Edge-V: Enabling Vehicular Edge Intelligence in Unlicensed Spectrum Bands

(Article begins on next page)

Edge-V: Enabling Vehicular Edge Intelligence in Unlicensed Spectrum Bands

Francesco Raviglione*, Claudio Casetti*, Francesco Restuccia†
*Politecnico di Torino †Northeastern University
E-mail: {francesco.raviglione, claudio.casetti}@polito.it {f.restuccia}@northeastern.edu

***Abstract*—Cutting-edge advances in wireless networking will soon enable a new generation of safer, smarter, and more autonomous vehicles. These vehicles will rely on real-time execution of complex Deep Learning (DL) tasks as well as high-speed multimedia streaming between road users for navigation purposes. Relying entirely on cellular networks (i) puts an unnecessary burden on an already overcrowded and expensive licensed spectrum; (ii) increases the latency of edge-offloaded tasks to intolerable levels for vehicular applications. Alongside the usage of a proper network infrastructure, vehicles will need to support on-board and offloaded cooperative intelligence. On this basis, we propose `Edge-V`, the first framework enabling practical vehicular edge intelligence and high-speed vehicular connectivity, using only unlicensed spectrum bands. Through a DSRC link, `Edge-V` acquires real-time localized knowledge, and coordinates the use of point-to-point millimeter Wave (mmWave) technologies to deliver high-bandwidth connectivity between vehicles. `Edge-V` also foresees smart offloading if on-board computing resources are insufficient. We prototype and evaluate `Edge-V` in a real-world laboratory testbed, showing its advantages with respect to cellular and cloud-based approaches.**

***Index Terms*—Vehicular Edge Computing, Vehicular Edge Intelligence, VEI, mmWave, Vehicular Networks**

I. Introduction

The automotive field is experiencing, in recent years, an impressive technological evolution, involving both communication technologies and on-board capabilities. High-bandwidth Vehicle-to-Vehicle (V2V) connectivity will enable a darth of use cases, leveraging also on-board sensors, such as LIDAR, which, however, can generate Terabytes per hour of data [1]. In addition, real-time analysis of sensor data requires complex Deep Learning (DL) algorithms, such as image segmentation, to detect the road. With the aim of enabling such use cases, it may become pivotal to deploy the intelligence at the edge of vehicular networks, giving birth to the concept of Vehicular Edge Intelligence (VEI). Furthermore, these use cases often require the execution of computationally expensive tasks, with strict latency requirements and the need for high throughput links. To this aim, task offloading can be deployed as a technique to make them practically feasible in vehicular networking scenarios. 5G connectivity is often proposed as a solution to tackle the low latency and high throughout requirements of Vehicle-to-Everything (V2X) applications and task offloading. However, sharing sensor data and offloading tasks with 5G would put enormous stress on an already overloaded licensed spectrum band. We thus propose `Edge-V`, the first open framework combining different technologies (Dedicated Short-Range Communications, mmWave and standard Wi-Fi) to enable VEI, on-board AI/ML and task offloading by leveraging only unlicensed spectrum bands. Our approach (i) reduces the usage of expensive 5G spectrum, (ii) reduces task latency through V2V cooperation, and (iii) may prove essential where 5G is limited or absent [2]. We show in Section V that cloud-based offloading incurs up to 59% more latency than our VEI approach. Existing vehicular technologies such as Cellular-V2X Mode 4 and IEEE 802.11p cannot guarantee VEI requirements, as their peak data rates are typically below 30 Mbit/s [3]. Although mmWave vehicular networking has been proposed [3], [4], it suffers from relatively high path loss and loss of connectivity due to blockages [5]. Thus, it is critical to combine the action of different mmWave and non-mmWave technologies to concretely realize the much-needed VEI. `Edge-V` has been designed to enable high-bandwidth, reliable and effective VEI while operating in unlicensed spectrum only. Starting from its architecture, a full-fledged prototype based on open source software and customizable hardware has been developed, supporting mmWave, IEEE 802.11p and IEEE 802.11ac, and extensively tested in a laboratory environment to showcase its advantages with respect to a baseline cloud-only approach.

