Resource Allocation for Vehicle-to-Vehicle Communications under Intermittent Cellular Coverage

vorgelegt von
M. Sc.
Taylan Şahin

an der Fakultät IV – Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr.-Ing. habil. Thomas Magedanz
Gutachter:	Prof. Dr.-Ing. habil. Adam Wolisz
Gutachter:	Prof. Dr.-Ing. habil. Falko Dressler
Gutachter:	Prof. Dr. Klaus Mößner
Gutachter:	Dr. Mate Boban

Tag der wissenschaftlichen Aussprache: 23. Mai 2023

Berlin 2023

To my mother and father, the ones whose “coverage” I am always in...

Acknowledgements

This thesis would not have been possible without a few very special people. First of all, I would like to sincerely thank my thesis advisor Prof. Dr. Adam Wolisz. He has supported this thesis from day one and provided me invaluable guidance and feedback with his deep experience in the field, with endless motivation. Next, I would like to thank Dr. Mate Boban, my supervisor at Huawei Munich. He offered me this thesis opportunity and always showed great support at every stage. His research skills and agility have always inspired me, even in my daily life. Following, I would like to thank Dr. Ramin Khalili from Huawei, who guided me in the world of artificial intelligence. Our lively and hands-on discussions have shaped this thesis a lot, thanks to his great skills and experience.

This thesis work was conducted at Huawei Technologies in Munich, Germany, for which I am grateful for their funding. During my six years at Huawei, first as a Master's student, then as a doctoral candidate, I had a chance to meet great people. I would first like to thank my former supervisors Josef Eichinger, Dr. Chan Zhou, and Assoc. Prof. Dr. Alexandros Kaloxylos, as well as Dr. Ömer Bulakçı and Dr. Onurcan İşcan. They never hesitated to provide their full support when I needed help. Without them, my professional career would simply not have been possible. I would like to acknowledge my peers Prajwal Keshavamurthy, Tapisha Soni, Ramya Panthangi, and Yunyan Chang for their efforts in our simulation development team, as well as supportive and friendly discussions we had. I would like to thank Dr. Serkan Ayaz, Dr. Mario Castañeda García, Dr. Daniel Medina, Dr. Apostolos Kousaridas, Dr. Panagiotis Spapis, Dr. Malte Schellmann, Dr. Samer Bazzi, and many other colleagues for the interesting technical and casual discussions we had. I was lucky to have very nice friends as doctoral mates: Anastasios Kakkavas, Dr. Marcin Iwanow, Dr. Marcin Pikus, Dr. Naveed Iqbal, Dr. Tobias Laas (special thanks to him for reviewing my journal publication and the thesis abstract), and Dr. Yilin Li.

Finally, I would like to express my deep gratitude to my family and my wife Dilara for their immense support. They are the ones to whom I go and who make me feel comfortable in my most difficult times, during this thesis journey, and always.

Munich, 2022
Taylan Şahin

Abstract

Vehicle-to-vehicle (V2V) communication is a key technology to enable safer, more efficient, and more comfortable road traffic. The stringent reliability and latency requirements of V2V messages necessitate efficient radio resource management given the scarce spectrum and the dynamic vehicular environment. Under cellular network coverage, the resource allocation can be centrally coordinated by a base station (BS), which can efficiently ensure collision-free transmissions. When out of coverage, vehicles resort to distributed mechanisms, which yet suffer from degraded communication quality due to the vehicles' limited local view.

In this thesis, we propose a novel approach for V2V communications in expected, delimited out-of-coverage areas (DOCAs), whereby a centralized scheduler pre-assigns resources to the vehicles via the BSs surrounding the area, before vehicles enter it. We first explore the feasibility of this approach by exploiting the road and data traffic information available in coverage to reserve and provision the resources. While the required number of resources does not grow prohibitively with increased reliability targets, the rate of successful V2V transmissions gets highly impacted by various factors such as vehicle mobility, thus necessitating efficient means to cope with uncertainties in DOCAs.

As a predictive method for resource allocation, we propose a vehicular reinforcement learning scheduler, VRLS, which is applicable to DOCAs that vary in vehicle density, mobility, wireless channel characteristics, and resource configurations. VRLS can significantly increase resource utilization efficiency by requiring fewer resources than state-of-the-art distributed scheduling solutions to support the same reliability targets. Nevertheless, considering that the performance of learning-based solutions may degrade upon parameter distributions much beyond their training environment, we propose a hybrid scheme that combines the centralized RL-based and the distributed sensing-based scheduling approaches. We show the performance benefits of such a solution under heavily congested road traffic due to an accident, as compared to either of the centralized or the distributed solutions.

Finally, we shift our focus to those areas under network coverage where vehicles suffer from rather short and unpredictable coverage interruptions to the BSs. We consider an extension of our RL-based approach for this problem. The proposed solution performs better than the state-of-the-art baseline in the cases of coverage losses, especially under high traffic load and lower frequency of scheduling updates, otherwise delivering similar performance.

Zusammenfassung

Die Kommunikation zwischen Fahrzeugen (Vehicle-to-Vehicle, V2V) ist eine Schlüsseltechnologie für einen sichereren, effizienteren und bequemeren Straßenverkehr. Die strengen Anforderungen an die Zuverlässigkeit und Latenz von V2V-Nachrichten erfordern angesichts des knappen Spektrums und der dynamischen Verkehrslage eine effiziente Verwaltung der Funkressourcen. Bei einer Mobilfunknetzabdeckung kann die Ressourcenallokation zentral von einer Basisstation (BS) koordiniert werden, die effizient kollisionsfreie Übertragungen gewährleisten kann. Außerhalb der Netzabdeckung müssen die Fahrzeuge auf verteilte Verfahren zurückgreifen, die jedoch aufgrund der eingeschränkten lokalen Sicht der Fahrzeuge eine schlechtere Kommunikationsqualität zur Folge haben.

In dieser Arbeit wird ein neuartiges Verfahren für die V2V-Kommunikation in erwarteten, abgegrenzten Gebieten ohne Netzabdeckung (delimited out-of-coverage areas, DOCAs) vorgeschlagen, bei dem ein zentraler Scheduler den Fahrzeugen über die umliegenden BSs Ressourcen zuweist, bevor die Fahrzeuge in das Gebiet einfahren. Es wird zunächst die Machbarkeit dieses Konzepts untersucht, indem die in der Abdeckung verfügbaren Straßen- und Datenverkehrsinformationen zur Reservierung und Bereitstellung der Ressourcen genutzt werden. Während die benötigte Anzahl an Ressourcen mit zunehmenden Zuverlässigkeitszielen nicht übermäßig ansteigt, wird die Rate erfolgreicher V2V-Übertragungen durch verschiedene Faktoren wie der Fahrzeugmobilität stark beeinflusst, sodass effiziente Wege, um mit den Unsicherheiten in den DOCAs zurechtzukommen, erforderlich sind.

Als prädiktive Methode für die Ressourcenallokation wird ein Vehicular Reinforcement Learning Scheduler (VRLS) vorgeschlagen, der auf DOCAs anwendbar ist, die in Bezug auf Fahrzeugdichte, Mobilität, Funkkanaleigenschaften und Ressourcenkonfigurationen variieren. Der VRLS kann die Effizienz der Ressourcennutzung erheblich steigern, weil er weniger Ressourcen benötigt als verteilte Scheduling-Lösungen nach dem Stand der Technik, um die gleichen Zuverlässigkeitsziele zu unterstützen. Allerdings ist zu bedenken, dass die Leistung von auf Lernen basierenden Lösungen bei Parameterverteilungen, die weit über ihre Trainingsumgebung hinausgehen, beeinträchtigt sein kann. Daher wird ein hybrides Verfahren vorgeschlagen, das die auf zentralisiertem verstärkendem Lernen basierenden und die auf verteiltem Sensing basierenden Scheduling-Konzepte kombiniert. Es werden die

Leistungsvorteile einer solchen Lösung bei stark überlastetem Straßenverkehr aufgrund eines Unfalls im Vergleich zu einer zentralisierten oder verteilten Lösung gezeigt.

Schließlich wird der Fokus auf die Bereiche mit Netzabdeckung verlagert, in denen Fahrzeuge unter eher kurzen und unvorhersehbaren Unterbrechungen der Netzabdeckung zu den BSs leiden. Der vorgeschlagene, auf verstärkendem Lernen basierende Ansatz, wird auf dieses Problem erweitert. Die vorgeschlagene Lösung schneidet bei Abdeckungsverlusten besser ab als der Stand der Technik, insbesondere bei hoher Netzauslastung und geringerer Häufigkeit von Scheduling-Updates, wobei ansonsten eine ähnliche Leistung erzielt wird.

Contents

Chapter 1

Introduction

1.1 Background and Motivation

Mobility is one of the pillars of human civilization: transportation of people and goods is essential for economic existence [1]. Across the centuries, the advances in industrialization and transportation systems have mutually benefited each other. On the other side, the safety of individuals has been being threatened by road transport, with a growing impact. Annually, road traffic accidents claim more than 1.3 million lives worldwide, making them the leading cause of death for children and young adults [2]. The most significant risk factors include human errors, such as speeding and distracted driving, besides unsafe vehicles and road infrastructure. In addition to safety, road traffic is also challenging our society and nature due to congestion that results in increased travel times and air pollution.

The ever-growing societal and economic impact of road traffic since the mid-twentieth century has led to a family of technologies named *intelligent transport systems (ITS)* [3]. ITS, and in particular cooperative-ITS (C-ITS), aims to provide innovative services for safer, more efficient, and smarter road transport [4]. Based on the ever-evolving needs and challenges of road transport, ITS applications cover a wide variety of use cases, ranging from collision avoidance systems to highly automated driving, from infotainment services to remote vehicle diagnosis, where *vehicular communication* is a key enabler technology [5].

Commonly referred to with the umbrella term *"vehicle-to-everything" (V2X)*, vehicular communications entail the following two-way connectivity between the vehicles and the entities around them, as illustrated in Figure 1.1:

- vehicle-to-vehicle (V2V): vehicles communicate with other vehicles.

- vehicle-to-infrastructure (V2I): vehicles communicate with the infrastructure around them, such as traffic signals or tolls – collectively referred to as roadside units (RSUs).

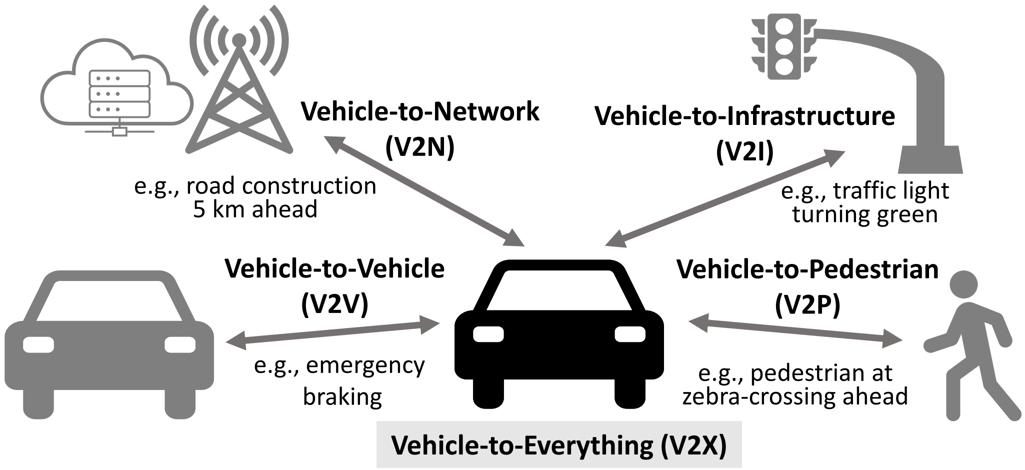

Vehicle-to-Network (V2N)
e.g., road construction 5 km ahead
Vehicle-to-Infrastructure (V2I)
e.g., traffic light turning green
Vehicle-to-Vehicle (V2V)
e.g., emergency braking
Vehicle-to-Pedestrian (V2P)
e.g., pedestrian at zebra-crossing ahead
Vehicle-to-Everything (V2X)

Fig. 1.1 Connectivity enabled by vehicle-to-everything (V2X) communications (based on [6]).

- vehicle-to-pedestrian (V2P): vehicles communicate with pedestrians, cyclists, drivers, or passengers, via the hand-held devices carried by them.
- vehicle-to-network (V2N): vehicles communicate with the mobile network, e.g., to access cloud services on the Internet.

V2X communications unlock awareness among the users of the road traffic, beyond the capability and range of humans or other technologies such as sensor-based systems (e.g., cameras and radars) [7]. By communicating with their surroundings, vehicles can "see" around corners, blind spots, or "through" other vehicles, which allows more time and information to warn drivers and take suitable action. When integrated with other systems, such as intelligent driving applications, V2X would allow vehicles to cooperate, and therefore avoid accidents, drive in a fuel-efficient way, and offer enjoyable journeys.

Currently, V2X communications are supported by two families of standardized radio access technologies (RATs):

1. WAVE/ITS-G5: Wireless Access in Vehicular Environments (WAVE) [8] and Dedicated Short Range Communications (DSRC) [9] in the United States, together with its European counterpart ITS-G5 [10]. It is the first family of standards that introduced a radio technology to support V2V as well as V2I communications, based on the IEEE 802.11p standard [11], which is a Wi-Fi-based technology.

2. Cellular V2X (C-V2X): C-V2X is based on the Fourth Generation Long Term Evolution (4G LTE) and the Fifth Generation New Radio (5G NR) wireless standards developed by the 3rd Generation Partnership Project (3GPP) [12]. C-V2X offers direct

V2V communications, as well as V2P, V2I, and V2N communications based on the mobile cellular radio technology.

Significance and Characteristics of V2V Communications

Among the V2X connectivity opportunities, V2V plays an especially important role in establishing road safety and efficiency. V2V communication takes place over the direct link between the vehicles in proximity using short-range communications, and can be established anytime and anywhere, in an ad hoc manner, without the need for vehicles to associate themselves with an access point, e.g., a base station (BS) in the case of cellular networks[1], to join the network. Utilizing such direct, peer-to-peer links for local traffic leverages the following potential gains [14], [15], [16]:

- Hop gain: A single transmission over the single hop between the vehicles would use radio resources more efficiently as compared to multiple transmissions required when relaying the data via an intermediate network node or nodes such as BSs. In addition, by eliminating additional processing or transmission delays at the relaying node, the time it takes to deliver information can be reduced.
- Proximity gain: Closer distance between the vehicles as compared to longer distances to the BS (or any network access point in general) results in more favorable channel conditions. With such links, V2V transmissions can make use of higher data rates at lower transmission power (or lower power consumption at the same data rate), and lower propagation delays.
- Reuse gain: Shorter distances between the vehicles require less transmit power to achieve good transmission quality, which in turn also minimizes the interference to the other links using the same resources. By spatially reusing the radio resources among the vehicles, overall spectral efficiency of the network can be increased.

With the above gains, V2V communication enables an efficient way to serve vehicular data traffic by offering power savings, higher data rates, lower delays, robustness to the absence or failure of network access points or infrastructure, and by offloading the links between the BSs and other users in the case of cellular networks.

V2V communication predominantly forms the basis for the most safety and traffic efficiency applications in ITS via *cooperative awareness*, that is, the knowledge of the presence

[1] Communication of vehicles with each other via the cellular network infrastructure is rather referred to as "V2N2V" communications in the literature [13].

and status of surrounding vehicles [17]. This is achieved by vehicles periodically broadcasting, i.e., regularly "announcing" their location, speed, bearing, etc. to their surroundings by transmitting frequent messages, e.g., every tenth of a second [18]. In addition, in case of unexpected events, such as an emergency braking or a road warning, vehicles notify their surroundings by transmitting event-triggered messages [19]. To allow efficient use of the radio resources and prevent congestion of the communication channel, the message generation rate is in general variable, such as based on the vehicle speed (faster the vehicle more frequent the messages), dynamics (e.g., new message generation upon direction change), properties (e.g., police car transmitting more frequent messages), and events (e.g., higher message rate upon sudden slowdown) [18], [19], [20].

V2V communications pose stringent quality of service (QoS) requirements, where the transmitted messages need to be delivered with high reliability in a certain time limit (referred to as latency) [21]. The required reliability should be ensured at a certain communication range to account for the timely reaction of the vehicles, which depends on the use case as well as the velocity of the involved vehicles [22]. Further, the use cases should operate under expected vehicle densities according to the scenario (e.g., rural or urban, day or night, etc.), which in turn implies that the communications system should also support a high load of data traffic when necessary [23]. Given the scarce spectrum, maintaining the required QoS for V2V communications, therefore, calls for efficient ways of resource allocation.

Radio Resource Allocation for V2V Communications

Conventionally, there exist two approaches to radio resource allocation for V2V communications: distributed and centralized [24], as illustrated in Fig. 1.2. With the distributed approach, vehicles autonomously select the radio resources that they will transmit, mainly by sensing whether the resources are occupied by other vehicles' transmissions or not. In the centralized approach, resources for V2V transmissions are coordinated by a central entity available, e.g., via cellular networks, based on the vehicle requests.

Among the standardized RATs, C-V2X offers a centralized resource allocation mode where the resources for V2V communication are coordinated by the BS [12]. C-V2X also offers a distributed resource allocation mode, in which vehicles select resources autonomously based on the specified sensing algorithm, without the need for a BS. The WAVE/ITS-G5 utilizing the Wi-Fi-based IEEE 802.11p standard also relies on a specific distributed scheduling scheme based on carrier-sensing mechanisms [11].

The distributed approach has the advantage of not relying on a central coordinator, such as a BS in the case of cellular networks, hence also eliminating the necessity of a network infrastructure deployment. On the other hand, the lack of a centralized controller complicates

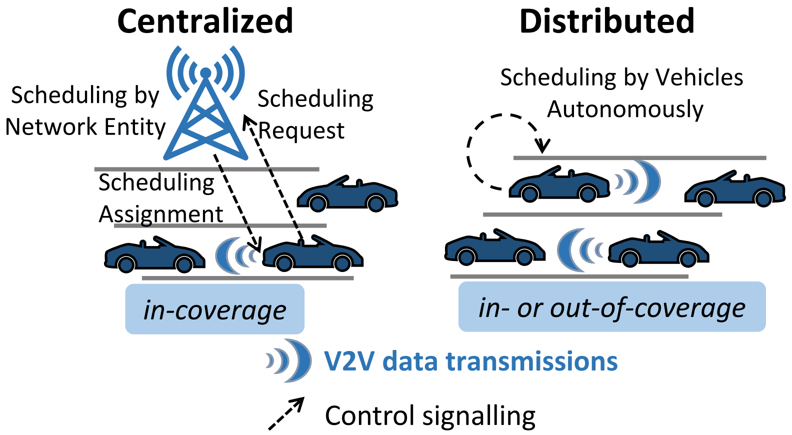


Fig. 1.2 Two approaches to radio resource allocation for V2V communications.

the resource allocation task. Since sensing measurements of the vehicles are local and transient, V2V transmissions become prone to the so-called "hidden node problem", whereby vehicles can only hear their immediate neighbors but no other nodes in the network. This results in conflicts in resource selection, thereby degrading the communication performance especially in high density and high mobility scenarios. While several techniques, such as cooperation between the vehicles, can mitigate the hidden node problem, these, however, create inefficiencies in terms of resource utilization, such as by requiring additional signaling, delay the transmissions, reduce the throughput, and necessitate complicated measures to be implemented preferably by all vehicles.

In the centralized approach, the global view of the network at the centralized entity allows an interference-free assignment of resources, which brings more efficient resource utilization. Initial studies evaluated the centralized scheme in cellular networks in comparison to various distributed schemes, where more than 65% increase in the effective communication distance (the distance at which at least 95% of relevant V2V messages are received successfully) is shown in [25], and around 125% increase of the inter-vehicle range at which 90% packet reception rate is achievable is shown in [26], under the same traffic density. Among the recent works, a centralized resource allocation algorithm in [27] is shown to outperform the distributed scheduling specified by the cellular standard with a packet reception rate close to the upper bound (no resource collisions) under low load, and by almost doubling the distance at which 90% reception rate is achieved, under high load.

Intermittent Coverage Problem

Although centralized resource allocation offers superior V2V communication performance over distributed approaches, the availability of the entity coordinating the resources is not always guaranteed. Specifically, in cellular networks, vehicles can lose connectivity to the BSs at any time [28]. This *intermittent coverage problem* could be due to insufficient infrastructure deployment, such as a lack of BSs covering the entire roadway, creating so-called "coverage gaps". Further, despite any deployment, vehicles may still travel through areas that physically impede their connection to the BSs, such as tunnels. More inevitably, highly dynamic vehicular environment may result in sudden changes in wireless channel (e.g., deep fading due to blocking objects), interrupting the links between the vehicles and the BSs. In addition, external interference such as originating from a malicious source or an adjacent frequency band may also cause unsuccessful signal reception on these links (as well as on the V2V links), which we however leave out of our scope. While coverage gaps or areas such as tunnels or underground spaces outside the coverage could be known to the network operator, thus being expected and predictable, more abrupt and shorter losses "within coverage" are rather unpredictable.

