness and overall network performance. In this dynamic environment, both the management of in-vehicle communications and time-sensitive issues must be addressed. High vehicle speeds can cause delays, leading to missed critical decision windows and affecting performance. That is why this first experiment aimed at demonstrating that our SDN controller could guarantee real-time management of multiple radio access technologies. For the evaluation part, as with the earlier discussion on time impact in dynamic environments, we will focus on evaluating time consumption within this framework.

6.1 Defining Evaluation Metrics and Analyzing Results

This study will explore the time the SDN controller takes to send flow rules and the switching manager consumes per switch.

Testing 1 - Time taken to send flow rules The test measured the time taken to send flow rules in the EVA scenario at different Signal-to-Noise Ratio (SNR) levels (5 dB, 15 dB, 30 dB) and different vehicle speeds (10 km/h, 30 km/h, 50 km/h, and 70 km/h). The idea is to demonstrate that our solution, from the SDN controller perspective, remains functional under these variable conditions. The flow rule transmission times are presented in Table 2.

Table 2. Time taken to send flow rules

SNR Level	Speed Gap (km/h)	Time Taken (ms)	Distance (m) based on Speeds 30 km/h, 50 km/h, 70 km/h
5.0	10.0	24.596	0.205 / 0.342 / 0.478
5.0	30.0	24.272	0.202 / 0.337 / 0.472
5.0	50.0	24.582	0.205 / 0.341 / 0.478
5.0	70.0	23.819	0.198 / 0.331 / 0.463
15.0	10.0	24.164	0.201 / 0.336 / 0.470
15.0	30.0	27.427	0.228 / 0.380 / 0.532
15.0	50.0	24.098	0.200 / 0.335 / 0.469
15.0	70.0	23.691	0.197 / 0.329 / 0.461
30.0	10.0	23.857	0.199 / 0.332 / 0.465
30.0	30.0	22.970	0.191 / 0.320 / 0.449
30.0	50.0	22.958	0.191 / 0.320 / 0.448
30.0	70.0	25.227	0.210 / 0.351 / 0.491

As shown in Table 2 the time to send traffic rules ranges from 22.9 ms to 27.5 ms, depending on the SNR and speed differences. The distance covered at typical vehicle speeds is between 0.19 m and 0.54 m, a small range with minimal impact on receiving traffic rules within the communication coverage area.

Testing 2 - Time taken to switch interfaces This test focuses on the average and stability of the time taken to switch between ITS-G5 and C-V2X to reflect possible impacts. The idea is to demonstrate that our solution, from the SDN-enabled devices perspective, remains functional under these variable conditions (RX power, latency). In order to simulate the performance under different communication conditions, it is based on the adjustment of the emulator parameters in the eNB configuration file. And, it is based on the previously formulated ML algorithm that the internal value settings, parameter integration, and flow rule transmission are used to calculate the time consumed. The result for this testing is presented in Table 3.

As given in Table 3, it takes an average of 3.62 s to 3.73 s to switch between different technologies. During this period, a vehicle travels around 30 m to 73 m. This means that the switching must be completed within 75 m to avoid communication interruption, which is generally acceptable for low speed vehicles since ITS-G5 and C-V2X have a coverage range of more than 300 m [17]. However, at high speeds, reducing the switching time is critical for maintaining communication continuity and ensuring safety [21]. The switching time depends on the software and hardware implementation of the wireless technologies within

Table 3. Time taken to switch interfaces

Switching Time and Distance Data	**C-V2X to ITS-G5**	**ITS-G5 to C-V2X**
Maximum (s)	10.60	11.62
Minimum (s)	2.01	2.22
Mean (s) excluding Max&Min	3.62	3.73
Median (s) excluding Max&Min	3.22	3.25
Standard Deviation (s) excluding Max&Min	1.63	1.76
Distance based on different speed	30.17 / 50.28 / 70.39	31.08 / 51.81 / 72.53

the MK6 OBU boards and implementation of our switching manager module. A future work focuses on optimization of this switching time. These preliminary assessments seem to show that our solution could be capable of guaranteeing the extremely short lead times required for critical services.

7 Conclusions and Future work

This study defined and implemented a new SDN controller for wireless vehicular networking in railway and road environments. The solution enhances connection stability by dynamically managing the switching between communication interfaces based on real-time conditions, supporting cooperative intelligent transportation systems. Initial evaluations suggest that our solution could be capable of guaranteeing extremely short timescales for managing multiple radio interfaces (in the order of milliseconds) in mobile SDN nodes, and could therefore represent a major step forward compared with traditional SDN solutions that mainly focused on either a wired network or stationary wireless nodes.

In future work, it is possible to further optimize the ML algorithms used in this study, particularly in predicting latency and RX power thresholds, to improve the speed and accuracy of interface switching decisions. The optimization process will explore more advanced models to reduce the computational overhead while ensuring real-time performance. Reducing the interface switching time is also critical to maintaining communication stability, especially in high-speed environments. Additionally, future efforts will explore emerging technologies like 6G to extend coverage and enhance performance in areas with limited infrastructure. Furthermore, applying our SDN controller in a multi-hop scenario, and strengthening the security of SDN controllers against network attacks and data breaches will be another priority.

Acknowledgments. The work presented was performed in the framework of the French FLEXMOVE project supported by the ADEME agency within the FRANCE 2030 program and the European PODS4RAIL project GA 101121853.