II. Review of VEI, mmWave and task offloading

While V2X has been extensively researched in the past years, with the advent of access technologies such as IEEE 802.11p and C-V2X, VEI is still in its infancy, given the rapid development of DL-based algorithms specifically designed for vehicular applications [6]. The key protocols for V2X data exchange are IEEE 802.11p – its evolution, IEEE 802.11bd, is currently being standardized – and C-V2X being the application of 3GPP cellular standards to V2V and Vehicle-to-Infrastructure (V2I) networks. Recent research has investigated the application of mmWave technologies to vehicular networks, focusing also on IEEE 802.11ad [7] which works in the 60 GHz spectrum. Molina-Galan *et al.* [4] propose to decouple the mmWave data and control plane in V2X scenarios, and suggest leveraging sub-6 GHz C-V2X Mode 4 as control plane to schedule the data transmission over mmWave. Their proposal is evaluated through simulations. Raviglione *et al.* [7] presented instead one of the first field-test campaigns with mmWave applied to V2I communications, showing how it can provide very low delays and high throughput up to more than 100 m in Line-of-Sight conditions. However, they do not consider a VEI scenario and focus on the wireless performance only. The closest to our work is [3], where Li *et al.* propose a combination of SDN and mmWave links, with backhauling based on IEEE 802.11ad and C-V2X, and DSRC to carry control plane signals. `Edge-V` focuses instead on unlicensed spectrum technologies, and does not involve C-V2X communications.

A significant amount of work, e.g., [2], [8] has focused on developing algorithms for offloading tasks in the vehicular edge. For example, Tang *et al.* [9] proposed a cache enabled task offloading system, focusing on offloading tasks to nearby Road Side Units or base stations. However, these proposals are evaluated only through simulations, without considering the limitations of real hardware and the design of a full-fledged framework. Instead, our goal is to provide a framework to test these and other approaches on the field, with a strong focus on real hardware and on the combination of real-world wireless technologies.

III. Framework Description

Figure 1 shows a high-level overview of `Edge-V`, which combines computation and communication in unlicensed spectrum to deliver V2X and VEI capabilities to existing vehicular networks.

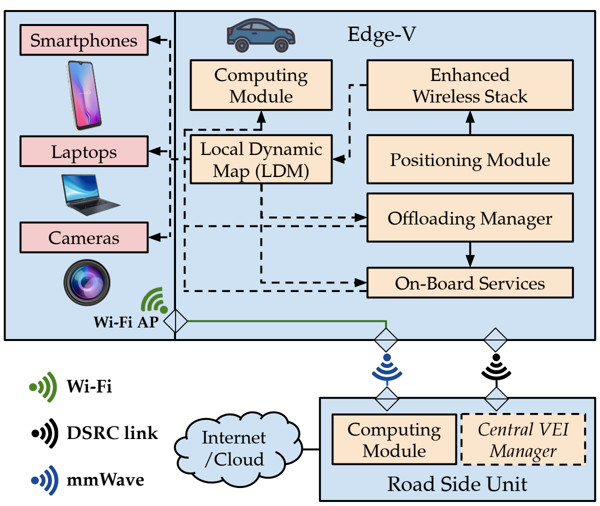

Fig. 1. High-level overview of the `Edge-V` framework.

A. `Edge-V`: Modules and Interfaces

The main components of `Edge-V` are summarized below. As can be seen, `Edge-V` is designed to be deployed, with different modules, on both connected vehicles and infrastructure nodes (i.e., Road-Side Units – RSUs –, which can be exploited for the connection to a MEC server or cloud data center).