Upon losing connectivity to the centralized entity managing the V2V communication resources, irrespective of the reason or type of the loss, vehicles have to resort to distributed resource allocation, which is, however, not efficient from the resource utilization perspective as emphasized above. Depending on the traffic demand and the availability of resources, along with the characteristics of the interruptions (e.g., duration and location), intermittent network coverage could severely impact the performance of V2V transmissions, mainly in terms of reliability and latency. Given that the service requirements of V2V applications must be satisfied irrespective of cellular network coverage, intermittent coverage poses a key but often omitted problem that needs to be addressed in V2V communications.

From a conventional resource management perspective, various approaches can tackle this problem. The most straightforward one is to provision additional resources. This is, however, not feasible given the scarce and costly spectrum. Alternatively, lower data traffic could be admitted to the available resources. On one hand, transmission parameters of vehicles, e.g., message transmission rate, can be adjusted via admission or congestion control. While avoiding violation of quality guarantees for specific services, these methods restrict, e.g., the lower-priority ones, and further require mechanisms that can adapt to dynamic load conditions. On the other hand, fewer vehicles can be admitted to the road sections of interest, such as by lane closures or varying the road speed limitations. While such mobility measures can establish road safety, they create additional congestion and delays in road traffic. The intermittent coverage problem thus necessitates more efficient methods of resource allocation.

1.2 Goals, Contributions, and Outline of the Thesis

In this thesis, we are concerned with the impact of intermittent connectivity between the vehicles and the cellular network infrastructure managing the V2V resource allocation, on the performance of V2V communications. In this regard, we aim at efficient utilization of radio resources while ensuring reliable inter-vehicle communication under variable circumstances of cellular network connectivity. To this aim, we provide novel methods for resource allocation with the intend of filling the above-mentioned performance gap between the centralized and distributed ways of scheduling. Our main approach follows the question *whether we can exploit the benefits of centralized radio resource allocation to enhance the performance of V2V communications under expected and unexpected intermittent coverage conditions*. Differing from previous approaches, we propose a centralized solution that provides resources for V2V communications proactively, before the vehicles experience any coverage interruptions. Our focus primarily lies on V2V communications in known or expected areas outside the cellular network coverage. We then extend our solutions also to the areas with (poor) coverage, where vehicles rather suffer from unexpected and relatively shorter connectivity interruptions to the BSs.

Fig. 1.3 provides the outline of this thesis work. Chapter 2 introduces the relevant background and related work. We first provide an overview of ITS and V2X communications and elaborate on the need for V2V communications. We then present existing resource allocation (RA) techniques applicable to V2V communications and how the standardized RATs address the RA problem. Later, we review and analyze the performance of the state-of-the-art RA methods and expand on the intermittent coverage problem. In the remaining of Chapter 2, we turn our attention to the domain of artificial intelligence (AI) and machine learning (ML). Our focus is on reinforcement learning (RL), which we apply in our proposed methods. We review the basics and various types of RL algorithms and present the specific algorithm that we employ. We finalize Chapter 2 by surveying the applications of RL to the resource allocation problems in V2X communications.

Chapter 3 presents the system model we consider in this study, followed by the definition of the key performance metrics we utilize to evaluate the proposed algorithms. We have employed realistic models for vehicular road and data traffic, as well as wireless communications, and considered mainly simulation as a research tool in our evaluations. The remaining of Chapter 3 describes our simulation environment. We have combined widely-accepted simulator software, and developed additional functionalities to support V2V communication protocol based on the cellular standard. To implement our RL-based solutions, we have re-used openly available ML software libraries, and developed an interface that enables interaction between our RL model and the network simulator.

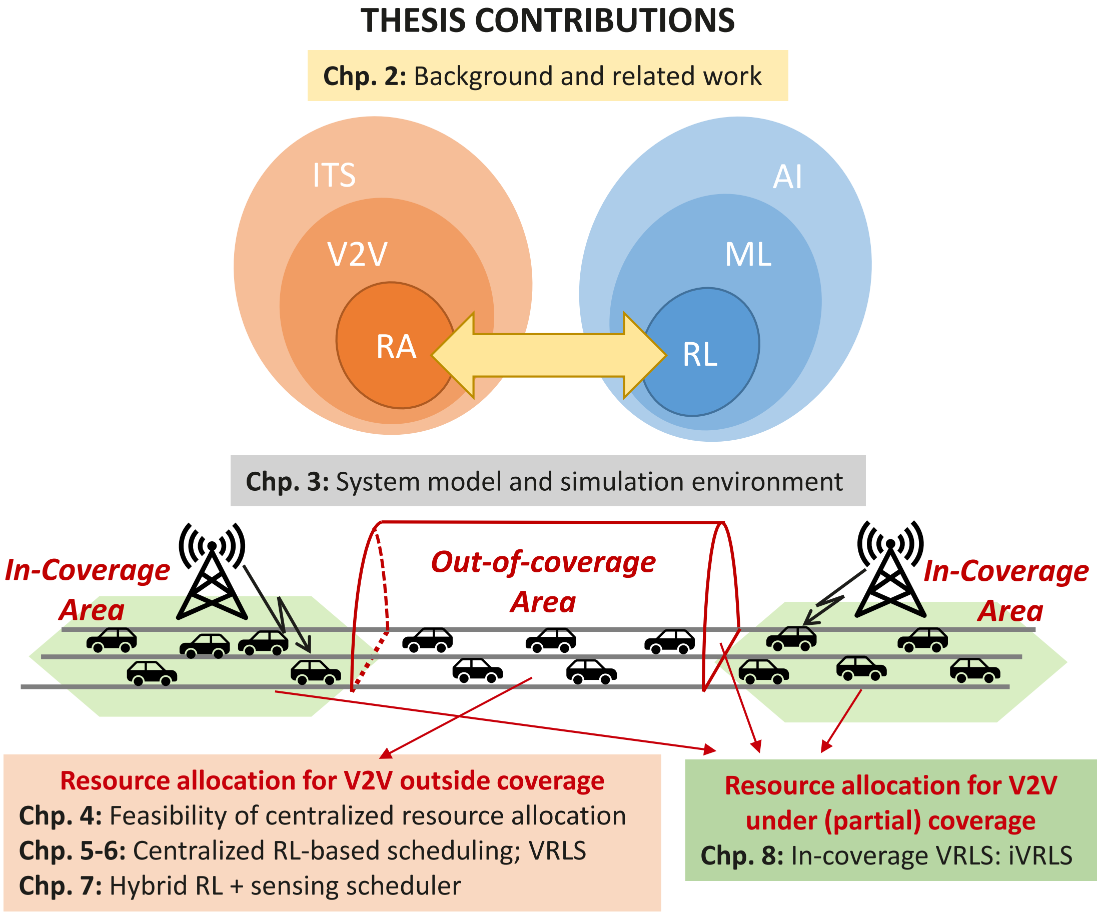

THESIS CONTRIBUTIONS
Chp. 2: Background and related work
ITS
V2V
RA
AI
ML
RL
Chp. 3: System model and simulation environment
In-Coverage Area
Out-of-coverage Area
In-Coverage Area
Resource allocation for V2V outside coverage
Chp. 4: Feasibility of centralized resource allocation
Chp. 5-6: Centralized RL-based scheduling; VRLS
Chp. 7: Hybrid RL + sensing scheduler
Resource allocation for V2V under (partial) coverage
Chp. 8: In-coverage VRLS: iVRLS

Fig. 1.3 Overview of the thesis contributions.

We begin our study with an exploratory work in Chapter 4. We study the feasibility of the idea of reserving and provisioning resources for V2V communications in known, delimited out-of-coverage areas (DOCAs) by a centralized entity, which assigns resources to each vehicle at the "edge" of such areas, before the vehicle enters the area. We first analyze the required amount of resources to support desired V2V communication reliability where vehicles utilize resources reserved for them, without any scheduling overhead. This preliminary analysis, therefore, gives an initial understanding of the boundaries of the resource allocation problem. The results indicate that the required amount of resources depends much more on the data traffic load (which would be impacted by the vehicle density and message generation rate) and size of the DOCA as compared to target reliability, hence serving as a guideline to consider these factors when allocating the resources for a DOCA.

We first analyze the required amount of resources to support desired V2V communication reliability where vehicles get non-interfering resources, thereby constituting an ideal scenario. This preliminary analysis, therefore, gives an initial understanding of the boundaries of the resource allocation problem. The results indicate that the required amount of resources depends much more on the data traffic load (which would be impacted by the vehicle density and message generation rate) and size of the DOCA as compared to target reliability, hence serving as a guideline to consider these factors when allocating the resources for a DOCA.

Reserving resources can be primarily beneficial for event-triggered messages, e.g., due to accidents, as these can not be pre-scheduled but typically require high reliability and low latency. On the other hand, the characteristics of the periodic type of traffic (e.g., periodicity, size, etc. per vehicle) that takes place in a DOCA can be well-known beforehand. Therefore, for this type of traffic, we propose pre-scheduling by a centralized entity. The entity has access to the BSs delimiting the DOCA, from which it obtains relevant information collected from the vehicles, and in turn, provides the scheduling outcomes to them as they approach the DOCA. The scheduling decisions are taken heuristically based on the predicted future locations of the vehicles and the resulting propagation and interference conditions among them, together with the latency requirements of V2V messages. We evaluate the performance of such a scheduler under varying characteristics of DOCA (traffic densities, vehicle speeds, transmit powers, etc.). Our results show that the rate of successful transmissions gets highly impacted by the prediction errors combined with varying conditions in the DOCA. Chapter 4 concludes that while the idea of pre-allocating resources for V2V communications in expected out-of-coverage areas is feasible, utmost importance is needed to take vehicular mobility, density, traffic load, and wireless channel characteristics into account when scheduling the resources. This calls for efficient, flexible, and practical algorithms as a solution.

While the mainstream approach for resource management is to formulate an optimization problem and solve it optimally or sub-optimally depending on the performance-complexity trade-offs, this becomes infeasible in vehicular networks due to their highly dynamic nature [29]. Such approach becomes even more challenging in our case since the resource allocation task relates to an out-of-coverage area where the conditions are constantly changing and only limited information from the area is available. Instead, machine learning methods could become useful. In particular, reinforcement learning (RL) has been proven to be successful in tasks having time-varying dynamic environments under uncertainty, and recently found promising applications in the wireless communications domain [30]. In RL, the problem is addressed by designing a reward signal that correlates with the ultimate objective, and the learning algorithm can automatically find out a satisfactory solution to the problem by training its policy to maximize the reward [30]. The possibility of flexibly designing such a reward signal makes RL-based approaches in particular attractive, as this avoids exact modeling of the system and designing the objective using conventional approaches [30]. Motivated by the successful applications of RL to the resource management problems in general [31], and to the vehicular networks in particular [29], we propose an RL-based approach for centrally scheduling the V2V communication resources. The proposed RL-based scheduler utilizes the information pertaining to the DOCA such as the occupancy of radio resources it obtained from the BSs delimiting it, and trained with the reward signal that we designed to maximize the reliability of V2V transmissions in the DOCA.

In Chapter 5, we first study the feasibility of such an RL-based solution by considering several sanity-check scenarios having a limited number of vehicles and resources, and basic mobility. These scenarios allow us to compute the optimal performance of a scheduler, and compare our solution with respect to that. Our evaluations show that the RL scheduler can learn to avoid resource collisions, namely concurrent or interfering transmissions, where it can converge to near-optimal solutions and outperform the existing distributed scheduling schemes in the considered settings. While our encouraging results motivated us to consider more complex and realistic scenarios, we had to modify our RL design to address desirable performance requirements in different target scenarios. It would be, however, impractical to re-design, re-train, and re-evaluate a new RL solution every time the environment changes, even if such a change is substantial. A practical RL-based solution should be applicable to different and varying conditions in the environment, which are natural to vehicular networks.

In Chapter 6, to overcome the above-mentioned challenges of RL, we propose a unified solution called *Vehicular Reinforcement Learning Scheduler (VRLS)*. VLRS is designed by unifying the state information and the reward signal input to the RL model, besides its other components, so that the structure of the solution remains the same irrespective of what kind of setting it is applied to. This enables broad applicability of VRLS to different practical scenarios having arbitrary sizes of out-of-coverage area, with any number of vehicles inside, utilizing an arbitrary number of resources in time and frequency. Further, such a design facilitates efficient and practical training over simpler and simulated environments. We show that, with limited or no retraining, the learning performed by VRLS over simplified environments can be transferred to a set of more realistic, complex environments varying in terms of mobility, wireless channel characteristics, area size, network load, and traffic within the scope of the practical settings we consider. VRLS outperforms the state-of-the-art distributed scheduling solution in terms of resource utilization, by reducing the packet loss by half in case of overloaded network conditions and performing very close to the maximum possible level under low load, while achieving a similar level of fairness and latency.

Nevertheless, a trained policy of VRLS may not tackle all possible circumstances in the environment it will be deployed. Namely, there could occur some unexpected or extreme situations such as a road congestion due to an accident, or cases for which the scheduling policy might not be trained. In these cases, the actions of the scheduling agent may simply become infeasible. In fact, VRLS assigns each vehicle only a single resource, and in the case of persistent resource collisions, vehicles would not have a chance to utilize another resource for their V2V transmissions. In order to overcome this problem, we propose a *hybrid* resource allocation solution in Chapter 7, which combines our centralized RL-based approach with the distributed sensing-based scheduling approach, for areas outside the network coverage. In this method, while the centralized RL scheduler “recommends” a *subset* of resources to each vehicle for their V2V transmissions in the DOCA, the vehicles determine their final resources by dynamically performing sensing on these resources in an autonomous manner. Consequently, resource allocation task is adapted to transient and local conditions of the vehicular network that might not always be predictable by the central scheduler. Our evaluations show that while this approach performs on a par with VRLS under expected conditions, the benefit comes into play under non-ideal conditions in the environment, such as when a road accident creates a congested, stop-and-go traffic. The hybrid solution can, therefore, supplement VRLS by interceding upon detection of such unexpected conditions in the network (e.g., the vehicular density increasing beyond a threshold, reported performance metrics not matching to requirements, etc.).

Finally, in Chapter 8, we turn our attention to the areas where vehicles suffer rather from shorter and unexpected interruptions on the cellular links to the centralized scheduler, such as due to poor channel conditions. Such imperfections are often omitted in the literature. In contrast, the proposed heuristics for scheduling V2V communications often require a high frequency of scheduling updates, thus increasing their dependency on the reliability of these links, in addition to incurring heavy signaling overhead. Therefore, more efficient algorithms that can operate at least equally well under realistic, intermittent coverage conditions are needed. Our solution targeting this problem, named iVRLS (in-coverage VRLS), is based on our centralized RL-based approach as in VRLS. Differently, owing to the cellular coverage, iVRLS can collect more accurate and up-to-date knowledge of the vehicular mobility and data traffic, and can provide rather frequent resource assignments that are possible anytime and anywhere, however at the cost of increased signaling overhead. Resource assignment of iVRLS is based on the estimated current and future interference conditions among the vehicles’ transmissions until the next assignment, whereby the next assignment might be delayed by any coverage loss. Besides iVRLS, we propose a simple enhancement to the existing centralized schedulers such that vehicles keep using their present resources in case they experience any connectivity interruptions to the scheduler, until they connect back. Our performance evaluations under realistic, non-ideal coverage conditions show that in comparison to a state-of-the-art centralized scheduling algorithm in [32], iVRLS achieves similar performance under a relatively low traffic load. With a high load, iVRLS can make more efficient use of the resources by delivering marginally better V2V communication reliability while requiring a lower rate of scheduling updates. In the case of coverage losses, iVRLS yields marginally larger V2V distances at which given target reliability is achievable. As such, iVRLS offers a robust alternative to existing schedulers under varying network coverage conditions.

With Chapter 9, we conclude the thesis by summarizing our contributions and providing an outlook for further research.

1.3 Publications by the Author

During this thesis work, the author has published the following papers. The pre-published parts of the thesis in these papers are indicated with a footnote at the beginning of each corresponding chapter.

1. [33] T. Sahin and M. Boban, “Radio Resource Allocation for Reliable Out-of-Coverage V2V Communications,” *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1-5, doi: 10.1109/VTCSpring.2018.8417747.

2. [34] T. Şahin, R. Khalili, M. Boban and A. Wolisz, “Reinforcement Learning Scheduler for Vehicle-to-Vehicle Communications Outside Coverage,” *2018 IEEE Vehicular Networking Conference (VNC)*, 2018, pp. 1-8, doi: 10.1109/VNC.2018.8628366.

3. [35] T. Sahin, R. Khalili, M. Boban and A. Wolisz, “VRLS: A Unified Reinforcement Learning Scheduler for Vehicle-to-Vehicle Communications,” *2019 IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, 2019, pp. 1-7, doi: 10.1109/CAVS.2019.8887834.

4. [36] T. Şahin, M. Boban, R. Khalili and A. Wolisz, “A Hybrid Sensing and Reinforcement Learning Scheduler for Vehicle-to-Vehicle Communications,” *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1136-1143, doi: 10.1109/IEEECONF44664.2019.9048691.

5. [37] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Şahin, and A. Kousaridas, “A Tutorial on 5G NR V2X Communications,” in *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1972-2026, thirdquarter 2021, doi: 10.1109/COMST.2021.3057017.

6. [38] T. Şahin, M. Boban, R. Khalili and A. Wolisz, “iVRLS: In-coverage Vehicular Reinforcement Learning Scheduler,” *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1-7, doi: 10.1109/VTC2021-Spring51267.2021.9448993.

7. [39] T. Şahin, R. Khalili, M. Boban and A. Wolisz, “Scheduling Out-of-Coverage Vehicular Communications Using Reinforcement Learning,” in IEEE Transactions on Vehicular Technology, vol. 71, no. 10, pp. 11103-11119, Oct. 2022, doi: 10.1109/TVT.2022.3186910.

In addition, parts of the ideas developed in this thesis work are protected by the following patent applications:

1. [40] T. Sahin and M. Boban, “Devices and methods for D2D communication,” Patent WO2 019 174 744, Sep., 2019. [Online]. Available: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2019174744

2. [41] T. Sahin, M. Boban, and M. Webb, “Network entity, user equipments and methods for using sidelink resources,” Patent WO2 020 244 741, Dec., 2020. [Online]. Available: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2020244741

Chapter 2

Background and Related Work

2.1 Intelligent Transport Systems and Vehicular Communications

Intelligent Transport Systems (ITS) refer to a collective set of technologies that provide services for different modes of transport and traffic management, aiming to enable safer and better-informed users, and more coordinated, “smarter” use of vehicles and transport networks, with a particular focus on road transport [3], [4]. ITS is standardized by international as well as local bodies, such as the International Organization for Standardization (ISO), and the European Telecommunications Standards Institute (ETSI) in Europe, respectively. The developed standards are subject to global and local regulatory frameworks, e.g., as defined by the International Telecommunication Union (ITU) and the Federal Communications Commission (FCC) in the United States (US).

While ITS provides technologies installed at the roadside or in vehicles, the recently established area of Cooperative ITS (C-ITS) is based on the *communication* between these systems. C-ITS encompasses a wide range of applications based on *vehicular communications* for the goal of travel safety, minimizing environmental impact, improving traffic management, and maximizing commercial and public benefits of transportation. Vehicular communications enable many C-ITS services ranging from cooperative awareness among vehicles to automated driving. These services would not have been possible or not as efficiently achievable via other technologies, such as sensor-based systems [5]. With vehicular communications, vehicles can exchange information indicating their existence and status, as well as conditions relating to their surroundings, such as forward collisions or road hazards. Further, vehicles can make use of each other’s sensor data shared via vehicular communications. This information exchange, in turn, enables vehicles to cooperate with their surroundings and coordinate their actions,

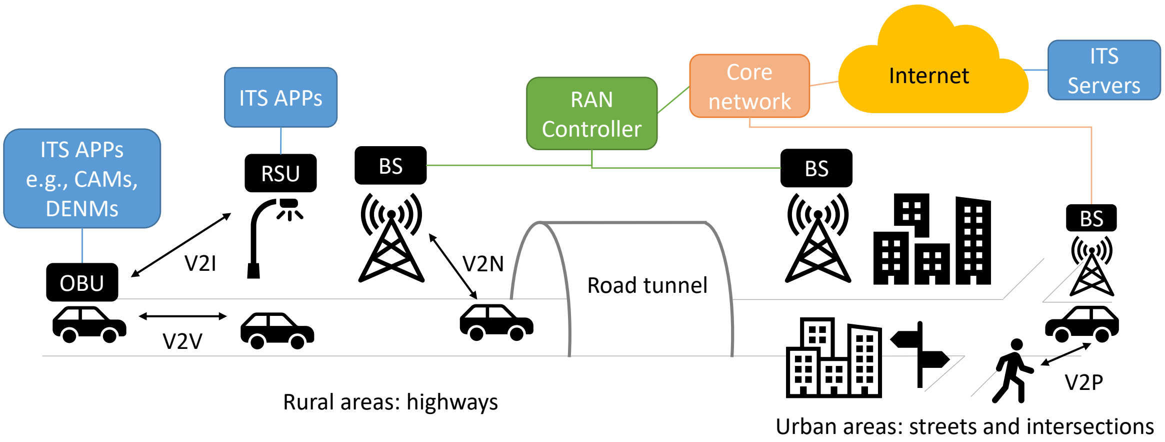


Fig. 2.1 A landscape of V2X communications.

such as realizing a lane-changing maneuver, managing an intersection, or driving together in groups, all in a safe, efficient, and automated manner. In this way, vehicular communications enable vehicles to anticipate and avoid dangerous situations, reduce collisions, and potentially save lives, besides mitigating traffic congestion and energy consumption, and enhancing the overall travel experience.