References

1. MAALOUL, S., ANISS, H., BERBINEAU, M. & MENDIBOURE, L. Evolution of Use Cases for Intelligent Transport Systems. *Cooperative Intelligent Transport Systems: Control And Management*. pp. 33 (2024)
2. Brahim, R., Collart-Dutilleul, S., Bon, P., Laharotte, P. & Faouzi, N. FERRO-MOBILE and Security for Low Moment of Traffic Level Crossing. *International Conference On Risks And Security Of Internet And Systems*. pp. 35-50 (2023)
3. Jodeau, J., Absi, N., Chevrier, R. & Feillet, D. The rail-road Dial-a-Ride problem. *European Journal Of Operational Research*. (2024)
4. ETSI Standard EN 302 663 V1.2.1, "Intelligent Transport Systems: Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band," 2013.
5. 3GPP Standard TR 36.300 v14.3.0, "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN), Overall description; Stage 2 (Release 14)," 2017.
6. Rammohan, A. & Others Revolutionizing Intelligent Transportation Systems with Cellular Vehicle-to-Everything (C-V2X) technology: Current trends, use cases, emerging technologies, standardization bodies, industry analytics and future directions. *Vehicular Communications*. pp. 100638 (2023)
7. S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Improving connectivity in vehicular ad hoc networks," *Computer Communications*, vol. 31, pp. 1653-1659, 2008.
8. Sylla, T., Mendiboure, L., Maaloul, S., Aniss, H., Chalouf, M. & Delbruel, S. Multi-connectivity for 5G networks and beyond: A survey. *Sensors*. **22**, 7591 (2022)
9. Sheikh, M., Halder, M., Kabir, S., Miah, M. & Khatun, S. SDN-Based approach to evaluate the best controller: Internal controller NOX and external controllers POX, ONOS, RYU. *Global Journal Of Computer Science And Technology*. **19**, 21-32 (2019)
10. N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling innovation in campus networks,” *Computer Communication Review*, vol. 38, pp. 69–74, 2008.
11. A. Boubakri and S. Gammar, "A New Architecture of Autonomous Vehicles: Redundant Architecture to Improve Operational Safety," *International Journal of Robotics and Control Systems*, vol. 1, pp. 355-368, 2021.
12. S. B. Weinstein, “SDN control in a combined sensor, information retrieval, and communications network for dangerous environments,” *Proc. IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems*, 2017.
13. S. Gyawali, S. Xu, Y. Qian and R. Q. Hu, "Challenges and Solutions for Cellular Based V2X Communications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 222-255, 2021.
14. H. Alshaer and H. Haas, "SDN-enabled Li-Fi/Wi-Fi wireless medium access technologies integration framework," *IEEE Conference on Standards for Communications and Networking*, 2016.
15. J. Huang, H. Cui and C. Chen, "Cluster-Based Radio Resource Management in Dynamic Vehicular Networks," *IEEE Access*, vol. 10, pp. 43562-43570, 2022.
16. K. Smida, H. Tounsi, M. Frikha, and Y. -Q. Song, “Efficient SDN controller for safety applications in SDN-based vehicular networks: POX, Floodlight, ONOS or OpenDaylight?,” *Proc. IEEE Eighth International Conference on Communications and Networking (ComNet)*, 2020.

17. G. Nurkahfi, A. Triwinarko, B. Prawara, N. Armi, T. Juhana, N. Syambas, E. Mulyana, E. Dogheche, and I. Dayoub, "On SDN to support the IEEE 802.11 and C-V2X based vehicular communications," *IEEE Access*, 2023.
18. Y. Zhang and J. Zhang, "Design and Optimization of Cluster-Based DSRC and C-V2X Hybrid Routing," *Applied Sciences*, vol. 12, no. 13, p. 6782, 2022.
19. H. Rizvi and J. Akram, "Handover Management in 5G Software Defined Network Based V2X Communication," *Proceedings of the International Conference on Open Source Systems and Technologies (ICOSST)*, 2018.
20. S. Manandhar, "MQTT based communication in IoT," M.Sc. thesis, Tampere University of Technology, 2017.
21. Z. H. Khattak, J. Rios-Torres, and M. Fontaine, "Impact of communications delay on safety and stability of connected and automated vehicle platoons: Empirical evidence from experimental data," *IEEE Access*, 2023.
22. D. M. Molla, H. Badis, L. George and M. Berbineau, "Coverage Extension of Software Defined Radio Platforms for 3GPP 4G/5G Radio Access Networks," 2021 *13th IFIP Wireless and Mobile Networking Conference (WMNC)*, Montreal, QC, Canada, 2021, pp. 55-62, doi: 10.23919/WMNC53478.2021.9619041.
23. srsRAN Project: Open Source RAN [Online], Available: https://www.srslte.com/ Accessed on 10/09/2024.
24. Mamushiane, L., Lysko, A., Kobo, H. & Mwangama, J. Deploying a stable 5G SA testbed using srsRAN and Open5GS: UE integration and troubleshooting towards network slicing. *2023 International Conference On Artificial Intelligence, Big Data, Computing And Data Communication Systems (icABCD)*. pp. 1-10 (2023)
25. Cohda Wireless, "MK6", [Online]. Available: https://www.cohdawireless.com/ solutions/mk6/, Accessed on 17/07/2024.