- *Radio Interfaces.* Each vehicle is equipped with three radio interfaces, two of which are external (DSRC and directional mmWave), and one of which is internal to the vehicle. The Wi-Fi Access Point (AP) interface is bridged with the mmWave one (the green line depicted in Figure 1), to provide the on-board devices with access to the Internet, through the RSU, and let them seamlessly communicate between each other and between different vehicles equipped with mmWave. These devices may include cameras on board of different vehicles, enabling use cases such as See-Through. Furthermore, the presence of the RSU enables a data exchange between the on-board devices through the RSU itself, realizing, if needed, a mmWave V2I2V (Vehicle-to-Infrastructure-to-Vehicle) communication. This approach enables the exchange of data between vehicles even in case of noticeable Non-Line-of-Sight (NLOS) blockage or distances higher than a few hundreds of meters.
- *Computing Module.* Each vehicle and infrastructure node (i.e., RSU) has computing resources in terms of CPU, GPU and RAM, which can be used to execute tasks and on-board services locally, or by other nodes to offload their tasks.
- *Local Dynamic Map.* `Edge-V` includes a standard-compliant enhanced Local Dynamic Map (LDM) inside each vehicle, which can be leveraged to optimize decisions such as where to offload data, or to determine dangerous road conditions. The LDM includes, among others, computational load and channel load metrics for each vehicle stored in the map, pivotal to enable VEI. It provides enhanced awareness to the Offloading Manager, and it is populated through standard-compliant vehicular messages, exchanged through the DSRC link, both from other vehicles through multi-hop or from RSUs.
- *Enhanced Wireless Stack.* Each vehicle includes a standard-compliant Intelligent Transport Systems (ITS) stack, which is enhanced to include an optional container in periodic messages, e.g., Cooperative Awareness Messages (CAMs) in Europe, Basic Safety Messages in the US, for the exchange of channel and node load information. The Enhanced ITS messages are broadcasted through the DSRC link.
- *Positioning Module.* Each vehicle is equipped with an embedded or external GNSS receiver, which provides Position, Velocity and Time (PVT) data to the ITS stack for the generation of standard-compliant messages.
- *Offloading Manager.* The Offloading Manager (OM) is a key component selecting which are the best vehicles to perform task offloading and to communicate with, and decides which is the best link to use. The decision hinges upon on the information available in the LDM, and can be based on AI-based algorithms or mathematical optimization. `Edge-V` does not define a specific algorithm, and lets the user flexibly implement any of these approaches, depending on the target deployment scenario. This component is also used to manage the local computing resources and to determine the best route towards any destination in case multiple mmWave hops are required.
- *On-Board Services.* They are the core V2X services running on the On-Board Unit. They may include See-Through, collision avoidance, object detection through task offloading and many others. The V2X services retrieve useful data either from the ITS stack or from the high capacity mmWave link, and can make use of the on-board computing resources.

They can also directly retrieve kinematic, dynamic, node and channel load and network data from the LDM.

- *Road Side Unit (RSU)*. `Edge-V` uses RSUs for connection to the broader Internet. This enables further offloading of DL-based tasks to the cloud in case no vehicles have enough on-board resources to reach the desired results. Optionally, a *Central VEI Manager* may be deployed on the RSU to centrally manage groups of vehicles in a centralized VEI approach. `Edge-V` is designed, however, to work independently of a centralized unit.

B. `Edge-V`: A Walk-Through

For the sake of clarity, we provide a walk-through of the main operations performed by `Edge-V` in a real-world case of low-latency DL task offloading for VEI. In our example, DL tasks are generated through multimedia sensors and captured through the Wi-Fi interface (**Step 1**). On the other hand, `Edge-V` is receiving enhanced ITS messages which are used to populate the LDM (**Step 2**). As DL tasks are being generated, `Edge-V` checks the available on-board resources thanks to the Offloading Manager, which directly gathers this information from the Computing Module (**Step 3**). If there are not enough resources on the vehicle (henceforth the *ego vehicle*), the Offloading Manager will select the best *remote vehicles* to offload the task to. To compute the needed information, the Offloading Manager will leverage the LDM, solve an optimization problem which has been previously programmed, and provide the result to the On-Board Service (**Step 4**). The latter can then offload the tasks thanks to the mmWave links (**Step 5**).