Figure 2.1 provides a landscape of vehicular communications as envisioned by C-ITS. Vehicular communications involve a variety of network entities and components, different types of communications among them, and wireless technology standards enabling these communications. In the following, we provide an overview of these and introduce the related terminology that we use in this thesis.

Communicating Entities

Vehicular communications involve the following communicating entities [13]:

- **Vehicles:** road vehicles carrying people or goods on public roads and highways, such as cars, buses, trucks, and motorcycles. Vehicles are equipped with an on-board unit (OBU) to communicate, which consists of a set of components, interfaces, and functionalities [42]. These include ITS applications, positioning and security processors, communication protocol stacks, antenna connectors, etc. required to support underlying V2X communication technology.

- **Vulnerable road users (VRUs):** non-motorized road users such as pedestrians, cyclists, pets, etc., as well as vehicles with less than four wheels such as mopeds and motorcycles [43]. VRUs are equipped with usually hand-held devices to communicate, which contain ITS applications and can support different communication technologies.

- **Roadside units (RSUs):** communication unit connected to stationary roadside infrastructure such as traffic lights and variable road signs. The road traffic management equipment installed along the roadside conveys traffic or traveler information to passing drivers, and is equipped with ITS applications that can support different communication technologies. RSUs may have a wired or wireless long-range backhaul connection, e.g., to the cellular network.

- **Cellular base stations (BSs):** stationary radio transceivers that provide mobile users (such as vehicles and VRUs) wireless access to the cellular network. Each BS serves a certain area, referred to as a cell. Depending on its size, the area served by a BS is called a macrocell, microcell, picocell, etc. [44]. Together with the users they serve, BSs constitute a radio access network (RAN) [45]. RAN is then connected to the core network, which provides access to the global Internet. Multiple BSs, e.g., serving a particular area, can be connected to a centralized RAN controller that provides several radio functionalities such as radio resource management [45].

The broad term *vehicular network* refers to the wireless interconnection of vehicles as well as other communicating entities surrounding them like the ones listed above. A narrower term, *vehicular ad-hoc network (VANET)*, refers to the wireless network formed solely by vehicles (as well as RSUs in some cases) in an ad hoc manner, i.e., without any dependency on any other infrastructure [46].

Types of Vehicular Communications

As introduced in Chapter 1, vehicles can have the following types of communications, which denote the endpoints (source and destination) of the information exchange [13], [47]:

- **Vehicle-to-vehicle (V2V):** exchange of information between vehicles, also referred to as *inter-vehicle communication (IVC)*.

- **Vehicle-to-infrastructure (V2I or I2V):** exchange of information between a vehicle and a roadside infrastructure using an RSU.

- **Vehicle-to-pedestrian (V2P or P2V):** exchange of information between a vehicle and a pedestrian or any other VRU using a hand-held device.

- **Vehicle-to-network (V2N or N2V):** exchange of information between a vehicle and a cloud server such as hosting ITS applications or services, via cellular network using BSs.

In this thesis, we use the terms V2V, V2I, and V2P to refer to the short-range communication taking place on the direct wireless link between the respective entities in close proximity. The data exchange between different endpoints could also take place *indirectly* using the cellular network, which are referred to as V2N2V, V2N2I, and V2N2P communications [13], respectively. All above ways of communication are collectively referred to with the umbrella term vehicle-to-everything (V2X) communications.

Fundamentals of Vehicular Communications

V2X communications involve the following fundamental properties of wireless communication [44]:

- **Modes of communication:** Transmission over the wireless medium is *broadcast* by its nature. Namely, a transmitted message from a network entity can be received by multiple entities in proximity regardless of the intended receiver of the message. However, the actual message sent can be intended as a *unicast* that targets a specific recipient, or as a *multicast* to address a group of recipients. While unicast communication is also referred to as point-to-point or one-to-one communication, multicast and broadcast communications are also referred to as point-to-multipoint or one-to-many communications.

- **Half-duplex radio:** Radio devices on vehicles are subject to so-called *half-duplex* constraint, which prevents them to transmit and receive at the same time in the same frequency band. This requires coordination of transmissions in time, e.g., to avoid simultaneous transmissions among vehicles that intend to address each other.

- **Data transmission and the communication channel:** V2X messages are transmitted in *packets* by the digital radio equipment of the vehicles. A message can be transmitted in one or more packets depending on its size. Each packet carries a certain number of bits, which may include error detection/correction and control information besides the data payload. Packets are transmitted over the wireless medium using certain amount of radio resources, with a certain modulation and coding scheme (MCS). A packet can be retransmitted multiple times to increase its reception probability. In this thesis, we assume that a message is always transmitted using a single packet, and without any retransmissions.

  While the wireless medium has the broadcast advantage, where the same transmitted message can be simultaneously received by multiple receivers, hence not requiring multiple transmissions, this turns into a problem in case multiple users transmit at the same time and frequency, where their transmissions mutually interfere with each other. Depending on the relative power and the coding of the interfering signals, interference can result in erroneous receptions. Successful reception of a single packet depends on the *signal-to-interference-and-noise ratio (SINR)* at the receiver. SINR reflects how strong the received, desired signal is in relation to the interfering signals and noise. The received instantaneous SINR per individual radio resource can be mapped into a single effective metric over the utilized set of resources, which is required to be larger than a threshold for successful reception, depending on the MCS used to transmit the packet.

Besides the interference and noise, radio channel over which the packets are transmitted is further susceptible to the following impediments that impact the SINR. First, the transmitted signal gets attenuated with distance, due to the dissipation of the radiated power, referred to as *path loss*. Second, any object between the transmitter and receiver can absorb, reflect, scatter, or diffract the signal, by attenuating its power, which is called as *shadowing* or *large-scale fading*. Third, in addition to or instead of the direct path between the transmitter and the receiver, called as *line-of-sight (LOS)* path, transmitted signal can reach receiver via several propagation paths or multi-path components (MPCs), each having a different amplitude and phase. Superposition of MPCs give rise to variations of the received signal over short distances in the order of the signal wavelength, called as *small-scale fading*.

Characteristics of Vehicular Communications

Vehicular communications exhibit several distinctive characteristics as compared to other types of communication networks. On one hand, they come with several attractive features, as follows [46]:

- **Higher power:** While device power is usually an issue in mobile networks, in the case of vehicles, the onboard battery can continuously provide energy for communication and processing purposes, without its lifetime posing any problem.
- **Higher computational capability:** On-board units of the vehicles can offer significant computing, communication, and sensing capabilities, unlike traditional mobile communication devices.
- **Predictable mobility:** Vehicular mobility tends to follow certain patterns, e.g., governed by road topology, speed limits, traffic flow, and planned route. Location information of vehicles is often available via satellite or radio-based positioning technologies.

Given the average speed and current velocity of a vehicle as well as the road trajectory, its future position is predictable.

On the other hand, vehicular communications need to cope with several challenges as in the following [46], [48]:

- **Large scale:** Unlike traditional networks with a limited number of users, vehicular networks may involve many participants, in principle spanning the entire road network, with a high amount of data traffic per user.
- **High mobility:** Vehicles travel in a wide range of mobility settings, alternating over time and space. In rural areas, vehicles may reach up to 500 km/h of relative speeds, where only a few vehicles per km are present. On the opposite extreme, highly dense traffic jams may occur during rush hours in the city centers. Separate motions and trajectories of different vehicles further create dynamically changing network topology where links between nodes frequently connect and disconnect, otherwise yielding variable inter-node gaps.
- **Environmental conditions:** The vehicular environment poses unfavorable channel conditions involving multiple blocking or reflecting objects such as buildings, vehicles (both static and mobile), and vegetation (e.g., trees) [49].
- **Security and privacy:** Vehicular networks raise concerns about security and privacy towards their successful deployment. On one hand, it is essential to avoid life-critical information to be modified, truncated, or inserted by an attacker, where only authorized users are allowed to manipulate the exchanged data. On the other hand, the privacy of the vehicular users should be respected, e.g., in terms of guaranteeing their anonymity and non-traceability.

Standardized Radio Access Technologies for V2X

Real-world deployment of vehicular communications involves diverse stakeholders ranging from car manufacturers to telecommunication equipment vendors, and from public transport authorities to mobile network operators. Therefore, interoperability of communications between them becomes a key issue, which can be assured via standardization of the underlying technology. Standardization helps maximize safety and quality while supporting regulation, legislation, as well as the enlargement of the vehicular communications technology market.

Presently, V2X communications are supported by two families of standardized radio access technologies (RATs): the Wi-Fi-based WAVE/ITS-G5 and the cellular V2X (C-V2X) standard, as shown in Fig. 2.2.

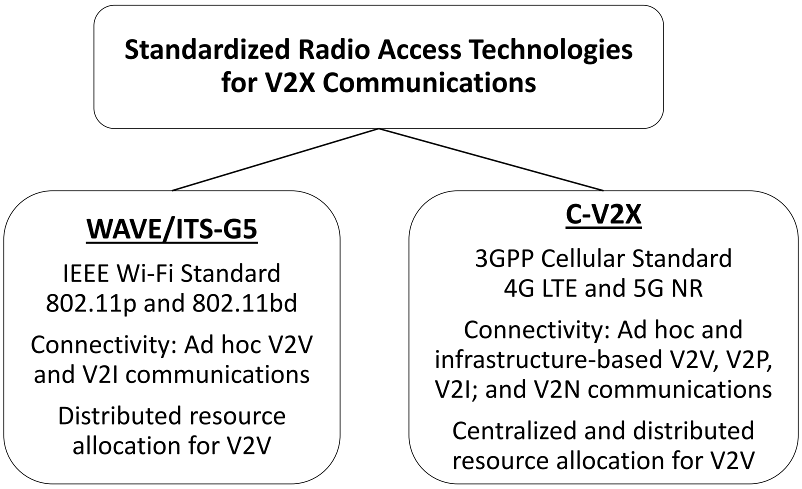

Standardized Radio Access Technologies for V2X Communications
WAVE/ITS-G5
IEEE Wi-Fi Standard 802.11p and 802.11bd
Connectivity: Ad hoc V2V and V2I communications
Distributed resource allocation for V2V
C-V2X
3GPP Cellular Standard 4G LTE and 5G NR
Connectivity: Ad hoc and infrastructure-based V2V, V2P, V2I; and V2N communications
Centralized and distributed resource allocation for V2V

Fig. 2.2 Standardized radio technologies for vehicular communications.

- **WAVE/ITS-G5 standard:** Wireless Access in Vehicular Environments (WAVE) [8], [50] and Dedicated Short-Range Communications (DSRC) in the United States, together with its European variant ITS-G5, is the first standard introduced for vehicular communications. The radio technology allows ad hoc communications between vehicles, as well as V2I communications between the vehicles and RSUs, without requiring any network infrastructure. The communication is based on the IEEE 802.11p standard (and its upcoming successor IEEE 802.11bd). IEEE 802.11p was developed as an amendment to the wireless local area network (WLAN) standard IEEE 802.11, to enable wireless communications in a vehicular environment.

  The word “dedicated” in DSRC refers to the dedicated 75-MHz spectrum at 5.9 GHz band allocated by the FCC for ITS operations. The term “short-range” refers to the communication that takes place over hundreds of meters, which is shorter than the distance typically supported by cellular technology [9]. The European counterpart of the standards ITS-G5 (G5 stands for the 5.9 GHz frequency band also allocated for ITS services in Europe) is developed by ETSI [10], which shares great similarities with DSRC [51].

- **Cellular V2X (C-V2X) standard:** C-V2X is the mobile cellular radio technology that supports vehicular communications based on the Fourth Generation Long Term Evolution (4G LTE) and the Fifth Generation New Radio (5G NR) wireless cellular standards developed by the 3rd Generation Partnership Project (3GPP)[1] [12]. C-V2X

[1] During the course of this thesis work, the LTE C-V2X standard has been transformed from a preliminary research work into first real products, whereas 5G NR C-V2X standard was not yet available.

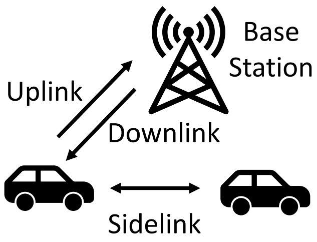


Fig. 2.3 Communication interfaces in cellular networks.

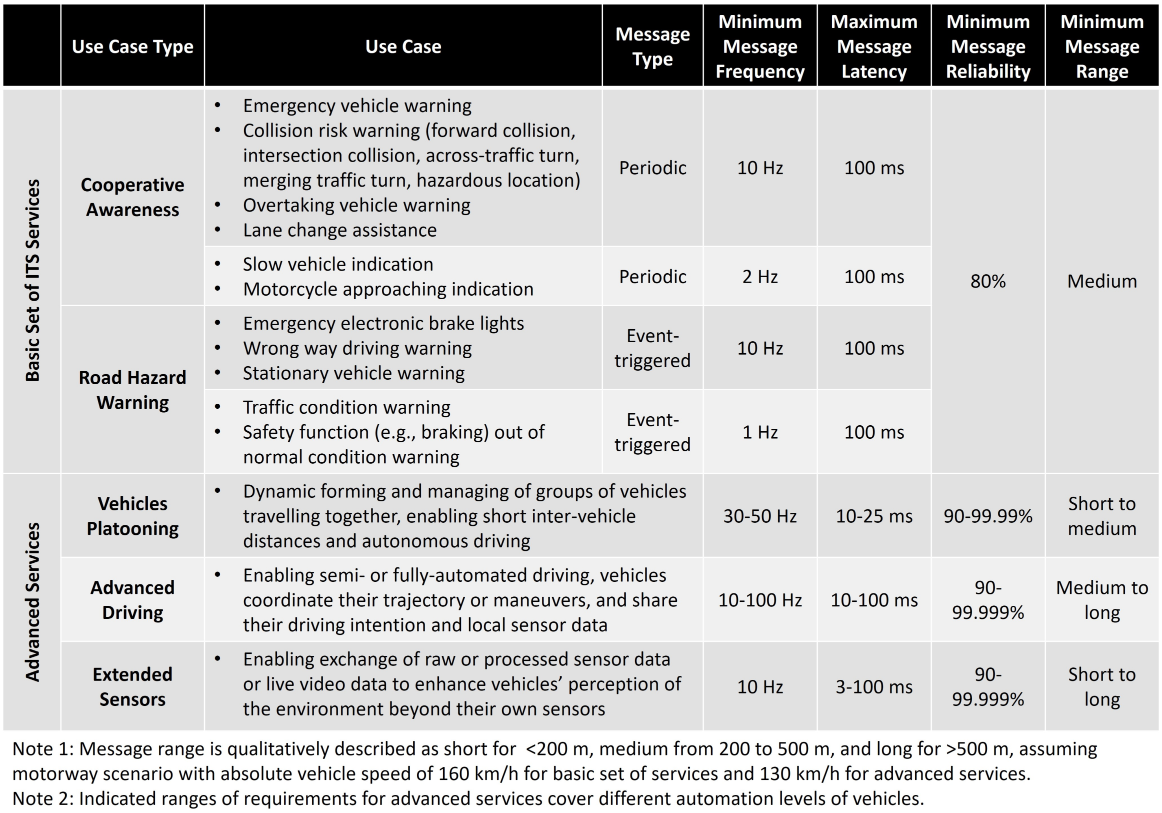
offers V2V, V2I, V2P, and V2N communications. V2N communication takes place over the traditional *uplink (UL)* and *downlink (DL)* that are used for one-way communication from the vehicles to the cellular base station (BS), and from the BS to the vehicles, respectively, as illustrated in Fig. 2.3. V2V and V2P communications take place over the interface termed *sidelink (SL)*, which enables direct two-way communication using a separate transceiver. Depending on whether RSUs are implemented as BSs or cellular user equipment, V2I communication in cellular networks takes place over uplink/downlink or sidelink, respectively.

For sidelink communications, the cellular standard offers two modes: centralized and distributed. In the centralized mode, resource allocation for V2V (as well as V2P or V2I) communication is coordinated by the BS. In the distributed mode, vehicles (and other user entities) do resource allocation autonomously without the need for a BS, thus it can take place outside the cellular network coverage as well, without any need for a cellular network infrastructure deployment.

Cellular Network Coverage Conditions

V2X communications may take place within the following scenarios in terms of the cellular network coverage [52]:

- **In-coverage:** communicating vehicles are all located within the coverage of one or more BSs that can serve the vehicles.
- **Out-of-coverage:** communicating vehicles are all located outside the coverage of a BS, i.e., without any access to the cellular network, such as when traveling through parts of roads lacking cellular deployment, e.g., tunnels or underground spaces.
- **Partial coverage:** when at least one of the communicating vehicles is within the coverage of a BS while at least another one is outside.



	Use Case Type	Use Case	Message Type	Minimum Message Frequency	Maximum Message Latency	Minimum Message Reliability	Minimum Message Range
Basic Set of ITS Services	Cooperative Awareness	• Emergency vehicle warning • Collision risk warning (forward collision, intersection collision, across-traffic turn, merging traffic turn, hazardous location) • Overtaking vehicle warning • Lane change assistance	Periodic	10 Hz	100 ms	80%	Medium
		• Slow vehicle indication • Motorcycle approaching indication	Periodic	2 Hz	100 ms		
	Road Hazard Warning	• Emergency electronic brake lights • Wrong way driving warning • Stationary vehicle warning	Event-triggered	10 Hz	100 ms		
		• Traffic condition warning • Safety function (e.g., braking) out of normal condition warning	Event-triggered	1 Hz	100 ms		
Advanced Services	Vehicles Platooning	• Dynamic forming and managing of groups of vehicles travelling together, enabling short inter-vehicle distances and autonomous driving		30-50 Hz	10-25 ms	90-99.99%	Short to medium
	Advanced Driving	• Enabling semi- or fully-automated driving, vehicles coordinate their trajectory or maneuvers, and share their driving intention and local sensor data		10-100 Hz	10-100 ms	90-99.999%	Medium to long
	Extended Sensors	• Enabling exchange of raw or processed sensor data or live video data to enhance vehicles' perception of the environment beyond their own sensors		10 Hz	3-100 ms	90-99.999%	Short to long

Note 1: Message range is qualitatively described as short for <200 m, medium from 200 to 500 m, and long for >500 m, assuming motorway scenario with absolute vehicle speed of 160 km/h for basic set of services and 130 km/h for advanced services.
Note 2: Indicated ranges of requirements for advanced services cover different automation levels of vehicles.

Fig. 2.4 ITS use cases and communications requirements (based on [7], [20], [21], [47]).

2.2 ITS Use Cases and the Need for V2V Communications

Types of messages

An overview of ITS use cases to be supported by vehicular communications with associated requirements is provided in Fig. 2.4. As introduced in Chapter 1, the basis for most of the traffic safety and efficiency applications is formed via i) local regular broadcast of periodic status messages; and ii) event-triggered messages that indicate hazard warnings. The first type of message is periodically broadcast from vehicles to constantly inform their surroundings about their position, speed, direction, etc., in order to establish cooperative awareness. Cooperative awareness message (CAM) [18] and the basic safety message (BSM) [53] are such messages specified by the ETSI and SAE (the US Society of Automotive Engineers) standards, respectively. The second type of message is triggered upon various events, e.g., to inform vehicles about unexpected conditions, such as road hazards. While ETSI has specified the decentralized environmental notification message (DENM) [19] for this purpose, SAE has specified various of such messages, e.g., emergency vehicle alert (EVA) and traveler information message (TIM), for different purposes, respectively [53].

Rate of messages

Among different types of messages, CAMs are expected to constitute 70% of the traffic load [54]. Generation of CAM is governed by the cooperative awareness basic service, which defines, in particular, the message generation rate. To allow efficient use of the radio resources and prevent channel congestion, the actual generation time interval is in general variable. The time-dependent behavior of the vehicles, such as their speed and changes in their direction, and the fact of being a special vehicle (e.g., an ambulance) influence the message generation rate [18]. To illustrate, vehicles transmit with the minimum packet generation periodicity of 1 Hz while moving at a speed of 14.4 km/h or below, while the maximum packet generation periodicity (i.e., 10 Hz) is reached when they travel at 144 km/h or above. Further, even at the same constant speed, a new message is triggered if a vehicle changes its direction at least for 4°, thus impacting the time interval between two consecutive CAMs. Event-triggered messages are also transmitted in a periodic manner for a limited time duration, with a minimum frequency depending on the use case, including the vehicle speed and transmission range. To illustrate, for a vehicle broadcasting that a safety function (e.g., braking) is out of its normal condition upon detecting it requires at least 1 Hz message frequency, whereas warning the following vehicles of a sudden slowdown of the traffic requires transmission of DENMs with at least 10 Hz [20].

Latency and reliability of messages

Transmitted messages need to be received within a maximum allowed time, referred to as latency. Latency denotes the one-way, end-to-end, maximum tolerable time from the generation of a packet at the source application until it is received by the destination application [21]. The maximum latency requirement is derived from the application operating requirements. While ITS services have a typical latency requirement of 100 ms, advanced use cases involving a high degree of automation can have a latency requirement as stringent as 3 ms.

ITS applications also demand the messages to be received using the communication system with high reliability, which is specified in terms of the probability that the recipient gets the transmitted packet within the specified latency. A commonly used metric to measure radio-layer message reception reliability is *packet reception ratio (PRR)*, also known as *packet delivery ratio (PDR)*, which can be simply expressed as the average ratio between the number of neighbors correctly decoding a packet at a given distance and the total number of neighbors at the same distance[2]. The reliability is in general related to the required latency;

[2] We provide the formal definition of PRR in Section 3.2

the lower the latency requirement of a transmission, the higher the expected reliability. For rural or highway scenarios, minimum reliability is specified as 80% of probability that the recipient gets a message within 100 ms [22] for the basic ITS services. Whereas the advanced services have more diverse and stringent requirements with up to 99.999% reliability and 3 ms of latency. It should be however noted that many of the use cases could also be supported using more lenient reliability or latency values at the cost of less optimal operation (e.g., in the case of platooning, larger distances between vehicles) [7].

Range of message reception

The required reliability of message reception needs to be ensured at a minimum communication range between the transmitter and the receiver to allow timely action by the applications (e.g., considering the driver's reaction time). The range is determined depending on the use case and the velocity of the involved vehicles. For instance, while warning messages in an urban intersection need to be received at a range of 50 m, messages transmitted by vehicles traveling on a highway with absolute speeds at 280 km/h require a range of 320 m [47].

Density of vehicles

Finally, the use cases should operate under an expected density of vehicles, e.g., expressed in terms of the number of vehicles per km^2. This also indicates that multiple vehicles within the same area run the same (and potentially additional) use cases in parallel. For example, at rural intersections, a vehicle density of 1500 vehicles/km^2 is expected, whereas the maximum density in urban scenarios is expected to be 10000 vehicles/km^2 [23]. Besides the type of the area, the actual vehicle density would also depend on the time of the day, weather conditions, traffic congestion, etc. There is also a correlation between vehicle density and speed in general, e.g., the more vehicles are on a road, the slower their speeds are [47].

General remark

Although none of the above ITS use cases demand high requirements in all dimensions simultaneously, when combined, they call for a communications system that is able to support high traffic load, high reliability, low latency, and long range.

The necessity of V2V communications

In the following, we discuss why V2V communications is necessary to address the challenging requirements of ITS applications when compared to other wireless technologies or ways

Table 2.1 Comparison of communication technologies for vehicular communications (derived from [55] and [56]).

Capabilities	**V2V**	**V2N**	**Wi-Fi**	**UWB**	**BT&ZB**	**VLC**	**Radar**	**NTN**	**DTV**
Range (approximate)	1 km	10+ km	1 km	10-30 m	10-100 m	20 m	2 km	0.1-600+ km	40 km
One-way to vehicle	✓	✓	✓	✓	✓	✓	✓	✓	✓
One-way from vehicle	✓	✓	✓	✓	✓	✓	✓		
Two-way	✓	✓	✓	✓	✓	✓			
Point-to-point	✓	✓	✓	✓	✓	✓		✓	
Point-to-multipoint	✓	✓					✓	✓	✓
Latency	0.2-1 ms	1 ms - 3.5 s	3-5 s	200 µs	200 ms	?	?	?	10-30 s

of communication. Table 2.1 summarizes different candidate technologies with respect to their capability to support vehicular communications.

- **V2V communications** intend line-of-sight or near line-of-sight direct communications between vehicular peers in close proximity[3] without needing to connect to a network access point. This enables a two-way unicast, multicast, and broadcast communication from and to vehicles within around a few hundred to thousand meters, depending on the underlying radio technology. As compared to longer-range and/or multi-hop communications, such as cellular V2N or V2N2V, V2V communications offer the advantages of the *hop gain*, *proximity gain*, and *reuse gain* that we introduced in Chapter 1. These gains allow low propagation delay and end-to-end latency, high data rates at low transmission power, and high spectral efficiency. Early works have shown that utilizing direct communications between the users can achieve better performance in terms of capacity, throughput, power efficiency, and spectral utilization compared to infrastructure-based communications using UL/DL (e.g., cf. [57], [58]). Further, the non-necessity of a centralized control enables a flexible organization of communications in an ad hoc manner.

- **Cellular V2N and V2N2V communication** is based on the long-range communications between the user and the BS that can take place across long distances. While microcells can typically cover a radius of 500 m, macrocells can reach a radius of 10 or even 30 km [59]. The communications between vehicle and BS can be carried out in a unicast fashion as traditionally supported by cellular networks. The recent specifications have also introduced the support for broadcast and multicast vehicular traffic on DL, i.e., from BS to vehicles [37]. The cellular network can support high data rates with low-to-medium latency. However, initial field trials presented in [60]

[3] While we focus on the V2V communications, the same also applies to V2I and V2P communications between the vehicles and nearby infrastructural or pedestal nodes.

with a simple setup involving a single BS and three vehicles show that the V2N2V communications yield around two times larger latency than direct V2V communications. Moreover, the simulation results in [61], [62], and [63] show that increasing the vehicular density can further increase the latency by up to two orders of magnitude when unicast DL messages are considered. Yet additional latency is expected due to forwarding and processing at the cellular core network when multiple BSs are involved. Another prominent drawback of V2NV communications is that vehicles need to be under cellular network coverage to communicate, which may not be possible at all times.

- **Wi-Fi** is a medium-range communication technology between the user device and an access point of a local area network (LAN), based on the IEEE 802.11 series of standards. Using Wi-Fi, vehicles can connect to a wireless access point of a LAN within tens to hundreds of meters (e.g., home LAN when garaged), which enables extensive upload and download of non-time-critical data [55]. However, Wi-Fi lacks the support for highly mobile users. In addition, similar to cellular systems where vehicles connect to BSs, Wi-Fi requires the association of vehicles with an access point to communicate, thus requiring them to be within the range of a LAN.

- **Ultra Wide Band (UWB)** is a technology for transmitting data by spreading the radio energy over a very wide frequency band (typically larger than 500 MHz) with a very low power spectral density [59]. While the low power spectral density limits the interference potential with conventional radio systems, the high bandwidth allows very high data throughput. The largest limitation of UWB for vehicle communications, however, is its limited range, which is typically up to a few tens of meters.

- **Bluetooth and Zigbee** are other short-range wireless communication technologies. While they are low cost, low power [44], and can serve V2I communications channels for stationary vehicles in close proximity (e.g., electronic payments at fast food drive-thrus), their operation is limited in terms of range and latency considering mobile and safety-critical communications required by ITS applications [55].

- **Visible light communications (VLC)** make use of visible portion of the electromagnetic spectrum at 430 – 790 THz band to transmit and receive information [64]. Such high frequency enables extreme data rates reaching up to 500 Mbps over relatively short distances of several meters [65], and is not interfered with by the highly crowded parts of the spectrum at lower frequencies. The advent of light-emitting diodes (LEDs) progressed the use of VLC, making several standards already available. Producing light has much lower energy consumption as compared to radio-frequency-based communication devices, and the light sources such as head and tail lights that are already available on the vehicles can be utilized for this purpose. VLC technology, however, is severely limited by several factors including disturbance of the transmission due to flickering lights, susceptibility to weather and ambient conditions, noise and interference coming from irrelevant light sources in the environment, and its inability to operate under non-LOS conditions [64], which are all common in vehicular environments.

- **Radar** (radio detection and ranging) is a technology primarily used for sensing the environment based on the principle of measuring the time of flight between the emitted signal and its received echo, typically operating at the millimeter-wave spectrum. Besides its sensor application as common in vehicular domain, radar can be also used for communication purposes [66]. However, the most limiting factor would be the two-way wireless communication capability and the need for line-of-sight between the transmitter and the receiver. While the messaging capability of radar can allow vehicles to derive useful information from the roadside, any system that can support a realistic data payload is not yet on the development horizon [55].

- **Non-terrestrial networks (NTNs)** offer high-capacity connectivity to their users via satellites, high-altitude platform systems (HAPS), and unmanned aerial vehicles (UAVs), especially in remote areas without terrestrial networks [67]. However, inherently larger link budgets and higher propagation delays associated with aerial networks, the differential delay introduced due to larger service area that impacts the random access procedures, and high Doppler shift between the aerial devices and vehicles may jeopardize safety-critical communications, thereby making the design of these systems not trivial [68]. Last but not least, the operation of NTNs is limited by their coverage, i.e., they can not offer connectivity in tunnels or underground environments, and are highly prone to atmospheric errors such as due to solar activity.

- **Terrestrial Digital Radio and Digital Television (DTV)** enable transmission of digital audio and video signals, respectively, at large communication ranges. While these technologies could potentially use terrestrial datacasting to communicate with vehicles, such applications have generally a broadcast nature, where the same information is sent to all the vehicles at the same time (e.g., announcement of lane closures, detours, malfunctioning traffic signals, etc.). In addition, the regional coverage aspect and one-way (radio station-to-vehicle) nature of these technologies prevent them from meeting wireless communications requirements of the vehicle safety applications considered by ITS [55].

In summary, when compared to other ways of communication, direct, short-range communications between vehicles, i.e., V2V communications, can uniquely meet the basic communications requirements of the applications presented in Section 2.2. Nevertheless, several issues such as service degradation in congested scenarios, difficulty coping with the compromised line of sight, and security problems remain a challenge for V2V communications.

2.3 Radio Resource Allocation for V2V Communications

As introduced in Section 2.1, V2V communications have stringent requirements that need to be satisfied, most significantly in terms of reliability and latency, which are far more demanding than that of traditional communications due to the safety-critical nature of the messages. The requirements need to be satisfied in a very dynamic environment, where vehicles unexpectedly appear and disappear within each other's communication range. Diverse relative speeds among the vehicles further result in varying distances between the transmitters and receivers. Moreover, the communication involves a high load of data traffic due to the transmission of frequent messages by a large number of vehicles, especially in dense situations such as during rush hours or traffic congestion. Achieving highly reliable, low-latency V2V communications within such a challenging environment in a resource-efficient manner depends highly on the approach taken to allocate the radio resources.

In this section, we first survey various possible techniques of radio resource allocation for V2V communications, which could be performed at different levels by different network protocol layers. Next, we present the resource allocation methods specified by the two standardized RATs, namely the IEEE 802.11p and the cellular. We then analyze the performance of the state-of-the-art resource allocation methods, including the standardized ones, by reviewing the works evaluated them. Finally, we elaborate on the intermittent cellular network coverage problem and its impact on the radio resource allocation performance for V2V communications.

2.3.1 Techniques of Radio Resource Allocation

Spectrum Allocation

The radio spectrum is a scarce resource that has to be allocated to many different communication systems and applications. The spectrum allocation policy should also consider

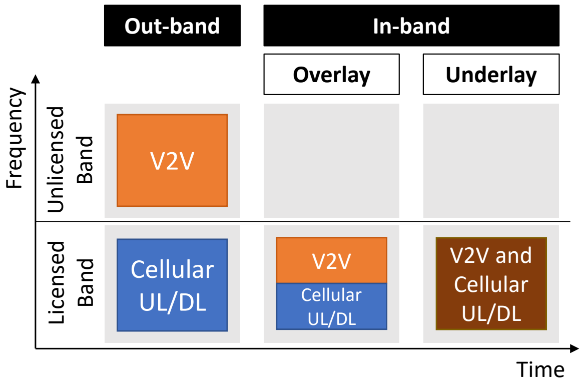


Fig. 2.5 Spectrum utilization options for V2V communications (adapted from [69] ©2014 IEEE and [70] ©2015 IEEE).

technological advances in radios to make spectrum allocation more efficient and flexible. At frequencies in the order of several GHz, wireless radio components come with reasonable size, power consumption, and cost. However, this frequency range of the spectrum has been getting extremely crowded. Therefore, technological developments to enable high-frequency systems would greatly reduce the spectrum shortage with the same cost and performance. Nevertheless, as the frequencies get higher, path loss becomes larger with omnidirectional antennas, thereby limiting the communication range [44].

The spectrum for V2V communications can be allocated in different ways, as illustrated in Fig. 2.5. V2V communications can make use of *licensed bands* or *unlicensed* bands. Licensed bands are assigned by regulatory bodies to specific mobile network operators. Whereas the unlicensed bands are open to any system subject to certain operational requirements. From the cellular networking point of view, the operation of V2V communications using licensed and unlicensed bands are also referred to as *in-band* and *out-band* operation, respectively [69]. In the case of in-band communications, V2V communications can use disjoint subbands from the cellular UL/DL transmissions. Such an approach is referred to as *overlay* V2V communications. On the other hand, the localized nature of direct links between vehicles allows the reuse of the same radio resources used for cellular UL/DL transmissions at the same time. This approach, called *underlay* V2V communications, under favorable conditions, enables the so-called reuse gain, thereby increasing the overall spectral efficiency of the system.

It is also possible that different *RATs* can co-exist at the unlicensed bands. Namely, V2V communications using the cellular and the IEEE standard can share the same spectrum. In this case, additional solutions, e.g., those based on the “listen-before-talk” mechanism, are required to mitigate the negative impact of the mutual interference between different technologies [71]. Related to the spectrum sharing problem, one interesting idea is the notion of

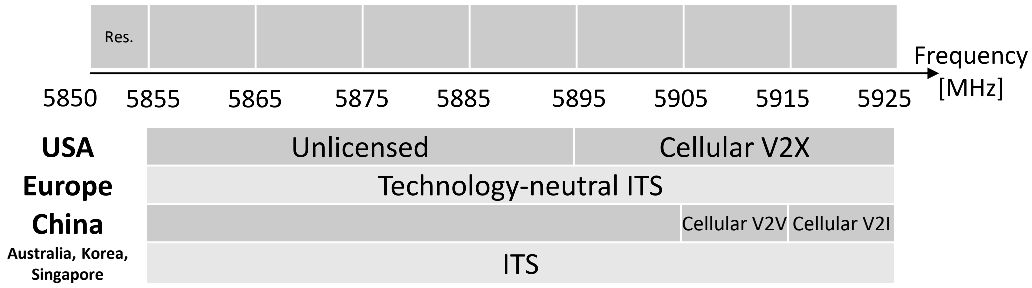


Fig. 2.6 Spectrum designations at 5.9 GHz in various countries (based on [74], [75], [76]).

a *smart* or *cognitive radio*. By sensing its radio environment, this type of radio can determine the frequency, time, and space, as well as other transmission parameters such as modulation and coding [59]. Such an approach would offer employing new frequency bands and opportunities for V2V communications. However, many technology and policy challenges need to be overcome before deploying such a radical spectrum allocation method [56].

For allocating and controlling the radio spectrum usage, most countries have government agencies in charge. In the United States, the spectrum for commercial use is allocated by the Federal Communications Commission (FCC). FCC first time allocated the 75-MHz spectrum in the 5.9-GHz band (5.850-5.925 GHz) to support the DSRC-based ITS applications, in 1999 [72]. This spectrum is divided into seven 10-MHz wide channels, as shown in Figure 2.6. FCC had initially designated the DSRC as the technology standard for ITS services. However, DSRC has not been meaningfully deployed, leaving the critical 5.9 GHz band mostly unused for decades. Based on this, the FCC has decided to transition away from DSRC services. Their recent announcement designates cellular V2X as the new technology standard for safety-related transportation and vehicular communications, due to its recent momentum. For this, FCC has reserved a part of the 5.9 GHz band for the cellular technology to enable substantial deployment of ITS services [73]. The order in November 2020 reallocated the lower 45 MHz of the 5.9 GHz band (5850-5895 MHz) from DSRC to unlicensed use for technologies such as Wi-Fi, and concluded that the United States should move forward with C-V2X in the 5895-5925 MHz portion of the band.

In Europe, spectrum regulations for the frequency range 5855-5925 MHz are technology-neutral and specified for the use of safety-related ITS services by the European Commission Decision in March 2020 [75]. In China, the decision in October 2018 assigned 5905-5925 MHz for V2V and V2I communications in two separate 10-MHz channels, using the cellular standard only. The 5855-5925 MHz band is also allocated to the use of ITS services in various other countries such as Australia, Korea, and Singapore [76]. In Japan, different from other countries, two bands are allocated for ITS at 755.5–764.5 MHz and 5770–5850 MHz, respectively.

While V2V communication using the LTE V2X standard is expected to operate at the 5.9 GHz ITS band, the succeeding 5G NR standard allows the spectrum resources to be also allocated from the licensed bands that are assigned to specific cellular network operators [37].

Admission and Congestion Control

Given the limited spectrum, vehicular networks necessitate control of wireless network resources to provide the required QoS for the ITS services. In vehicular networks, the number of nodes participating in communications is not always known, and more importantly, should not be restricted. Therefore the resource allocation mechanisms must properly scale with the varying vehicular density. While sparse traffic is common in off-peak hours or expected during the early stages of the market deployment, high traffic conditions can be expected during peak hours or in small areas such as urban crossroads. In the latter case, the data traffic can be seriously heavy, especially considering the safety messages such as CAMs (cf. Section 2.2) that are transmitted by all vehicles frequently (typically ten times per s). These circumstances further worsen the collision problem, which leads to applying admission control or congestion control mechanisms.

Admission control (AC) is the task of estimating the state of the network's resources and deciding which application data traffic can be admitted to the resources without requiring more resources than available and without violating previously made QoS guarantees [77]. AC is a standard procedure in cellular networks, which is typically performed by the BSs for the users in the cell they serve. This takes place when a user first time joins the network or hand-offs from another cell [78].

It is possible to perform AC in various ways. While optimal AC schemes are preferable, they are not always achievable, especially in realistic scenarios with large problem sizes and complex system parameter interdependence. As a result, heuristics and intelligent techniques are commonly used to find suboptimal AC schemes. AC schemes can be further classified as proactive (parameter-based) or reactive (measurement-based) [79]. Proactive schemes admit or reject the incoming traffic based on predictive/analytical evaluation of the QoS constraints. Whereas the reactive schemes make the admission based on the QoS measurements following the start of a transmission attempt.

AC targets controlling various communication performance metrics as desired, such as signal quality, traffic blocking or dropping probability, packet-level QoS parameters, and transmission rate [79]. For controlling signaling quality, AC can take interference, i.e., SINR levels, resource loading, effective bandwidth, power allocation feasibility, or transmitted/received power into account. Packet-level QoS parameters include the packet delay or packet dropping rate.

A special case of AC is congestion control, which targets the avoidance of congested communication channels. Congestion control adjusts the communication parameters such as the transmission rate, transmission power, and modulation among others, in order to control the congestion level on the channel and guarantee reliable communications [80]. Congestion control is especially relevant for vehicular communications. The performance challenges associated with high-density traffic are among the major obstacles to the widespread adoption of vehicular communication technologies. Irrespective of the underlying technology, V2V communications can face a congested radio environment due to limited spectrum combined with high vehicular density and frequent message exchange. This is commonly referred to as *scalability problem*. Congestion control techniques play an important role to manage channel load and radio interference, especially when an extension to the allocated frequency spectrum is not possible.

To determine the network congestion, V2V communication technologies usually employ channel utilization-based sensing mechanisms involving several measured metrics. The metrics include channel busy ratio, number of neighbors, and channel occupation ratio [81], as well as message relevance-based assessment (e.g., based on message and vehicle context) [82]. In turn, several parameters are adjusted to control the congestion, based on the employed algorithm. The most considered parameters are the packet generation rate, transmission power, and modulation and coding scheme [81]. Besides adjusting the transmission parameters, vehicles may also simply drop the packet transmission (and/or any re-transmissions) as in the case of admission control [80].

Medium Access Control

In order to allow many vehicles to share a finite amount of radio spectrum simultaneously and benefit from the capacity of the communication channel, the available resources must be divided among the vehicles. This is the responsibility of the *multiple access* schemes or protocols, also referred to as *medium access control (MAC)*.

One straightforward MAC scheme is to partition the channel in time, frequency, code, or space domain (as well as their combinations), and statically dedicate each partition to a vehicle. Such multiple access schemes are referred to as static *time division multiple access* (TDMA), *frequency division multiple access* (FDMA), *code division multiple access* (CDMA), and *space division multiple access* (SDMA), respectively. While this approach would ensure no collisions and uninterrupted transmissions, it is rather suitable for applications with deterministic or continuous stream of data, such as voice or video. In the case of

V2V communications, which mainly consists of periodic and aperiodic messages, the traffic is rather *bursty*. This type of traffic would leave the allocated resources underutilized, hence wasting them.