IV. Proof-of-Concept

A full-fledged Proof-of-Concept (POC) of `Edge-V` has been developed with open source software and customizable hardware, with the aim of (i) proposing for the first time a VEI testbed based on unlicensed spectrum technologies to allow further study by other researchers, (ii) as well as evaluating the performance of `Edge-V` in a number of significant use cases. Our POC is depicted in Figure 2.

We prepared 1 RSU and 3 On-Board Units (OBUs) as part of our prototype. As access technologies, we selected three IEEE-based standards, working on unlicensed bands, i.e. (i) IEEE 802.11p for the DSRC link, (ii) IEEE 802.11ad for the mmWave link, (iii) IEEE 802.11ac for the internal Wi-Fi AP.

Concerning the hardware, we utilize customizable PC Engines APU2E4 hardware boards, equipped with a quad-core embedded CPU and 4 GB of RAM. As operating system, we use OpenWrt-V2X 21.02, a patched version of the OpenWrt embedded Linux distribution enabling IEEE 802.11p communication [10]. Each APU board is equipped with an (i) Atheros AR5B22 module for the IEEE 802.11p interface, (ii) a Compex WLE1216V5-23 Multi-User Multiple Input Multiple Output (MU-MIMO) 4x4 IEEE 802.11ac card. We connected a MikroTik wAP 60G device to each APU board (both OBUs and RSU), through one of the available Ethernet ports. The wAP 60G are IEEE 802.11ad-compliant devices based on

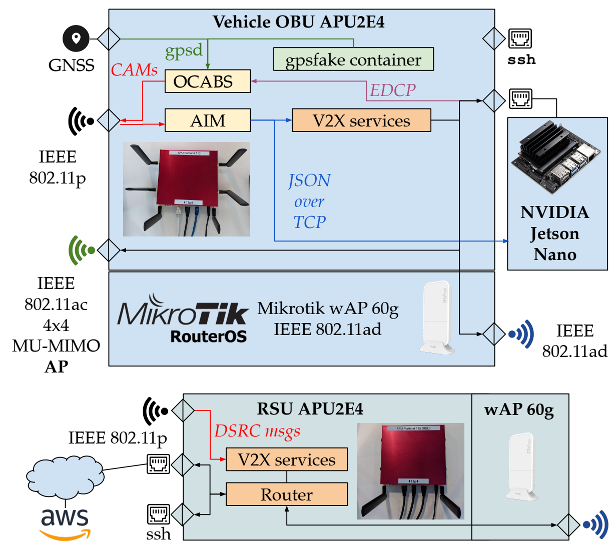

Fig. 2. The proof-of-concept prototype of `Edge-V`.

Qualcomm Atheros QCA6335 60 GHz chipsets, and equipped with 6x6 planar phased antenna arrays, covering an angular range of 60 degrees. Since the mmWave devices are still unable to establish direct peer-to-peer links, we deployed an additional MikroTik wAP 60Gx3 AP, to which all the devices in the testbed are connected. This way, the client devices appear as they are directly connected together. Finally, we equipped each APU board (except the RSU one) with an Nvidia Jetson Nano Development Kit for GPU computation.

As communication standard, we utilize the European ETSI ITS-G5. Thus, CAM messages are used to periodically broadcast, via IEEE 802.11p, dynamic vehicle information such as speed, acceleration, position and heading. Since there are no standardized messages carrying the node-load and channel-load information pivotal to VEI, we designed an additional optional container which can be inserted inside standard CAM messages[1]. This has been realized by upgrading the standard specifications and generating the related encoding and decoding functions using the *asn1c* tool. Moreover, we developed a new software module called *Open Cooperative Awareness (CA) Basic Service* (OCABS) to implement an ETSI CA Basic Service for the transmission of both standard and our enhanced CAMs[2]. OCABS needs to be fed with GNSS data, in order to properly encode and broadcast the CAMs. This data comes either from pre-recorded traces (which are then replayed in a container thanks to tools like `gpsfake`), or from a USB GNSS receiver, thanks to the default Linux GNSS daemon `gpsd`. Furthermore, we developed a novel *Automotive Integrated Map* (AIM)[3] to realize the `Edge-V` LDM and a UDP-based *Extra Device Communication Protocol* (EDCP)[2] to transfer with low latency CPU, GPU and RAM usage of the Nvidia Jetson Nano to OCABS. Each Nvidia board is

[1]https://github.com/francescoraves483/EnhancedCAMs-asn1
[2]https://github.com/francescoraves483/OCABS-project
[3]https://github.com/francescoraves483/AIM-AutomotiveIntegratedMap

running a service waiting for requests from OCABS. Each time a request is received, the load information is encoded and sent to the APU board thanks to EDCP.

V. Edge-V Evaluation

Thanks to our POC, we have evaluated our framework in a laboratory setting, with the aim of showcasing its advantages in real-world conditions, addressing two use cases.

The first use case generalizes to several automotive applications requiring a low-latency direct exchange of data, such as video streaming, See-Through, online gaming, and many more. To this end, two laptops are associated with the IEEE 802.11ac access point generated by the respective APU2 boards. We performed latency and throughput tests to evaluate the performance of our framework, through the POC hardware.

The second use case, instead, showcases a DL task offloading application enabled by Edge-V. In this use case, less safety critical tasks (but still with strict latency and throughput requirements) can be offloaded to other vehicles (or, if needed, to the infrastructure) that provide free resources.

Laboratory setting corresponds to nearly ideal conditions, with the devices in Line-of-Sight (LOS) and placed around 40 cm from each other. This allowed us to perform a baseline characterization of our framework.

A. Direct data exchange

This set of tests relied on the open source software available as part of the LTNT project[4], to reliably measure latency and throughput between two APU boards (i.e., OBUs) exchanging a flow of traffic. This allowed us to emulate the presence of multimedia and sensor data information, which needs to be exchanged between two vehicles.

Figure 3 shows the Cumulative Distribution Function (CDF) of throughput and Round Trip Time (RTT), measured between two APU boards, in a direct mmWave Vehicle-to-Vehicle communication (top plots), and between the same boards through the RSU, emulating a longer-range V2I2V scenario, in which the infrastructure node acts as message relay (bottom plots). The overall RTT always remains below 5 ms, with an average of around 1.4 ms for the direct communication case (V2V) and 3.7 ms for the relayed communication (V2I2V), demonstrating that mmWave is suitable for a very low-latency high-throughput scenario. Indeed, it reaches more than 500 Mbit/s when using TCP and UDP. This value, even though it could actually increase without the extra step through the mmWave AP and with more dedicated V2X devices, shows once more the advantages mmWave could bring to the next generation of automotive use cases.

B. Task offloading

The second use case is aimed at showcasing a DL task offloading application enabled by `Edge-V`. We developed an object detection *on-board service* based on the Microsoft Common Objects in Context (COCO) dataset. Each time, a vehicle can offload either to other vehicles or the cloud.

[4]https://github.com/francescoraves483/LTNT

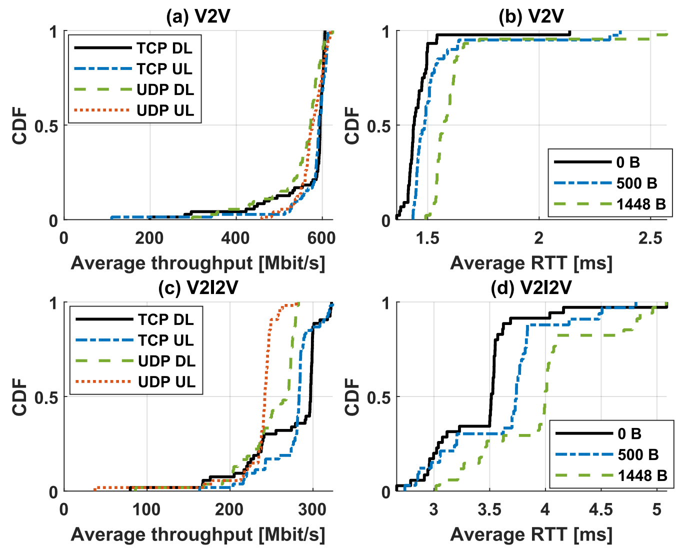

Fig. 3. CDFs of the mmWave latency and throughput during a one-day-long test. (a) V2V throughput (b) V2V latency (b) V2I2V throughput (through RSU) (d) V2I2V latency.