For applications with bursty traffic, random access strategies are commonly used to efficiently assign the channel to the active users. In random access techniques, also known as “contention-based” MAC schemes, users attempt to access the channel without (or with minimal) coordination. In the simplest approach, users can transmit data packets as soon as they are formed, which is the principle of the *ALOHA* protocol [83]. In case of collisions, packets are retransmitted after a random time. However, in this approach, users can start their packet transmissions at any time, and any partial overlap of two or more packets would damage the successful reception of all packets. Such partial overlap of transmissions could be avoided if all packet transmissions are aligned in time by synchronizing the users, which is the idea behind the slotted ALOHA protocol [84]. In *slotted ALOHA*, the time is divided into slots of a certain duration, and packet transmissions can only take place at the beginning of each time slot. This simply improves the maximum achievable throughput by a factor of two as in an unslotted ALOHA system.

In order to achieve much greater efficiencies of channel usage, MAC protocols that listen to the channel before transmission can be utilized, which exploit the information about the other users. This is the premise behind the *carrier-sense multiple access (CSMA)* [85] mechanisms. CSMA can be interpreted as a “listen before talk” mechanism, where users delay their transmission if they detect that another user is currently transmitting, based on energy-sensing. CSMA can be further combined with a collision avoidance technique (referred to as CSMA/CA), which can be summarized as follows [9]. User that has data to transmit first monitors the wireless channel for activity, through energy detection. If the sensed channel is idle, the user begins the transmission. Otherwise, it performs a random *backoff*, namely waits for a number of time slots before transmission. This backoff timer is also known as *contention window*. The countdown is paused during any non-idle interval. After transmitting, the sender user waits for an acknowledgment (ACK) from the recipient. If it does not receive the ACK within a timeout interval, it re-transmits the data after performing another random backoff. Data sent to a group is not acknowledged and is sent only once.

The most important challenge associated with the sensing-based mechanisms is the well-known *hidden-node problem*, where each node, i.e., vehicle, can only hear its immediate neighbors but no other nodes in the network [44]. This results in simultaneous transmissions coming from vehicles that are not able to sense each other interfering at a receiver located between them. As such, contention-based mechanisms can not guarantee successful transmissions or delay them. This creates problems for V2V communications considering the real-time safety-critical applications [86], [87].

A common way to avoid packet collisions due to hidden nodes is to utilize a handshake before transmission [44]. Such *collision avoidance* is established by the transmitters first indicating a request to send, and transmitting only after they receive an acknowledgment from the intended receiver(s) - known as the RTS/CTS (request to send/clear to send) mechanism. However, this mechanism and similar ones that rely on the receiver's feedback incur additional signaling. Further, they are infeasible for broadcast or multicast transmissions, which constitute most of the vehicular traffic (primarily, safety) since multiple receivers are involved [88], [89], [90].

Random access protocols work well with bursty traffic when there are many more users than available resources yet these users rarely transmit. In the case of long series of packets as in vehicular traffic, random access works poorly because most transmissions result in collisions [44]. In this case, performance can be improved by assigning resources to the users in a more systematic fashion through transmission *scheduling*. For scheduled access, the available spectrum is partitioned into time-, frequency-, code-, or space-division resources, i.e., based on TDMA, FDMA, CDMA, and SDMA, respectively (as well as combinations thereof). Transmission of each user can be scheduled on different resources in a way to avoid conflicts with neighboring users while making the most efficient use of the available resources. In distributed settings, i.e., without a central coordinator, users need to exchange control packets to coordinate among themselves and/or inform each other about the schedule of their transmissions. This however yields a high communication overhead. In addition, even in scheduling-based access protocols, some form of random access is required in distributed settings to make the initial reservation for the subsequent data transmissions.

A collision-free and efficient usage of the resources can be achieved when a centralized entity conducts the scheduling. By gathering information about users' transmissions, resource utilization, etc., the scheduler can efficiently coordinate the resource allocation. In vehicular networks, such a centralized scheduling entity could be an RSU, a cellular BS, or simply one of the mobile vehicular users acting as a cluster head.

Physical Layer Techniques

In addition to the above-mentioned MAC and higher-layer techniques, resource allocation could be also performed at the physical layer via various means. The most prominent ones include the multicarrier modulation, multiple antennas, transmission repetition, and link adaptation techniques.

Multicarrier modulation can achieve efficient usage of the spectrum, which realizes transmissions by dividing the data stream into much narrower subchannels rather than using the wideband channel. Orthogonal frequency-division multiplexing (OFDM) is such a modulation scheme, which encodes the data symbols by modulation onto closely spaced orthogonal subcarrier signals that constitute subchannels [44], [59]. In this way, a high-rate data stream is split into a number of low-rate streams that are transmitted over parallel, narrowband subchannels. The number of substreams is chosen to ensure that each subchannel has a bandwidth less than the coherence bandwidth of the channel, thereby relatively flat fading is experienced by the subchannels. Besides, with the use of guard intervals, OFDM can eliminate the inter-symbol interference on each subchannel. OFDM can be efficiently implemented digitally, and has been adopted by the standardized RATs for V2V communications, i.e., the IEEE 802.11p [11] and the cellular LTE and 5G NR sidelink [37].

The transmitted subchannels in OFDM need not be contiguous, so a large continuous block of the spectrum can be also shared among different users. This creates the possibility to combine OFDM with multiple access using frequency separation, referred to as orthogonal frequency-division multiple access (OFDMA), besides the time and coding separation of the users. The cellular standard for the sidelink utilizes a similar multiple access method called single-carrier frequency-division multiple access (SC-FDMA) [12]. In SC-FDMA, transmissions take place using a single carrier that offers higher transmit power efficiency and reduced cost of the power amplifier as compared to OFDMA.

Another technique to increase resource utilization efficiency is to employ multiple antennas at the transmitter and/or receiver sides. This technique can benefit the communications in three ways [59]: i) diversity gain, where the same information can be transmitted over different antennas, and combined or selected at the receiver side to increase the probability of successful reception by exploiting statistically independent channels; ii) capacity gain via spatial multiplexing, where multiple data streams can be transmitted in parallel using multiple-input multiple-output (MIMO) systems; and iii) SDMA, where multiple users can transmit using the same time and frequency, yet utilizing different directions established via beamforming by different antennas.

Another way to realize diversity at the physical layer is to repeat the transmission (after a period that achieves decorrelation), based on the receiver feedback. This procedure is referred to as automatic repeat request (ARQ) [59]. Alternatively, instead of discarding a corrupt packet, the receiver can store and exploit all it receives, e.g., by combining information from different transmission attempts, for successful decoding. This approach is called hybrid ARQ (HARQ). While these schemes are simple and efficient, they also have several restrictions [59]. ARQ requires the presence of error-detection code, and HARQ may involve additional coded bits to provide a stronger error correction capability, accompanying the data. In addition, a feedback channel is required, which should be also well protected against errors. Further, the time required to receive the feedback, and retransmit if necessary, incurs additional latency in the transmission of data packets.

When the knowledge of the transmission channel, known as the *channel state information (CSI)*, is available, the transmitter can exploit this information to adapt various parameters of its transmission based on the varying link conditions. This technique is known as *link adaptation*, comprising adaptive coding and modulation, and power control. The transmitter can choose the modulation format and coding rate that are best matching to the current situation of the link [59]. When the channel quality is good, e.g., having high SNR, a higher-order modulation, i.e., modulation format requiring less bandwidth can be selected to allow a higher data rate for each user (or allow more users while keeping their rate constant). Higher formats, however, are more sensitive to noise and interference, resulting in larger reuse distance. In OFDM, different modulation or coding can be selected for each subcarrier, which can be also updated in time. A similar adaptive technique can be applied to control transmit power as well. However, while higher transmit power reduces the probability of error among two nodes, this can cause significant interference to other nearby nodes, thereby requiring an optimization considering all nodes in the vicinity. CSI-based adaptive techniques are difficult to realize in practice also due to that they require the channel to be reciprocal, i.e., not changing its state between the time learning the channel and transmitting accordingly. Such an assumption is too optimistic for the case of highly dynamic vehicular networks [91]. In addition, CSI-based approaches become infeasible in the case of broadcast or multicast transmissions.

2.3.2 Resource Allocation in IEEE 802.11p

IEEE 802.11p defines the physical layer and MAC specifications for the WAVE/ITS-G5 standard for V2V communications [11]. IEEE 802.11p protocol is a variant of the 802.11a adapted to the dynamic conditions of vehicular environments. The notable modifications include reducing the data rate for more reliable communications at high speeds, and eliminating the LAN-based "handshaking" to reduce the system latency from seconds to milliseconds.

At the physical layer, the 802.11p protocol uses OFDM with a channel bandwidth of 10 MHz and 8 Modulation and Coding Schemes (MCS), offering a maximum rate of 27 Mbps. At the MAC layer, the 802.11p adopts the Enhanced Distributed Channel Access (EDCA) protocol, which combines the CSMA/CA protocol with support for differentiated services. To control the channel load to avoid situations where the channel load exceeds the total available capacity, Decentralized Congestion Control (DCC) mechanisms as described

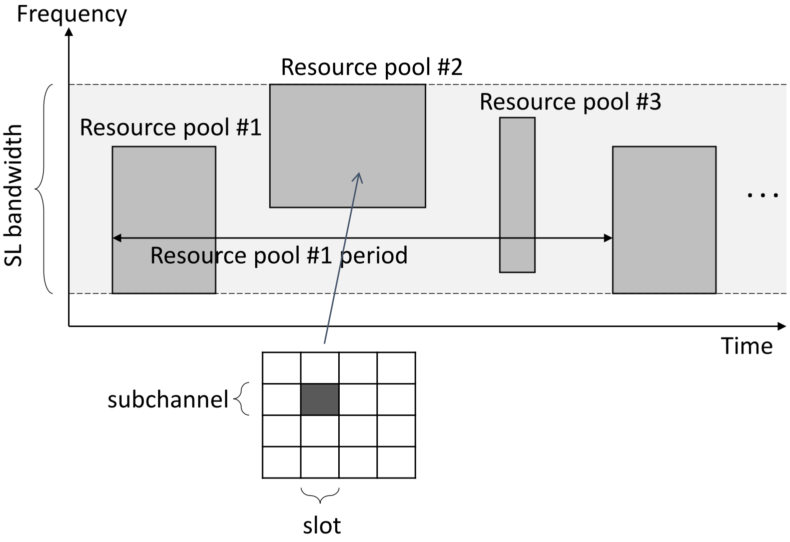


Fig. 2.7 Sidelink (SL) resource pools in the cellular standard for V2V communications (adapted from [37] ©2021 IEEE).

in Section 2.3.1 are introduced. These techniques are designed to meet the requirements of ITS applications, especially in terms of high reliability and low latency for road safety applications. DCC algorithms for the DSRC-based V2V communications are specified by the SAE J2945/1 [92] standard in the US, and the ITS-G5 based V2V communications by the ETSI 102 687 [93] standard in Europe.

IEEE 802.11p has the advantages of easy deployment, low cost, being a mature technology, and its native capability to support V2V communications in ad hod mode. However, it is subject to scalability problems and unbounded delays, and does not provide deterministic QoS guarantees [86]. Besides, due to its short-ranged radio, IEEE 802.11p can only offer limited connectivity. In order to support future vehicular applications that have stringent requirements, the IEEE has been working on a new protocol for V2V communications, named 802.11bd [94]. The main design goals include bringing higher throughput and higher reliability than IEEE 802.11p, together with a larger communication range, and support for speeds up to 250 km/h, mainly by improving the physical layer [95].

2.3.3 Resource Allocation in the Cellular Standard

V2V communications in the cellular networks utilize the sidelink (SL) interface, as mentioned in Section 2.1. SL was first introduced in LTE to support device-to-device (D2D) communications technology targeting public-safety and commercial use cases, with static users in mind. The standard has enhanced the specifications targeting V2V communications to mainly support the high mobility and high user density in vehicular environments.

V2V communications take place within the *resource pools* configured by the cellular network operator for the SL, both in LTE and 5G NR C-V2X[4]. A resource pool is organized into *subchannels* over the frequency domain and *slots* over the time domain, and can be configured in different sizes, as illustrated in Figure 2.7. The cellular network operator can configure different resource pools for transmissions and receptions such as belonging to different V2V applications, different users, or different areas. Further, different resource pools could be (pre-)configured for in-coverage and out-of-coverage users to avoid any interference between them. When multiple transmit pools exist, reception pools can be configured to cover (i.e., overlap with) all transmit pools, so that the users can receive transmissions of other vehicles transmitting in pools different from theirs. In addition, an *exceptional resource pool* can be (pre-)configured for users to utilize upon experiencing exceptional conditions such as radio failure or connection loss to the BS, or handover failures when transitioning between different BSs. Resource pool configurations are provided by the BSs to the users via regular broadcast or dedicated messages on DL. The (pre-)configured information can be provided to the user devices in their integrated circuit cards or as a factory setting as well.

Within a resource pool, the transmission of a packet takes place in units of frequency subchannels and time slots. For each SL data transmission, users also transmit an associated SL control information (SCI) using the SL control channel. The SL control channel shares the same slot as the SL data channel as illustrated in Fig. 2.8. SCI indicates the resources scheduled for the associated SL data, and carries the necessary information to decode it. To receive other vehicles' transmissions, a vehicle monitors only the control channel (rather than the whole resource pool), and determines whether an SCI has been transmitted. After decoding the SCI, the vehicle can then use it to decode the associated SL data.

The cellular standard offers two resource allocation modes to schedule SL transmissions for V2V communications, as mentioned earlier: i) centralized BS-scheduled mode, and ii) distributed user-autonomous scheduled mode. These modes are referred to as *mode 3* and *mode 4* in the LTE standard, respectively; and as *mode 1* and *mode 2*, in the 5G NR standard, respectively. The numbering follows the earlier terminology introduced for device-to-device (D2D) communications in older releases of LTE, as shown in Fig. 2.9.

For the distributed resource allocation mode, the cellular standard does not specify a particular congestion control algorithm but defines the related metrics and possible counter-

[4] Although there exist differences between the LTE and NR standards in terms of the structure of the resource pools (cf. [12]), we stick to their commonalities in this section.

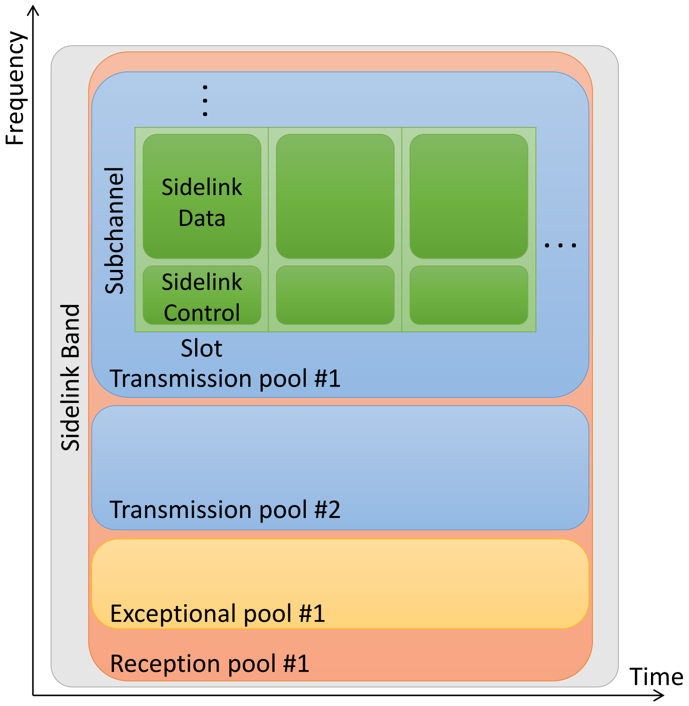
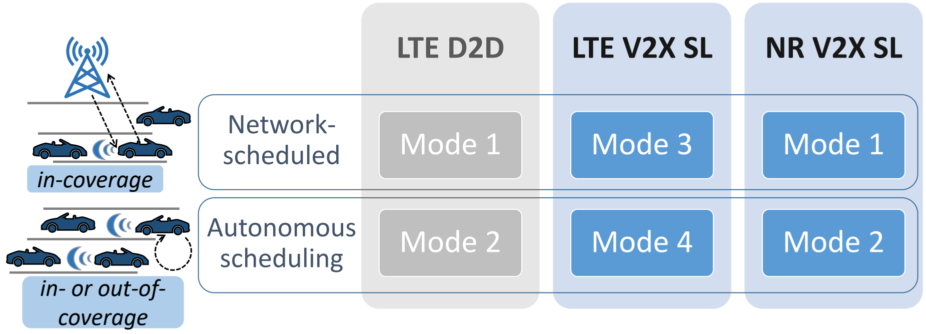


Fig. 2.8 Organization of sidelink resources in the cellular standard for V2V communications.



	LTE D2D	LTE V2X SL	NR V2X SL
Network-scheduled	Mode 1	Mode 3	Mode 1
Autonomous scheduling	Mode 2	Mode 4	Mode 2

Fig. 2.9 Resource allocation modes for sidelink in cellular communications.

measures to reduce the channel congestion in the ETSI specification 103 574 [96]. SAE has been also developing mechanisms under the SAE J3161/1 [97] specifications. Nevertheless, the DCC algorithms specified for the WAVE/ITS-G5 standard by SAE or ETSI can be also utilized for the V2V communications in the cellular standard.

In the following, we provide the details of the centralized and the distributed resource allocation modes. As the distributed mode, we present the LTE mode 4 algorithm, which we use as one of the benchmarks in our evaluations. We here note that the specifications of its successor standard 5G NR mode 2 were not available during the preparation of this thesis work.

Centralized Resource Allocation Mode

In the centralized resource allocation mode, a scheduling entity residing at the cellular BS coordinates the transmissions between the vehicles. For their V2V transmissions, vehicles first send a scheduling request (SR) to the BS over the uplink control channel. The SR indicates the size of the V2V transmission. In turn, a scheduling assignment (SA) sent from the BS to vehicles on the downlink control channel via the *DL control information (DCI)* message indicates the allocated resources (i.e., the subchannels and slots) for that vehicle's transmission. This approach is called *dynamic scheduling*. The centralized mode alternatively supports the allocation of periodically repeating resources (called *semi-persistent scheduling* (SPS) in LTE, and *configured grant* (CG) in 5G NR), for a periodic type of V2V traffic. For this type of scheduling, vehicles report their traffic characteristics to the BS, containing information about the periodicity and size of their messages. In turn, BS configures the vehicles with resources matching their requirements. Multiple SPS (or CG) configurations could be simultaneously activated for a vehicle to support different types of periodic V2V traffic. Despite defining mechanisms for centralized resource allocation, the cellular standard does not specify any particular algorithm for resource allocation, hence leaving it up to the implementation of the cellular network operators or vendors.

Traditionally, the scheduling entity resides at each BS, which allocates resources to the users it serves via a single transmission/reception point. However, recent developments in cellular networks focus on virtualized and splitted architectures, where different layers of the protocol stack are instantiated on different elements located in different parts of the network. In this framework, BSs can be realized with many remote radio heads deployed at different serving locations, which are in turn connected to a unit that is responsible for the MAC functionality, i.e., assignment of time/frequency resources to vehicles. Several such units can be further connected to a central unit that handles upper-layer protocols, such as the allocation of resource pools for different vehicles or areas. Multiple of these flexibly splitted units can be further interfaced with a RAN intelligent controller (RIC) deployed at the edge of the network. RIC can handle near-real-time or non-real-time functionalities, such as session management of vehicles, load balancing across different cells, slicing of network resources for vehicular services, and even training of machine-learning algorithms over data provided by the RAN [98].

Distributed Resource Allocation Mode – the LTE Mode 4 Algorithm

In the distributed resource allocation mode of the cellular standard, vehicles autonomously make semi-persistent resource (re-)selections from the (pre-)configured resource pool for

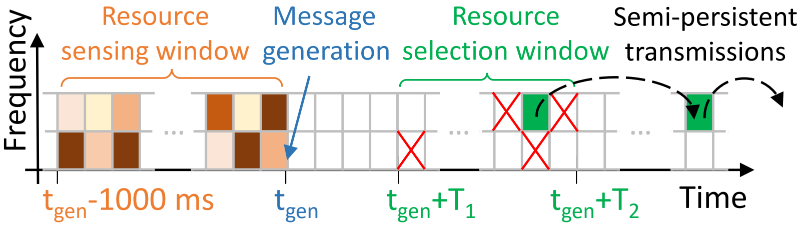


Fig. 2.10 Illustration of distributed scheduling mechanism LTE mode 4. Subchannels sensed with large power (represented with darker brown colors) are excluded from selection (crossed). Selected subchannel (green) among the remaining ones is utilized to transmit semi-persistently [39] ©2022 IEEE.

V2V communications, based on sensing [99], [100]. Specifically, in the LTE mode 4 algorithm, as illustrated in Fig. 2.10, upon a message generation at time t_{gen}, vehicle selects a single subchannel from an upcoming resource selection window between $t_{\text{gen}} + T_1$ and $t_{\text{gen}} + T_2$. The vehicle can transmit using the selected subchannel on a periodic basis for C_{resel} times, i.e., semi-persistently, where C_{resel} takes a random value from a predefined interval. After C_{resel} transmissions, the vehicle makes a new resource re-selection with probability $1 - P_{\text{keep}}$, otherwise keeping the same subchannel by setting a new random value for C_{resel}. Each resource (re-)selection is based on sensing results of the past 1000 ms from t_{gen}, excluding the slots the vehicle transmitted, as no sensing was conducted due to the half-duplex radio constraint. The vehicle further excludes the subchannels where it sensed an average received power larger than a predefined threshold $\text{Thr}_{\text{sense}}$. It sorts the remaining subchannels with respect to their average Received Signal Strength Indicator (RSSI), and selects a subchannel randomly from the top 20% (the lowest RSSI).

The sensing mechanism enables vehicles to find a “free” subchannel, or, in case of heavy resource use, a subchannel with relatively low interference. On the other hand, the randomization aims at mitigating the persistent resource conflicts due to multiple vehicles continuously selecting the same subchannel. Nonetheless, since sensing measurements are limited in time and space, vehicles in mode 4 are prone to the well-known hidden-node problem. To illustrate, if vehicles far apart cannot sense each other’s transmissions and select the same subchannel, their transmissions can interfere on a receiver located between them.

2.4 Performance of Radio Resource Allocation Methods for V2V Communications

It is a difficult task to establish a fair comparison between different resource allocation mechanisms proposed so far in the literature for V2V communications since each of them assumes different architecture, underlying technology, a specific class of applications, and operation under certain scenarios and conditions.[5] Nevertheless, by categorizing them with respect to several aspects, it is possible to analyze and compare certain characteristics of different approaches to resource allocation.

The most compelling approach to radio resource allocation for V2V communications is to have a distributed scheme when compared to centralized or clustered schemes. Distributed schemes, which constitute the majority of the proposed methods in the literature, do not require any central coordinator to manage the channel access and assign resources, and can better adapt to dynamic topology changes without intervention. However, the dynamic assignment of resources to each vehicle in such a distributed and ever-changing environment is a challenging task.

The contention-based methods such as the IEEE 802.11p, work fine under sparse vehicular density. However, they cannot handle dense traffic environments. Transmission collisions are inevitable when the network load is high since vehicles randomly access the channel when they have data to transmit. Such an approach can not guarantee QoS requirements for critical road safety applications, mainly in terms of low latency and high packet delivery rate [104], [105]. Further, broadcast safety messages without handshake and acknowledgment mechanisms increase the collision probability due to hidden nodes. Several techniques have been proposed to improve the scalability of contention-based MAC protocols under heavy load conditions. These are mainly based on adaptively adjusting the transmission parameters such as the minimum contention window and the transmission power control, with the 802.11p standard in mind [106].

Among the contention-free techniques in the literature, TDMA-based approaches have formed a majority, due to several benefits such as not requiring frequency synchronization or code assignment algorithms as in the case of FDMA or CDMA techniques, respectively [87]. While contention-free protocols can provide bounded latency, they require a regular exchange of control messages in order to maintain the schedule and time synchronization among all the vehicles in the network. To illustrate, in order to establish a conflict-free reservation, vehicles

[5] For the reader's reference, among the methods discussed here, detailed reviews of the resource allocation techniques for V2V communications within the framework of the standardized RATs, i.e., the cellular and the IEEE 802.11p could be found in [24], [101], and [102]. Comprehensive reviews of resource allocation mechanisms based on non-cellular technologies can be found in [87] and [103], and in their references.

need to periodically broadcast frame information that contains the slot ID and their states to all their one-hop neighbors, which incurs significant communication overhead, especially in dense scenarios. Further, due to the lack of a central coordinator to schedule time slots, the possibility of access and merging collisions still exists. Various approaches take advantage of the road and traffic flow characteristics to reduce such collisions in a distributed manner by pre-allocating sets of time slots to vehicles based on their location (e.g., road segment, lane), speed, direction, etc. [107], [108], [109]. These, however, often require complex methods to determine certain thresholds over space between vehicles to enable resource use among them.

More recently, hybrid mechanisms are proposed, which combine contention-based and contention-free access by dividing the time into two periods for medium access and slot use, respectively. While offering better performance in network throughput, access delay, and stability, they require careful adjustment of the interval ratio according to the varying vehicle density.

Various distributed resource allocation schemes with similar techniques have been also proposed for the cellular standard to support V2V communications, during or even before the standardization efforts developing the LTE mode 4 algorithm. This line of research was initially based on advancing the prior resource allocation techniques developed for D2D communications in the cellular standard, where the users are assumed to be static or have low mobility. In this context, different sets of users are pre-configured with different *resource pools*, in which they do contention-based resource allocation. In [110], vehicles make use of the position information transmitted by other vehicles, in order to choose the resources from the resource pool for V2V transmissions without any central supervision. In [111], the resource pools are created in a time-orthogonal manner, with respect to orthogonal road traffic crossing the intersections. Further, vehicles perform sensing-based resource selection inside each pool. Similarly, in [112], an additional resource pool is allocated exclusively for vehicles inside the intersections. Authors further consider a highway scenario, where time-orthogonal resources are allocated for equal sections along the road, spatially alternating in the two directions. At the same time, a separate resource pool orthogonal in frequency is used by the vehicles driving in the fast lanes.

Performance of LTE mode 4 is evaluated in numerous works [113], [114], [115], [116], [117], [118], and [119]. These studies have analyzed how the performance of V2V communications varies in terms of reliability and latency, with respect to changing parameters of the mode 4 algorithm such as P_{keep} and $\text{Thr}_{\text{sense}}$, under varying environmental conditions in terms of vehicle density, etc. Remarkably, the results demonstrate that the parameters of the mode 4 algorithm need to be carefully tuned, and a unified parameter configuration needs to be adopted by all vehicles in order to maximize the V2V communication performance, especially in high-density scenarios.

Several works have focused on improving the performance of mode 4. Most of them target the persistent collision problem due to (re-)selection of the same resources by different vehicles. Authors in [120] and [121] propose reserving resources at each resource selection instance and alternately using them to mitigate the collision probability. Other works propose exchange of information among the vehicles, such as channel measurements [122], locations [123], or status and reservation information of resources [124], [125], [126]. Revisions to the sensing mechanism are also proposed in several works, mainly by considering different weighting strategies for selecting the resources as in [127] and [128]. Overall, while proposed extensions to mode 4 enhance the reliability and latency of V2V transmissions, they either require additional signaling or increase resource occupancy, hence making them less efficient from the resource utilization point of view. In addition, the parameters of the extended methods require careful optimization in order to achieve desired performance, which needs to be determined and continually adjusted according to highly dynamic conditions in the environment. This increases their implementation complexity, and reduces the probability of realizing their full benefits in practice. Detailed reviews of the works related to the distributed resource allocation mode in cellular networks for V2V communications can be found in [101] and [102].

The above-mentioned drawbacks and limitations of the distributed resource allocation methods can be avoided by utilizing a hierarchical or centralized topology where the resource allocation is managed by a central node. In VANETs, such an access mechanism can be realized with cluster-based approaches, where vehicles in a small area can be grouped into a cluster, and one vehicle in each cluster is selected as the cluster head to act as a local central entity that coordinates the channel access and assigns resources for the group. Such topology aims to reduce the overhead in the one-hop neighborhood by centralizing the resource allocation function at the cluster head. Related works include [90], [129], [130], as well as [131] that utilizes predefined clusters among the vehicles based on the number of hops between them. Clustered protocols can effectively avoid access collisions, provide fair channel access, and increase throughput via spatial reuse of resources. However, the high mobility of the vehicles affects the stability of the cluster heads, which leads to network problems and performance degradation. The main related challenges are the communication overhead due to the exchange of messages required to elect a cluster head, and to maintain and manage the cluster members in a highly dynamic topology, as well as the inter-cluster interference problem when two or more clusters approach each other [87]. In addition, clustered methods are not suitable for high vehicle density scenarios, where their performance degrades due to the high collision rate caused by the inter-cluster interference problem [87].

Another approach to enable centralized resource allocation is based on the idea of exploiting the presence of RSUs and their large transmission range to coordinate the resources for vehicles in a contention-free way, thus ensuring real-time and reliable delivery of V2V messages. Several TDMA-based MAC protocols that utilize RSUs to assign time slots and disseminate the control information are proposed in [105], [132], [133], which can reduce the channel allocation delay and scheduling overhead. These methods comprise two stages, in which vehicles first send requests to RSU for resources upon entering the coverage of an RSU and receiving its beacon message containing its identity. In turn, RSU allocates a particular non-colliding resource to each vehicle, and broadcasts the allocation map to all vehicles in its area. The proposed methods show clear advantages over distributed scheduling methods, especially in terms of packet loss rate and transmission delay, while using the same amount of resources and achieving less communication overhead and improved fairness for the channel access. However, given that RSUs are fixed nodes,the main challenges in these solutions are therefore the short stay period of vehicles in an RSU region, and the inter-RSU interference in overlapping regions of different RSUs. The latter problem is addressed by the methods proposed in [105] and [134] where separate neighboring RSU areas are allocated with different orthogonal frequencies, and frequencies are reused along the road, thereby achieving high throughput and low access delay.

Cellular networks offer an attractive alternative to sole usage of RSUs for centralized scheduling of V2V communications, thanks to the wide availability of BSs deployed along roads with large area coverage, high capacity, mobility support, and centralized architecture [135]. Early works have evaluated the performance of the centralized scheduling in cellular networks for V2V communications in comparison to the competitive IEEE-based standards that are based on distributed resource allocation, prior to the specification of the distributed scheduler mode 4 in LTE. In [25], the proposed centralized resource allocation method based on LTE is shown to outperform the IEEE 802.11-based distributed scheme in terms of higher spectrum efficiency and lower latency. The effective V2V communication distance can be increased by more than 65% or, for the same relevance distance, the number of supported vehicles can be increased by a factor of six. In [136], impact of different centralized scheduling schemes (called as *sequential* vs. *simultaneous*; the former resembling TDMA solutions based on IEEE 802.11p) and resource allocation policies (based on channel quality feedback; with and without frequency reuse) were evaluated. While the simultaneous scheme significantly reduces the average time required for the exchange of information among the platoon members, reporting of channel quality improves the resource utilization, and frequency reuse ensures high-reliability, low-latency, and high-capacity V2V transmissions that enable very short inter-vehicle distances in the platoon. In [137], authors compare the performance of another proposed centralized scheme based on the cellular standard with a distributed scheduling method that was proposed prior to the standardization of LTE mode 4, besides the IEEE 802.11p. In particular, the proposed method outperforms both distributed methods, where it can achieve a 10% larger packet reception ratio and a 10 times lower update delay than the IEEE 802.11p standard.

Authors in [138] compare the performance of the centralized and distributed scheduling modes in LTE, i.e., mode 3 and mode 4, as well as the IEEE 802.11p, by considering a platoon scenario under different vehicle densities. LTE mode 3 is shown to greatly improve the performance compared to mode 4 and IEEE 802.11p, enabling shorter inter-vehicle gaps of 0.8 m with a guaranteed crash rate $\leq 1\%$ irrespective of the surrounding traffic density, which translates into higher traffic efficiency and safety. This is achieved by higher reliability and shorter latency of the V2V transmissions via efficient resource reuse that cannot be achieved by distributed methods. Despite providing very low latency, 802.11p is shown to suffer from the increasing collisions with the load, and mode 4 is prone to collisions due to re-selection or persistent usage of the same resources by the vehicles in proximity, as also shown in other studies, e.g., in [139] and [140]. In [138], it is also shown that LTE mode 4 can outperform IEEE 802.11p in the case of periodic traffic due to its semi-persistent resource scheduling, whereas the resource re-selection parameters require careful tuning to prevent persisting collisions. Better scaling of mode 4 is due to its channel access mechanism, which is based on semi-persistent transmission, unlike CSMA/CA that is purely contention-based.

State-of-the-art algorithms found in the literature for the centralized scheduling of V2V communications in cellular networks propose heuristics based on locations of vehicles to enable resource reuse [32], [137], [141], [142]. A resource allocation algorithm designed for superior reliability is proposed in [141], called *allocation with Maximum reuse Distance (MD)*. MD uses a simple yet powerful heuristic, which allocates time resources to all vehicles in cyclic order following their positions. Thus, MD tries to maximize the average distance between the vehicles using the same resource, with the goal of minimizing interference. MD is analytically shown to outperform other centralized scheduling algorithms in the literature from [137] and [142], as well as the random resource allocation. Yet, as also the authors indicate, MD is far from practical implementation in reality. The scheduling assignments are required to be sent for *all* vehicles in the environment simultaneously at a time, repeating with the V2V message generation rate, thus leading to impractical processing and signaling overhead. Besides, MD considers only a single resource in the frequency dimension for the assignments (hence could only assign different time resources).

A more practical version of MD is proposed in [32], with a similar name *Maximum Reuse Distance (MRD)*, which also shares the same goal of maximizing the distance between reused resources as MD. When a vehicle requests a resource, MRD finds the time resource that is used by the farthest vehicle with respect to the requesting vehicle, and finds the frequency resource sharing the found time resource, in the same way. In case there remain unused resources in the resource pool, then, instead, MRD assigns one of them randomly. MRD does not rely on heavy processing and signaling as MD, as it schedules vehicles asynchronously, in *multiples* of their message generation periodicity, by sending a single assignment to a single vehicle at a time. MRD is also shown to outperform the same benchmark algorithm in [137].

There are several other centralized resource allocation algorithms in the literature, however, considering impractical assumptions. An early work [143] within the framework of D2D communications in cellular networks proposes an algorithm based on power control, using the channel state information (CSI) of all V2V links, which becomes infeasible in broadcast scenarios. Authors in [144] consider an algorithm based on clustering of vehicles and applying graph-based solutions for a road intersection scenario, again using the channel information on the V2V links reported by the vehicles. Other clustering-based methods proposed in [145] and [146] by the same authors are also challenging in terms of implementation due to their requirement of careful re-forming of clusters as vehicles move, which brings impractical complexity and processing overhead to the scheduler. Applicability of some other cluster-based solutions such as [147] are also limited to unicast or multicast V2V scenarios. There is also a large number of works assuming underlay conditions, i.e., V2V communications using the resources shared with cellular uplink and/or downlink communications, such as [148] and [91], and the others in their references. These studies, however, share the common optimization task of maximizing the sum rate and prioritizing the V2I (or V2N) links, and require at least partial knowledge of CSI consisting of slowly varying parameters (path loss and shadowing).

Alongside the centralized and distributed solutions, there is a limited amount of work in the literature on *hybrid* approaches that combine centralized and distributed methods for scheduling V2V communications. The work in [149] proposes the coexistence of two radio interfaces per vehicle, enabling switching between direct D2D-based communications with centralized resource allocation mode and distributed 802.11p-based communications for latency optimization. A more relevant hybrid approach is proposed by [150], where scheduling of vehicular users by base station is followed by distributed scheduling to reduce the base station loading in terms of signaling and computation. In the distributed mode, vehicular users select resources that are divided into geographical zones, by estimating

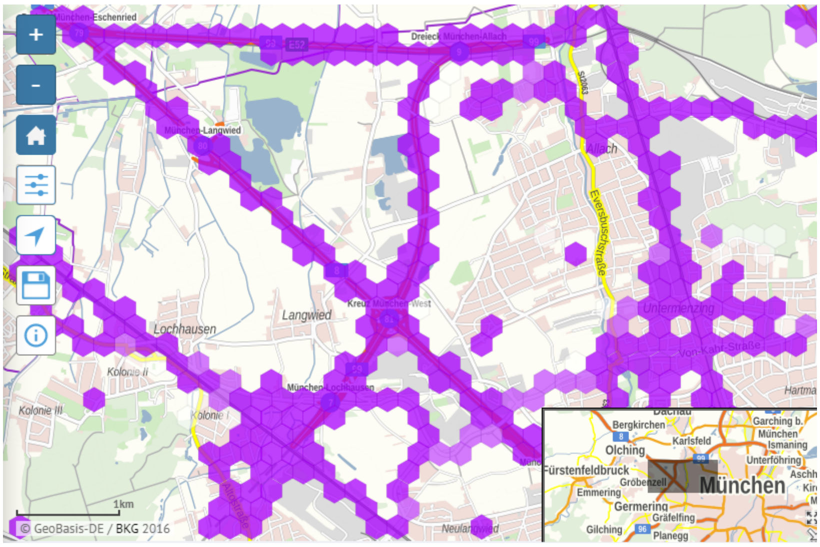


Fig. 2.11 A network coverage map showing parts of roads lacking LTE coverage (areas without purple hexagons) in a region close to Munich, Germany (screenshot from the interactive application in [151]).

co-channel interference and comparing them with threshold values determined by the base station, for the purpose of maximizing link reliability and sum rate.

2.4.1 Intermittent Cellular Network Coverage Problem

Omnipresence of network infrastructure such as RSUs and BSs that can provide centralized resource allocation functionality for V2V communications is not guaranteed. In general, deployment and operation of network infrastructure is challenging due to required network planning, additional costs, and their management and maintenance. In particular, for vehicular networks, ubiquitous presence of an infrastructure surrounding all vehicles at all times is practically inconceivable [152]. Despite any deployment, vehicles may still travel through areas where connection to the network infrastructure is physically impeded, such as inside tunnels. More inevitably, sudden changes on the wireless channel conditions (e.g., deep fading due to blocking objects) are also possible, resulting from highly dynamic vehicular environments, which impact the connectivity to the network infrastructure.

Coverage degradation may arise due to multiple reasons [28], among which we are interested in the two following typical ones in this thesis:

- **Expected coverage interruption** happens in certain pre-known areas such as road sections without network infrastructure deployment, e.g., road tunnels, often referred to as *coverage gaps* or *holes* in the network. Upon entering such areas, users completely lose the connectivity to the network infrastructure, prohibiting transmission of any type of signal on both ways. Network operators are typically aware of these areas.

- **Unexpected coverage interruption** happens due to unexpected conditions in the propagation environment under the presence of the network infrastructure. It could be caused by a sudden blockage of the link between the users and the network infrastructure due to the motion of vehicles or other objects in their surroundings. Additionally, a malicious attack or unintended interference may also avoid successful transmission of data between the users and the network infrastructure, which we however leave out of the scope of our study. Unexpected coverage interruptions are therefore rather temporary, local, yet unpredictable as compared to expected coverage interruptions.

From the perspective of resource allocation for V2V communications, when vehicles lose the connectivity to the central coordinator located at the network infrastructure, centralized resource allocation mechanisms will fail, and vehicles will have to resort to distributed mechanisms. Depending on the network conditions such as availability of the resources and the traffic demand, combined with the characteristics of the interruption (e.g., duration, location, etc.), performance of the V2V transmissions could get severely affected, leading to deterioration in continuity and reliability of services. Given that the QoS requirements of V2V applications need to be satisfied irrespective of network coverage [12], intermittent coverage poses one of the key problems that need to be addressed in V2V communications. This important problem has unfortunately not been treated much in the literature.

In cellular networks, the intermittent coverage problem is expected as one of the key problems, where the links between the BSs and vehicles get interrupted. The problem could arise during early deployment stages due to lack of BSs supporting LTE or NR C-V2X communications, or due to insufficient deployment to cover the entire roadway. To illustrate the actual deployment problem with regards to availability of cellular networks, in the UK, only 66% of the main roads is covered by all 4G (LTE) operators in 2021 [153]. Similarly, users in Germany spent on average 77% of their time with a 4G or better connection in 2019, otherwise connecting to 3G and 2G networks [154]. The analysis covers all network operators in 401 rural and urban districts where the users are most commonly frequent. Fig. 2.11 taken from an interactive coverage-map published by the German Federal Network Agency exemplifies a geographical area around the city of Munich where parts of the roads lack LTE coverage provided by one of the network operators [151]. The problem is much worse in the case of 5G, which is being newly deployed [155]. In the US, 51-92% of the interstate highways are covered by 5G, depending on the operator, according to an analysis in 2021 [156]. Whereas in Germany, 5G is reported in 2022 to be available only 18% of the time across the motorways [157]. Besides the limited deployment of cellular network infrastructure, the deployment of the C-V2X related protocols and functionalities are only at a primeval stage [158], [159]. Field trials in [160] demonstrate how the quality of communications between the vehicles and the network degrade when vehicles travel at locations far from the BS antenna site, i.e., at "cell edge" areas. Such conditions are shown to degrade the performance of vehicular applications using V2N communications.

As per the cellular standard specifications for V2V communications [12], in the case of connection interruptions to the BS, which controls the resource allocation, vehicles need to switch their resource allocation mechanism. Specifically, upon experiencing the connection interruption long and often enough (e.g., up to a configurable timer), vehicles have to stop using their BS-scheduled resources and resort to a random resource selection procedure using the exceptional resource pool. Vehicles then switch to sensing-based resource selection mechanism (mode 4 in LTE or mode 2 in NR) after they start sensing and collect observations for sufficiently long time (1000 ms). The standard does not offer any other mechanisms to tackle the intermittent coverage, e.g., in case of expected interruptions.