Three OBUs implement the vehicle edge, while an Amazon AWS Virtual Machine has been used to implement the cloud, reachable through our laboratory network. OBUs rely on a less accurate, though less computationally expensive, object detection model, i.e., *Faster R-CNN Large*, with MobileNet V3 backbone [11] and run on Jetson Nano boards. Instead, the cloud uses YOLOX-s [12], more computationally expensive but also more accurate.

We implement both an *Object Detection Offloading Manager*, running on one OBU and an *Object Detection Offloading Worker*, running on the other OBUs and on the cloud. The first includes a full COCO dataset, to emulate frames coming from an actual on-board camera, on which an object detection task is needed. We also deploy, as part of the OM, a simple offloading strategy which tries to minimize the latency. To this aim, the OM has been programmed to select the best task destinations based on the information available inside AIM, i.e., distance from target, RSSI of the mmWave channel to target, available resources on the target. The algorithm selects a set of promising vehicles, located at a distance lower than a given threshold (we selected 140 m, starting from the results presented in [7]), with an RSSI higher than a given threshold (i.e., -65 dBm [7]) and with enough available resources. It then offloads to the nearest one. If no vehicles are available for offloading, the task is offloaded to the cloud. Each Offloading Worker is then set to receive frames from the Offloading Manager, perform the inference, and then return the detection results in a JSON format to the sending node.

Figure 4 depicts the results of a test session on the whole COCO dataset. The frequency at which the different images are offloaded has been varied from one image every 1.6 s (i.e., 0.625 Hz) to 10 images per second. We then measured the average end-to-end latency (with 95% confidence intervals) and the mean average precision, both when offloading to other vehicles is enabled (our `Edge-V` approach), and when considering only the cloud as available option. With the aim of providing comparable results, we measured the time needed by the cloud to perform inference on the Nvidia Jetson Nano model and assigned that computing time, for each image, to each actual inference on the Nvidia boards, instead of considering the embedded board computing times. This is technically sound, as actual vehicles are expected to provide much better computation capabilities.

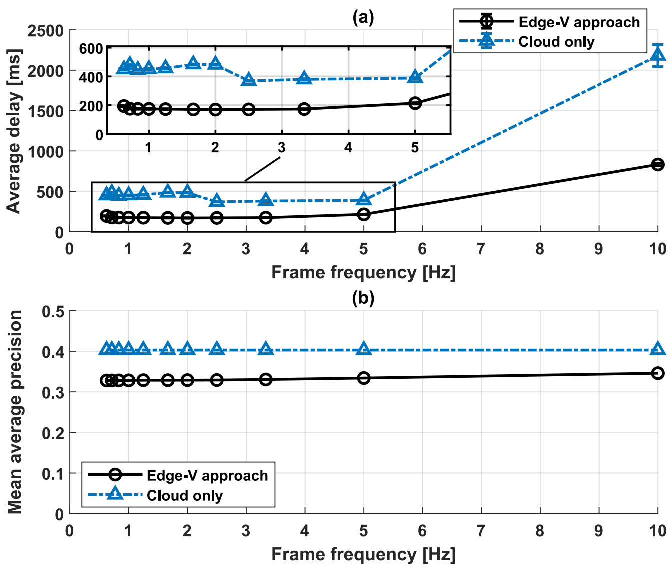

Fig. 4. (a) End-to-end processing and network latency and (b) Overall mean average precision accuracy of the object detection, as a function of the image generation frequency.

We notice that offloading to nearby vehicles helps to reduce the overall latency significantly, at the expense of a slight precision reduction. Up to 5 Hz, offloading to nearby vehicles can help to reduce the end-to-end latency of more than 150 ms, with up to 286 ms latency saving at 1.67 Hz (i.e., a 59% improvement). This comes at a reduction of the precision from 0.403 to around 0.328, which is much less impacting than the actual latency reduction. Furthermore, it is possible to notice a slight increase in latency in the vehicle edge, and a corresponding accuracy increase, at frequencies higher than 3 Hz. This is due to the fact that each OBU, in our testbed, can perform inference only on one frame at a time, due to the resources available in each Nvidia Jetson Nano. Therefore, when the frequency is high enough, the offloading manager may find that no vehicle is available for offloading, thus automatically sending that frame to the cloud.

VI. Conclusions and Future Work

This paper has presented a framework which leverages the combination of unlicensed spectrum technologies, together with an embedded customizable offloading module, to enable full-fledged Vehicular Edge Intelligence (VEI). We have designed the architecture of our framework, named `Edge-V`, and developed a POC testbed, based on open-source software and low-cost customizable hardware. Thanks to our POC and to extensive laboratory tests, we have showcased the advantages of `Edge-V`, proving how it can lead to almost 60% latency reduction in DL-task offloading, with respect to a cloud-only baseline. Future work will focus on the development and proposal of a system model for the OM.

Acknowledgment

This study was carried out within the MOST – Sustainable Mobility National Research Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1033 17/06/2022, CN00000023) and was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART"), as well as by the National Science Foundation (NSF) under Grant No. CNS-2134973. This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission and the NSF can be considered responsible for them.

References

[1] M. Drago, T. Zugno, M. Polese, M. Giordani, and M. Zorzi, "MilliCar: An ns-3 module for mmWave NR V2X networks," in *Proceedings of the 2020 Workshop on ns-3*, 2020, pp. 9–16.

[2] P. Dai, K. Hu, X. Wu, H. Xing, and Z. Yu, "Asynchronous Deep Reinforcement Learning for Data-Driven Task Offloading in MEC-Empowered Vehicular Networks," in *IEEE INFOCOM 2021*, 2021.

[3] Z. Li, T. Yu, R. Fukatsu, G. K. Tran, and K. Sakaguchi, "Towards Safe Automated Driving: Design of Software-Defined Dynamic MmWave V2X Networks and PoC Implementation," *IEEE Open Journal of Vehicular Technology*, vol. 2, pp. 78–93, 2021.

[4] A. Molina-Galan, B. Coll-Perales, and J. Gozalvez, "C-V2X Assisted mmWave V2V Scheduling," in *2019 IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, 2019, pp. 1–5.

[5] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated Access and Backhaul in 5G mmWave Networks: Potential and Challenges," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 62–68, 2020.

[6] H. Ye, L. Liang, G. Ye Li, J. Kim, L. Lu, and M. Wu, "Machine Learning for Vehicular Networks: Recent Advances and Application Examples," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 94–101, 2018.

[7] F. Raviglione, M. Malinverno, S. Feraco, G. Avino, C. Casetti, C. F. Chiasserini, N. Amati, and J. Widmer, "Experimental Assessment of IEEE 802.11-Based V2I Communications," in *18th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*. ACM, 2021, p. 33–40.

[8] B.-J. Qiu, C.-Y. Hsieh, J.-C. Chen, and F. Dressler, "DCOA: Double-Check Offloading Algorithm to Road-Side Unit and Vehicular Micro-Cloud in 5G Networks," in *GLOBECOM 2020*, 2020, pp. 1–6.

[9] C. Tang, C. Zhu, X. Wei, Q. Li, and J. J. P. C. Rodrigues, "Task Caching in Vehicular Edge Computing," in *IEEE INFOCOM Workshops 2021*, 2021, pp. 1–6.

[10] F. Raviglione, M. Malinverno, and C. Casetti, "Characterization and Performance Evaluation of IEEE 802.11p NICs," in *1st ACM MobiHoc Workshop on Technologies, MOdels, and Protocols for Cooperative Connected Cars*. ACM, 2019, p. 13–18.

[11] A. Howard, M. Sandler, G. Chu, L. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," *CoRR*, vol. abs/1905.02244, 2019. [Online]. Available: http://arxiv.org/abs/1905.02244

[12] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.