A preliminary analysis of the mode switching procedure is conducted in [161]. Besides the case of forced switching, as in the case of unexpected loss of the cellular BS coverage, authors take account of different switching strategies. The proposed switching decisions are based on cost functions considering the load conditions of the cell or the SINR of the signal transmitted from the BS and received at the vehicle. Authors further decompose the mode switching procedure into subsequent stages, and analyze the impact of different strategies on the switching latency. The same authors evaluate impact of mode switching on the reliability of V2V packet transmissions in [162], in a scenario considering a highway with adjacent regions of cellular coverage and non-coverage. In the case of high traffic load, successful packet reception probability goes down from 90% to 60% when vehicles switch from the centralized to the distributed resource allocation (i.e., from mode 3 to mode 4) using the same amount of radio resources in the considered scenario. In another study by the same authors [163], latency of the switching procedure is evaluated to be in the range of $100-150$ ms depending on the traffic density, and packet reception probability is shown to deteriorate by nearly 50% in the worst case in a realistic road traffic scenario with a highway tunnel.

2.5 Reinforcement Learning

In this section, we review the fundamentals of reinforcement learning (RL) and deep learning by taking the sources [164] and [165] as a reference, respectively, and following their notation. After accommodating the necessary background, we provide a taxonomy of RL algorithms.

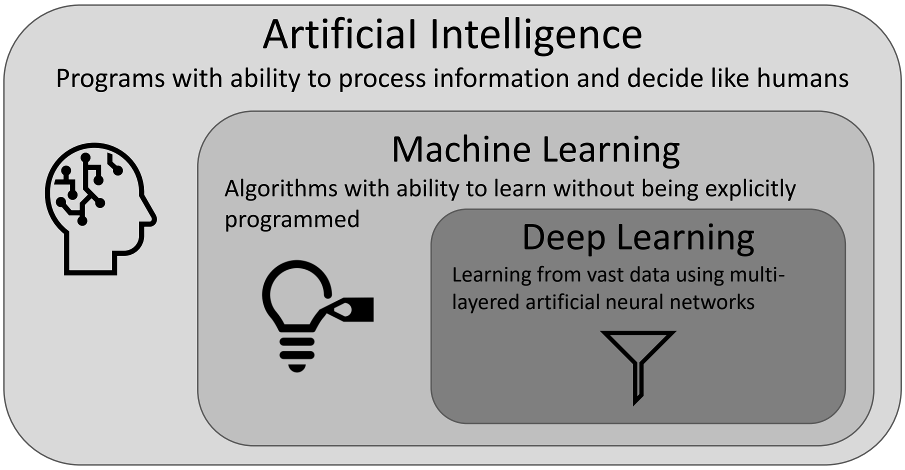

Artificial Intelligence
Programs with ability to process information and decide like humans
Machine Learning
Algorithms with ability to learn without being explicitly programmed
Deep Learning
Learning from vast data using multi-layered artificial neural networks

Fig. 2.12 Deep learning as a part of broader concept of machine learning within the context of artificial intelligence (adapted from [165]).

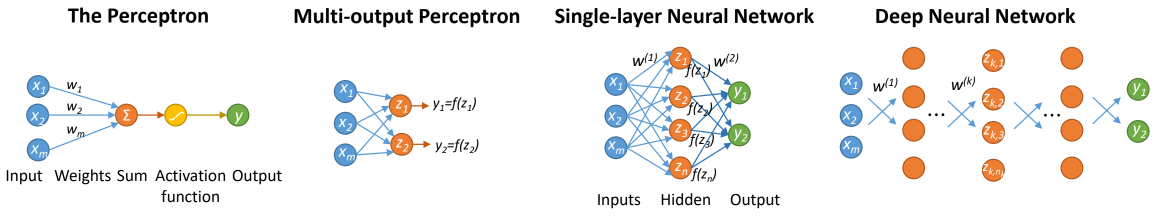
Finally, we present the specific RL algorithm called *Asynchronous Advantage Actor-Critic (A3C)*, which we consider in our proposed methods.

RL is a computational approach to the overall problem of learning and decision-making to achieve goals [164]. Different from other computational approaches, RL captures the paradigm of learning found in nature, where an animal explores its environment and interacts with it in order to gather food and other rewards. With that, RL puts an emphasis on learning performed by a goal-directed *agent* based on direct interaction with its *environment*, i.e., from *experience*, without requiring exemplary supervision or complete models of the environment. Interactions of the agent with the environment *reinforce* (or inhibit) particular patterns of behavior depending on the resulting *reward* (or penalty).

RL has recently led to exciting achievements that were previously out of reach for a machine. The applications span a wide range from self-driving vehicles to robotics, from healthcare applications to financial investments. The recent revolution of RL to solve complex and diverse tasks across numerous domains is thanks to its combination with *deep learning*, where RL utilizes *deep artificial neural networks*.

2.5.1 Deep Learning

Deep learning is a subset of *machine learning (ML)* methods within the overall context of *artificial intelligence (AI)*, as represented in Fig. 2.12. Deep learning utilizes multi-layered artificial neural networks to automatically extract useful patterns or features in raw data, and uses them to learn to perform the task [166]. Traditional ML approaches use hand-engineered features to perform a task. Such an approach is time-consuming, fragile, and not



$$y = f\left(\sum_{i=1}^{m} x_i w_i\right) \qquad z_i = \sum_{j=1}^{m} x_j w_{j,i} \; y_i = f(z_i) \qquad z_i = \sum_{j=1}^{m} x_j w_{j,i}^{(1)} y_i = f\left(\sum_{j=1}^{n} g(z_j) w_{j,i}^{(2)}\right) \qquad z_{k,i} = \sum_{j=1}^{n_{k-1}} g(z_{k-1,j}) w_{j,i}^{(k)}$$

Fig. 2.13 From a perceptron to a deep neural network (adapted from [165]).

scalable in practice, while highly relying on engineering skills and domain expertise. The key advantage of deep learning is to avoid this by automatically using a general-purpose learning procedure, and learning underlying features directly from data in a hierarchical manner. The theory of deep learning dates way back to the 1950s. However, its application has recently experienced a revival due to the greater availability of data, processing hardware, and open-source software tools such as *TensorFlow* [167].

Deep Neural Networks

The fundamental building block of deep learning is just a single artificial neuron, also known as a *perceptron*. Neural networks are composed by stacking and layering perceptrons. As illustrated in Fig. 2.13, perceptron takes a set of inputs $x_1,...,x_m$, each multiplied with a corresponding real-valued weight w, and adds them together. The result is then passed through a *non-linear activation function* f that produces the final output y. The purpose of activation functions is to introduce nonlinearities into the network to deal with nonlinearities in data, as illustrated in Fig. 2.14, which is extremely important in real-life applications. Common examples of non-linear activation functions include the *sigmoid*, *hyperbolic tangent (*tanh*)*, and *rectified linear unit (ReLU)* functions.

Multiple perceptrons could be stacked together to create a *multi-output neural network*, as shown in Fig. 2.13. Since the input is densely connected to every weight of all perceptrons, this structure is also called *dense layer* or *fully connected layer*. A *single-layered neural network* further contains a single *hidden layer* between its inputs and outputs, as shown in Fig. 2.13. Different from the input and output layers, hidden layers are typically unobserved and not strictly enforced either, thus called hidden. The example in Fig. 2.13 is a generic *feedforward* neural network architecture. The term *neural network architecture* refers to how the neurons, i.e., perceptrons, are connected to each other.Finally, a *deep neural network* is composed by stacking multiple hidden layers, as illustrated in Fig. 2.13.

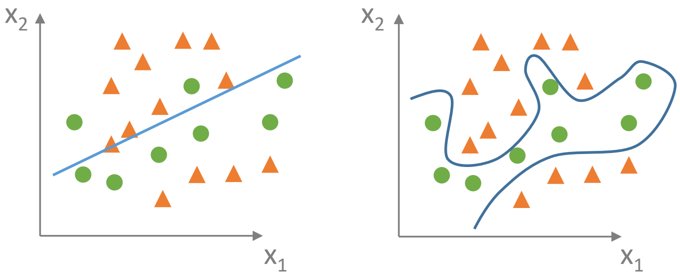


Fig. 2.14 Linear (left) and non-linear (right) activation functions to create boundaries for classifying task of a scattered data (adapted from [165]). While linear functions can only produce linear decisions no matter the network size, non-linearities allow the network to approximate arbitrarily complex functions.

Without hidden layers, a neural network can represent only a very small fraction of possible input-output functions. However, a neural network with a single hidden layer with a sufficient number of perceptrons can approximate any continuous function. This is referred to as *universal approximation* property of one-hidden-layer neural networks [168]. Yet, both experience and theory show that approximating the complex functions needed for many AI tasks is made easier by deep neural networks with many hidden layers. The successive layers in a deep neural network increasingly abstract the representations of the "raw" input. Each perceptron in the network provides a feature contributing to a hierarchical representation of the overall input-output function of the network [164].

Convolutional Neural Networks

Deep convolutional neural network (CNN) is a type of neural network that has proven to be very successful in practice, including the above-mentioned impressive applications performed by RL. This type of network is distinguished for processing high-dimensional data arranged in arrays, such as images. Convolutional neural networks are advantageous over traditional fully-connected networks, by overcoming their limitations in several aspects. First, fully connected neural networks can only take one-dimensional and fixed-size inputs, thus requiring down-sizing and flattening of higher-dimensional data. Such processing, in the case of images, reduces two-dimensional pixel values into one dimension, thereby losing any spatial information [169]. The idea of the convolutional neural network is to connect *patches* of multi-dimensional input data to the hidden layer neurons. Each neuron is connected to a *region* of the input, and the spatial structure is preserved by using a sliding window to define connections [165]. In order to extract particular features, convolution neural network applies a *set* of weights, i.e., "filters" to a given patch locally. The input is element-wise multiplied with the filter weights and the output is summed, followed by shifting the patch [165]. This

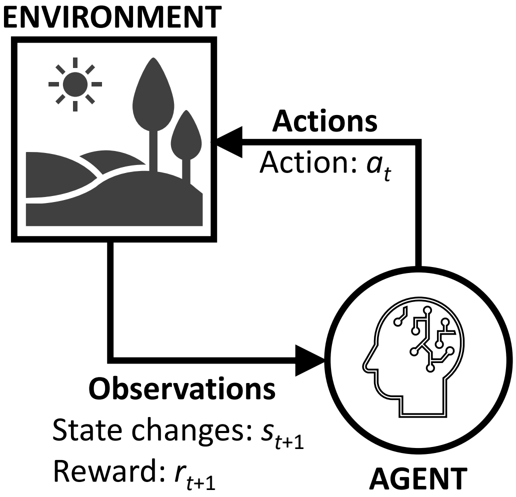


Fig. 2.15 Key components of reinforcement learning (based on [170]).

corresponds to the convolution operation, hence giving the convolutional neural network its name. Second, in the case of a large number of variables in data, a fully connected network requires a much larger number of weights to be trained. Typically, an image with several hundreds of pixels would already lead to several tens of thousands of weights in the case of a fully connected network with a hundred neurons in its first hidden layer [169]. On the other hand, in convolutional neural networks, multiple filters are applied, while the weights of each filter are shared spatially.

2.5.2 Key Components of Reinforcement Learning

In the core of an RL algorithm lies the ***agent***, which is the learner and decision maker. Agent interacts with its ***environment***, which comprises everything outside the agent. During the interaction, the agent selects ***actions*** and environment responds to them presenting new situations to the agent, as illustrated in Fig. 2.15. The actions also give rise to ***rewards*** from the environment, which the agent tries to maximize over time by choice of its actions. Everything inside the agent is known and controllable, while its environment is incompletely controllable and may not be completely known.

Specifically, the agent and the environment interact over a sequence of time steps $t = 0, 1, 2, ...$ At each time step t, the agent receives some representation of the environment's ***state*** s_t, which is a concrete and immediate situation in which the agent finds itself. Based on s_t, the agent selects an action a_t from the set of possible actions that the agent can take in the environment. One step later, in part of its action's consequence, the agent receives a numerical reward r_{t+1} and finds itself in a new state s_{t+1} of the environment. This gives a rise to sequence of state-action-rewards as $s_0, a_0, r_1, s_1, a_1, r_2, s_2, a_2, r_3, ...$. The time steps do not need to correspond to fixed intervals of real time, but rather can refer to arbitrary successive stages of decision making and acting. The agent's goal is to maximize the total amount of reward it receives, which is in general expressed by a specific function of the reward sequence, called ***return*** R_t. In the simplest case, the return is the sum of the future rewards

$$R_t = r_{t+1} + r_{t+2} + r_{t+3} + ... + r_T, \tag{2.1}$$

starting from time step t until the final one T, considering an *episodic task*. In episodic tasks, the agent-environment interaction breaks into natural parts, such as plays of a game, trips through a maze, or any other type of recurring interactions. It is also common to consider return as a *discounted sum of rewards* as

$$R_t = \sum_{i=t} \gamma^i r_{i+1}, \tag{2.2}$$

by multiplying future rewards with a discount factor $0 \leq \gamma \leq 1$. γ is used to adjust the importance the agent pays to immediate rewards in comparison to future ones. Note that the returns at successive time steps hold the following recursive property, which is important to RL algorithms:

$$R_t = r_{t+1} + \gamma R_{t+1}. \tag{2.3}$$

The agent's behaviour is formally expressed by a ***policy*** π, which is a mapping from the states of the environment it perceives to the actions to be taken when in those states. A stochastic policy $\pi(a|s)$ is expressed as the probability of selecting action $a_t = a$ given a state $s_t = s$. The reward signal founds the primary basis for altering a policy: if a selected action leads to a low reward, then the policy may be changed in order to select another action in the same situation in the future. On the other hand, actions do not only influence immediate rewards, but also subsequent situations, i.e., states, and through them, the future rewards. Namely, actions may have long-term consequences, involving delayed reward. Thus, a correct action choice requires foresight or planning. It may be better to sacrifice immediate reward to gain more long-term reward.

While reward signal reflects what is good or bad in an immediate sense, a ***value function*** specifies what is good in the long run to the agent. The value $v_\pi(s)$ of a state s is defined as the total amount of reward an agent can expect to accumulate over the future, i.e., the expected return, starting from that state, using the agent's current policy π. It is denoted as

$v_\pi(s) = \mathbb{E}[R_t|s_t = s]$ and v_π is called *state-value function for policy* π. Similarly, value of taking action a in state s under a policy π is denoted as $q_\pi(s,a) = \mathbb{E}[R_t|s_t = s, a_t = a]$ and q_π is called as *action-value function for policy* π. Values indicate the long-term desirability of a given state by taking into account the subsequent states and rewards that are likely to follow.

Another component of some RL systems is a ***model*** of the environment. The model is defined by the *dynamics function* $p(s', r|s,a) = \Pr\{s_t = s', r_t = r|s_{t-1} = s, a_{t-1} = a\}$ of the environment, which consists of probability of a state and reward occurring at time t, given particular values of the preceding state and action, for all s', s, r, and a. Namely, given a state and an action, the model can predict the resultant next state and reward. Such well-defined formulation of transition probabilities constitutes a *Markov decision process (MDP)*, which is a mathematically idealized form of the RL problem, enabling precise theoretical statements to be made. Optimal policies and value functions for finite MDPs with complete knowledge can be reliably computed by *Bellman equations* or using *dynamic programming* methods such as *policy iteration*. However, in practice, learning an optimal policy rarely happens. In most of the RL problems either the MDP model is not known or it is too large to be practically utilized by the agent. In particular, extensive memory and computation per time step would be required to create accurate approximations of value functions, policies, and models. In cases when a problem has the bottleneck of constructing an accurate model of the environment, model-free methods become advantageous.

Model-free agents estimate the value functions v_π and q_π from experience. As an example, if the agent follows policy π and maintains an average of returns that it encounters for each state (or each state-action pair), then the average will converge to the state's actual value $v_\pi(s)$ (or $q_\pi(s,a)$) as the number of times the agent's visit per state (and action) approaches to infinity. For a task with small number of states, it is possible to form such approximations by using arrays or tables with each entry corresponding to each state (or action-state pair). Such RL methods are called *tabular methods*. However, majority of practical cases of interest has a large number of states, where keeping a table becomes not feasible. Besides an extensive memory required for such table, any encountered state will likely to have been never seen before. It would be therefore necessary to *generalize* from previous encounters with states that are somehow similar to the unseen ones. *Function approximation* is such technique enabling to construct a function by taking examples from it. Using this method, referred to as *approximate solution method*, agents could learn to maintain v_π and q_π as *parametrized functions*, with number of parameters much less than the number of states. As an example, a linear function of state features could be used for this purpose, with parameters being the weights of each feature. More generally, the function could be computed by a deep neural network, with parameters being the connection weights of the network.

In the following, we review the methods that estimate the value function without any model of the environment, namely the model-free prediction, which forms the basis of the RL algorithms provided later in Section 2.5.3.

Model-free prediction

The simplest idea to estimate the value function is the so-called *Monte Carlo (MC)* learning. MC estimates the expected return of a state by its empirical mean, that is, by averaging the return over samples or experiences collected upon visiting that particular state through multiple episodes. After each episode, the value function is updated to estimate the mean, in an incremental fashion as

$$v(s_t) \leftarrow v(s_t) + \alpha(R_t - v(s_t)) \tag{2.4}$$

(in the simplest case $\alpha = 1/N(s_t)$, where $N(s_t)$ is the number of visits), thus moving the mean estimate in the direction of error. As a limitation, MC works only with episodic problems, and requires the agent to reach to the end of the episode to collect the actual return R_t before making any updates.

An alternative method to MC learning is *temporal-difference (TD)* learning, which can learn from incomplete episodes, with the concept of *bootstrapping*. Bootstrapping substitutes the actual return with its existing estimate as:

$$\begin{aligned} v_\pi(s) &= \mathbb{E}[R_t | s_t = s] && \text{(from definition)} && (2.5) \\ &= \mathbb{E}[r_t + \gamma R_{t+1} | s_t = s] && \text{(from Eq. 2.3)} && (2.6) \\ &= \mathbb{E}[r_t + \gamma v_\pi(s_{t+1}) | s_t = s]. && && (2.7) \end{aligned}$$

Update of the value function with TD learning thus takes the form

$$v(s_t) \leftarrow v(s_t) + \alpha(r_{t+1} + \gamma v(s_{t+1}) - v(s_t)), \tag{2.8}$$

where $r_{t+1} + \gamma v(s_{t+1})$ is called the *TD target*, and $\delta_t = r_{t+1} + \gamma v(s_{t+1}) - v(s_t)$ is called the *TD error*. With such updates, TD can learn *online*, i.e., step-by-step, as opposed to episode-by-episode learning by MC. On the other hand, TD target is biased estimate of $v_\pi(s_t)$ as compared to actual return R_t used by MC, which is unbiased estimate of $v_\pi(s_t)$. Yet, TD target has much lower variance than the return, as it contains a single random action-transition-reward tuple as compared to many random actions, transitions and rewards in the return. The idea of TD learning could be generalized as *n-step* predictions, that is, the estimate of the value function after n steps could be used to update the original value

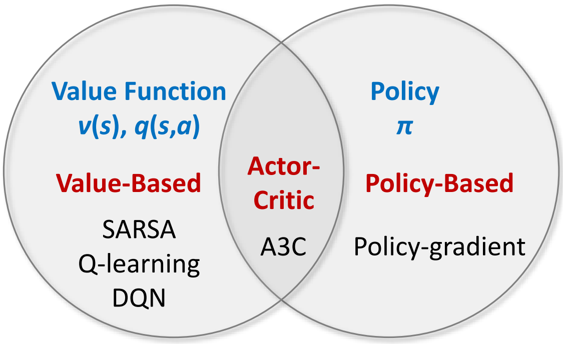


Fig. 2.16 Types of reinforcement learning algorithms (adapted from [170]).

function as

$$v(s_t) \leftarrow v(s_t) + \alpha(R_t^{(n)} - v(s_t)), \tag{2.9}$$

where $R_t^{(n)} = r_{t+1} + \gamma r_{t+2} + ... + \gamma^{n-1} r_{t+n} + \gamma^n v(s_{t+n})$ is the n-step return. Given this formulation, n-step methods span a spectrum with TD methods at one end with $n = 1$, and MC methods at the other end with $n = \infty$.

2.5.3 Reinforcement Learning Algorithms

RL algorithms can be classified based on whether they have a value function that they represent, whether they have a policy that they represent, or whether they have both, as illustrated by the Venn diagram in Fig. 2.16 [170].

Value-based Methods

The agents that learn values of actions and select actions based on their estimated values are called *value-based* agents. Given the estimated value function, the simplest action selection rule is to select the action with the highest estimated value, that is the *greedy* action with respect to the value function.

Greedy action selection exploits current knowledge to maximize immediate reward, however does not try seemingly poor actions at all to see if they might really be better. A simple alternative is to select a random action every once in a while, i.e., with small probability ε. This allows the agent to randomly *explore* states that it might otherwise never see. Methods using this near-greedy action selection rule are called *ε-greedy methods*. An advantage of these methods is that, in the limit where the number of actions goes to infinity, each action will be sampled infinitely number of times, thus ensuring the estimate $q(s,a)$ to converge. However, such asymptotic property does not guarantee a practical effectiveness in practice. The need to balance *exploration vs exploitation* arises as a distinctive challenge of RL.

Value-based methods involving $v(s)$ requires the model of the MDP, as one needs to explicitly estimate the value of each action to suggest a policy. If the model is not known, then the alternative would be to estimate action-values $q(s,a)$ rather than state values, and make the greedy policy improvement over them. The method that applies TD learning to estimate $q(s,a)$ is called *SARSA*, as an acronym for state-action-reward-state-action, indicating the updates based on a transition from state-action pair to state-action pair as:

$$q(s_t,a_t) \leftarrow q(s_t,a_t) + \alpha(r_{t+1} + \gamma q(s_{t+1},a_{t+1}) - q(s_t,a_t)). \tag{2.10}$$

Another type of value-based RL method is *Q-learning*, which uses the following updates in its simplest form:

$$q(s_t,a_t) \leftarrow q(s_t,a_t) + \alpha(r_{t+1} + \gamma \max_a q(s_{t+1},a) - q(s_t,a_t)). \tag{2.11}$$

Q-learning is exactly like SARSA, except it does not follow the same policy it uses to select actions when evaluating the target to update the value function, but considers an action choice in a greedy fashion by simply taking the maximum value of q over it. Instead of using tabular methods, the value function can be represented as $q(s,a;w)$ using a function approximator with a set of parameters w, such as a deep neural network. Function approximation is a powerful, scalable way of generalizing from a state-space much larger than the memory and computational resources. *Deep Q-network (DQN)* is such RL agent proposed in [171] and [172], which yielded the impressive result of achieving beyond human-level control in famous Atari video games.

While value-based methods are shown to learn super-human policies, they exhibit several very important downsides, even on relatively simple tasks. First, the agent can only handle discrete action spaces and is usually successful in small action spaces. Second, since policy in value-based methods is deterministically (or ε-greedily) computed from the maximization of the value function, it can not learn policies that can be stochastic, i.e., changing according to some probability distribution. A typical example is the rock-paper-scissors game, where a deterministic policy can be easily exploited, while a uniform random policy is optimal [170]. Third, in some problems, it could be more convenient to represent a policy than the value function, where the value function is complicated to approximate whereas the policy could be more compact.

To address these challenges, one could instead resort to *policy-based* methods. Policy-based methods directly optimize the policy that governs the action selection instead of relying on the value function. This idea constitutes the basis of *policy gradient methods*, which can: i) learn specific probabilities for taking the actions, thus learn a stochastic policy; ii) learn appropriate levels of exploration and approach deterministic policies asymptotically; and iii) naturally handle continuous action spaces [164].

Policy Gradient Methods

Policy gradient methods learn a parameterized policy $\pi(a|s,\theta) = \Pr\{a_t = a|s_t = s, \theta_t = \theta\}$ that predicts the probability distribution of actions given a current state s, with parameters θ. The action selection could be, for example, according to an exponential *softmax* distribution over parameterized numerical preferences $h(s,a,\theta)$ formed for each state-action pair:

$$\pi(a|s,\theta) = \frac{e^{h(s,a,\theta)}}{\sum_b e^{h(s,b,\theta)}}, \tag{2.12}$$

where the denominator normalizes the action probabilities in each state to sum to unity. The action preferences themselves could be parametrized arbitrarily, such as using a deep neural network, where θ is the vector of all connection weights of the network.

Policy parametrization with softmax in action preferences has the following advantages as compared to ε-greedy action selection over action values as done by the value-based methods. While there is always a non-zero (i.e., ε) probability of selecting a random action in ε-greedy approach, approximate policies can approach a deterministic policy. Namely, if the optimal policy is deterministic, then the selection probabilities of optimal actions will be driven infinitely larger than the suboptimal actions. The second advantage is that, parameterized policies with softmax in action preferences also enable selection of actions with arbitrary probabilities. This allows action preferences to produce optimal stochastic policy (if permitted by the parameterization) in case of problems where the best approximate policy is non-deterministic (e.g., the rock-paper-scissors game).

In policy gradient methods, the RL agent aims at decreasing the probability of actions that result in low reward, while increasing the probability of actions resulting in high reward. The loss function of training policy gradient algorithms takes the form

$$-R_t \ln \Pr\{a_t = a|s_t = s, \theta_t\}, \tag{2.13}$$

namely the log-likelihood of selecting an action given a state, multiplied by the return. The updates to the policy parameters take the gradient descent that minimizes the loss function,

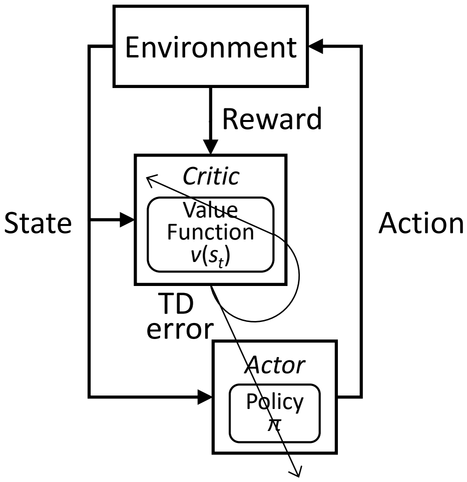


Fig. 2.17 Actor-critic architecture (based on [164]).

hence the name policy gradient:

$$\theta_{t+1} \leftarrow \theta_t + \alpha R_t \nabla \ln \pi(a_t|s_t, \theta_t). \tag{2.14}$$

The update increases the parameter vector in the direction proportional to the return, and inversely proportional to the action selection probability. The former favors the actions that yield the highest return, whereas the latter avoids the actions that are frequently selected to be at an advantage.

Actor-Critic Methods: Learning Both Value Function and Policy

Note that the updates required for policy-gradient methods as in Eq. 2.14 would require full return to be calculated for each time step by summing all future rewards until the end of the episode. Thus, the learning procedure would be of Monte-Carlo type (the exact policy-gradient algorithm is called *REINFORCE*, see [164]), which has the main drawbacks of requiring full episodes to be completed and exhibiting high variance, thus leading to slow learning, as discussed in Section 2.5.2. A more efficient variation would be to subtract a baseline function from the return R to reduce the variance of gradient estimation without changing its expected value, hence the direction of the descent. A good choice of a baseline function is the estimate of the state value, $v(s_t)$. In this case, the value function also needs to be estimated, for which one can use the methods approximating the value function as described in Section 2.5.2. The idea of combining the policy-based with value-based methods finally bring us to the actor-critic methods.

The proposition of using the value function estimate v in the policy gradient methods can be generalized to use v for estimating the actual return as well. As in TD learning, the estimated value can be used to construct one-step return $r_{t+1} + \gamma v(s_{t+1}, w)$, which is often superior to the actual return in terms of its variance and computational ease, even though introducing bias. Yet, the extent of the bias can be flexibly modulated with n-step returns. The estimated value can this way assess the action selection as in policy evaluation methods. When the state-value function is used to assess actions in this way, it is called a *critic* and the policy is called an *actor*, and the overall policy-gradient method is termed an *actor–critic method*.

Considering the one-step case, actor-critic methods replace the full return in Eq. 2.14 with the one-step return, besides using the learned state-value function as the baseline:

$$\theta_{t+1} \leftarrow \theta_t + \alpha[r_{t+1} + \gamma v(s_{t+1}, w) - v(s_t, w)]\nabla \ln \pi\{a_t|s_t, \theta_t\}. \tag{2.15}$$

Thus, the TD error $[r_{t+1} + \gamma v(s_{t+1}, w) - v(s_t, w)]$ is used to scale the policy gradient. This term is also called as *advantage*, as it indicates how much additional reward can selecting an action at a particular state bring, as compared to value of being in that particular state in general. This common variant of actor-critic methods is also called as *advantage actor-critic methods*. The natural method to learn state-value-function to be combined with the updates in Eq. 2.15 is one-step TD.

In general, the actor-critic agent learns two sets of parameters by the critic and the actor, as illustrated in Fig. 2.17. The critic updates the value function parameters w using any policy evaluation method, such as TD learning, and evaluates the actions taken by the actor, also using the same TD target. The actor takes the actions, and updates the policy parameters θ in the direction suggested by the critic, using any policy gradient method.

Actor-Critic for Continuing Problems

RL agents often deal with environments with continuing problems, for which the interaction between agent and environment goes on without terminating or start states, i.e., without any episodes. In such problems, episodic return formulation in Eq. 2.1 becomes infeasible as T and R would go to infinity. Instead, an *average reward* setting is considered for continuing problems [164]. In the average-reward setting, the quality of a policy π is defined as the *average reward* $r(\pi)$, while following that policy:

$$r(\pi) = \lim_{h\to\infty} \frac{1}{h} \sum_{t=1}^{h} \mathbb{E}[r_t|s_0, a_{0:t-1} \sim \pi] \tag{2.16}$$

Returns in the average-reward setting are defined in terms of differences between rewards and the average reward:

$$R_t = r_{t+1} - r(\pi) + r_{t+2} - r(\pi) + r_{t+3} - r(\pi) + \dots \tag{2.17}$$

also known as the *differential return*. Similarly, TD errors are expressed in differential form as:

$$\delta_t = r_{t+1} - \bar{r}_t + v(s_{t+1}, w_t) - v(s_t, w_t) \tag{2.18}$$

where $\bar{r}_t$ is an estimate of the average reward $r(\pi)$ at time t. With the use of average reward setting, the discounted reward formulation introduced in Eq. 2.2 also becomes unnecessary, since the average of the discounted returns is always $r(\pi)/(1-\gamma)$, which is essentially equal to the average reward (cf. Section 10.4, [164] for the proof).

Algorithm 1 gives the complete pseudo code for the one-step actor–critic for the continuing case, where the critic updates w by TD learning, and the actor updates θ by policy gradient using the TD error [164].

Algorithm 1 One-Step Actor-Critic for Continuing Problems [164]

```
Initialize policy parameters $\theta$ and state-value weights $w$ arbitrarily
Initialize $\bar{r}$, e.g., to 0
Initialize $s$
loop until convergence
    Take action $a \sim \pi(\cdot|s, \theta)$, observe $s'$, $r$
    Calculate TD error, $\delta \leftarrow r - \bar{r} + v(s', w) - v(s, w)$
    Update estimate of the average reward, $\bar{r} \leftarrow \bar{r} + \beta\delta$
    Update weights, $w \leftarrow w + \alpha\delta\nabla v(s, w)$
    Update parameters, $\theta \leftarrow \theta + \alpha\delta\nabla \ln \pi\{a|s, \theta_t\}$
    $s \leftarrow s'$
end loop
return $\theta, w$
```

Asynchronous Advantage Actor-Critic (A3C): Training Multiple Actors in Parallel

The actor-critic algorithm has the drawback that the agent only observes a certain region of state space at a time, which can improve its policy in that region, while performing suboptimally in other regions of the state space. Besides, since the agent updates its parameters and weights based on consecutive states and actions, the updates are correlated, which contain similar states and actions, thus, again limiting the generalization capability of the agent.

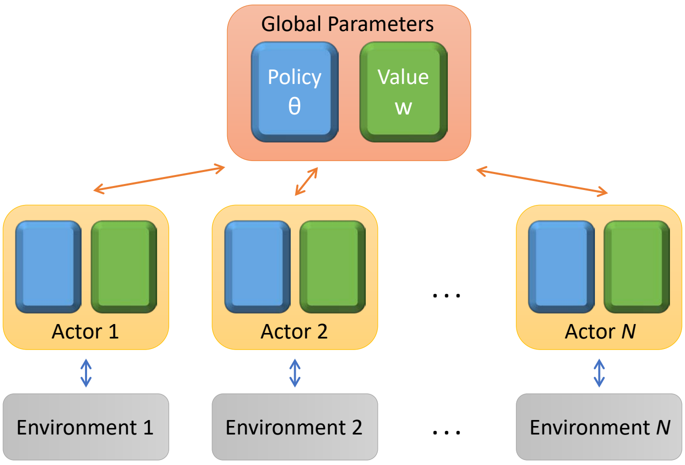


Fig. 2.18 A3C architecture (based on [174]).

As a solution, the authors in [173] have developed a new paradigm for deep learning. In this approach, multiple learners are executed on multiple instances of the environment in a parallel, asynchronous manner. The parallelism allows the learners' data to be uncorrelated, as they experience a variety of different states at any given time, exploring different regions of the state space. In addition to stabilizing learning, using multiple parallel learners reduces the training time roughly linear in the number of parallel learners. This simple yet effective idea is applicable to a large set of fundamental RL algorithms, enabling a robust usage of deep neural networks. The actor-critic variant, called *Asynchronous Advantage Actor-Critic (A3C)*, is shown to significantly outperform the other asynchronous versions of the standard RL algorithms on a variety of different tasks [173]. In A3C, multiple parallel actors collect experience from their own environment asynchronously, and update the parameters of a global policy and a value function that is shared by all parallel actors, as illustrated in Fig. 2.18. A3C also generalizes the one-step actor-critic algorithm with the idea of *n*-step bootstrapping, due to its benefits in terms of reducing the bias in learning as explained in Section 2.5.2. Algorithm 2 presents the pseudocode of the A3C algorithm from the actor-learner's perspective [173].

2.6 Reinforcement Learning in Resource Allocation for Vehicular Communications

RL has recently attracted the area of wireless communications. Despite being recent, it has found a wide range of applications, particularly targeting the recent and next generation of wireless networks including Internet of Things (IoT) [227], heterogeneous networks

Algorithm 2 A3C (per actor-learner) [173]

```
Initialize global shared policy parameters $\theta$ and state-value weights $w$
Initialize actor-learner-specific parameters $\theta'$ and weights $w'$, and counter $t = 1$
**loop** until convergence
    Reset gradients, $d\theta \leftarrow 0$ and $dw \leftarrow 0$
    Synchronize actor-learner-specific parameters, $\theta' \leftarrow \theta$ and $w' \leftarrow w$
    $t_{\text{start}} \leftarrow t$
    Initialize first state $s$
    **loop** until terminal state or for $t_{\text{max}}$ steps
        Take action $a \sim \pi(\cdot|s, \theta')$, observe $s'$, $r$
        $t \leftarrow t + 1$
    **end loop**
    $R = \begin{cases} 0 & \text{if terminal state} \\ v(s_t, w') & \text{else (bootstrap from last state)} \end{cases}$
    **for** $i \in \{t-1, \dots, t_{\text{start}}\}$ **do**
        Calculate $n$-step return, $R \leftarrow r_i + \gamma R$
        Accumulate gradients wrt. $w'$, $dw \leftarrow dw + \partial(R - v(s_i, w'))^2 / \partial w'$
        Accumulate gradients wrt. $\theta'$, $d\theta \leftarrow d\theta + (R - v(s_i, w'))\nabla \ln \pi\{a_i|s_i, \theta'\}$
    **end for**
    Perform asynchronous update of $\theta$ and $w$ using $d\theta$ and $dw$, respectively.
**end loop**
```

Table 2.2 Overview of works applying RL to resource allocation for V2X communications (derived from [24], [175], [176], [177], [178], [179], [180], and their references).

Problem	RL Method and References	Objective
Optimization of underlay V2V and V2N	DQN: [181], [182], [183], [184], [185], [186], [187] DDPG: [188]	Minimizing interference to V2N links while meeting the latency constraints of V2V messages
	ACRL: [189] DQN: [190], [191] DDPG: [192]	Jointly optimizing mode selection and resource allocation
	DQN: [193], [194], [195]	Reducing channel state information-related signaling overhead
V2V channel allocation	DQN: [196]	Reducing queueing delay or transmit power
	DQN: [197]	Minimizing age of information
V2V collision management	Q-learning: [198], [199], [200], [201] DQN: [202]	Optimizing contention window size selection
	DQN: [203]	Optimizing cognitive radio channel selection
V2N uplink/ downlink scheduling	Q-learning: [204], [205], [206] DQN: [207], [208]	Maximizing number of vehicle download requests on DL
	Q-learning: [209] DQN: [210]	Optimizing UL/DL frame ratio
	Q-learning: [211], [212] DDPG: [213]	Optimizing DL beamforming
User association, load balancing, vertical handoff	Q-learning: [214]	Optimizing spectrum sharing between cellular V2N and Wi-Fi
	Q-learning: [215]	Optimizing mobility management between RSUs and BSs
	Q-learning: [216]	Optimizing load balancing across macro, pico, femto cells
	A3C: [217]	Vehicle-cell association for maximizing sum rate
Vehicular cloud optimization	PI: [218] DQN: [219], [220], [221], [222] A2C: [223] A3C: [224] BE: [225] SAC: [226]	Joint optimization of networking, caching, and computing resources

(HetNets) [228], unmanned aerial vehicle (UAV) networks [229], and satellite communications [230]. When combined with deep learning, RL has been shown to be an effective tool to address various problems and challenges of communication across different network layers, including network access, resource management, routing, traffic balancing, and security. Application of deep RL to wireless communications and networking are surveyed by several works such as [175], [176], and their references.

The application of RL in vehicular networks is also in its nascent phase. An overview of applying ML techniques to the challenges of vehicular networks is presented in [231] and [232], where the authors put a special emphasis on applying RL in particular for network resource management. The authors point out that the highly dynamic nature of vehicular networks challenges the conventional methods for resource management. Traditional methods typically formulate an optimization problem and solve it optimally or suboptimally depending on the complexity-performance trade-off. As the network topology and channel quality in vehicular networks vary continuously, such a conventional approach potentially needs to be rerun every time a minor change occurs, hence yielding huge overhead. Instead, RL could be an alternative effective solution, which interacts with, and adapts its actions to the unknown environment. Besides, RL natively supports the sequential decision-making encountered in the resource allocation problems [233].

In the rest of this section, we review the literature on works applying RL for the resource allocation problem in vehicular communications. As summarized in Table 2.2, we have categorized the works with respect to different problems they address and the RL method they utilize. The related surveys can be found in [24], [177], [178], [179], and [180]. We finalize the section by providing a summary and our general remarks on applying RL to our problem.

Optimization of Underlay V2V and V2N Communications

The majority of works applying RL to the resource allocation problem in V2X networks aims at optimizing an underlay operation of V2V and V2N links, considering decentralized schemes. In the considered problem, vehicles make resource selection based on local observations, which are independent of V2N communications that take place between the vehicles and a base station. With the assumption that orthogonal resources are allocated for V2N links beforehand, the objective is to minimize interference to V2N links while meeting the latency constraints of V2V messages. A common approach to the problem in the related works is to model it as a multi-agent RL task, where each vehicle is regarded as an agent, making its own decisions on resource allocation. In [181] and [182], authors consider scenarios with broadcast and unicast V2V messages, respectively, and combining both in [183]. In the proposed method, vehicles select a sub-channel and a power level for their V2V transmissions, based on their local observation (state) of the environment. The state information characterizing the environment is defined as the instantaneous channel information of the V2V link and V2N link, the interference level and selected sub-channels of neighbors in the previous time slot, the remaining amount of traffic of the transmitting vehicle, and the remaining time to meet the latency constraints. In the broadcast scenario, the number of times that the V2V message has been received by the vehicle and the distance to the vehicles that have broadcast are additionally considered in defining the state. The performance evaluations show better V2N link capacity and a larger ratio of satisfied V2V links when compared to random resource allocation and the clustering-based heuristic scheme in [147] that utilizes the position information of vehicles. The proposed scheme in [184] further considers a per-packet-priority metric for the same objective.

In the above-mentioned works, DQN-based RL methods are utilized, and it is assumed that a single deep neural network is shared across all vehicles for their actions. A performance gain with respect to [183] is shown by the same authors in [185] and [186], as well as in [187] when each vehicle has its own DQN for the resource allocation purpose, trained independently. Also in [188], each vehicle, as an agent, explores the environment in a distributed fashion and makes strategy decisions based on its own observations. Authors further consider non-orthogonal multiple access (NOMA) technology to reuse the V2N spectrum for V2V communications and deem the transmission power to be allocated as a continuous variable, and employ a deep deterministic policy gradient (DDPG) algorithm to handle such action space.

Several works consider further enhancements to the distributed learning setting targeting better joint optimization of V2V and V2N links. Authors in [189] address the problem of joint resource allocation and mode selection between cellular V2N and V2V links, using an actor-critic RL (ACRL) algorithm instead of Q-learning based approach common to the above-mentioned works. The proposed method further considers vehicular users with poor learning performance to transfer learning from expert users to enhance learning efficiency and convergence speed. In [190], authors utilize a multi-agent double deep Q-learning (DDQN) algorithm to tackle the same problem. The study in [191] additionally considers resource sharing between V2V pairs in different modes for the same problem, while tackling it with a federated learning setting considering limited local training data at vehicles. Another study [192] considers a different use case involving multiple platoons where platoon leads (PLs) attempt to access the frequency spectrum aiming at disseminating the V2V messages between their followers while keeping an updated connection with the RSU. The objective is to minimize the age of information (AoI) at the RSU, which is defined as the amount of time elapsed since the most recent information update available from a corresponding platoon. The spectrum access is modeled as a multi-agent RL problem, where the state observed by the PL contains the CSI of the V2V and V2N channels, previous interference from other platoons to PL, the AoI of PL, the remaining intra-platoon payload, and the remaining time budget. The action of each PL consists of the subchannel selection and mode selection of PL, and the power control, determined by a twin delayed deep deterministic policy gradient (TD3) algorithm. The gains are shown with respect to a simpler DDPG algorithm in terms of intra-platoon data rate, average AoI, and V2V message transmission probability.

As a drawback, all of the proposed schemes above require instantaneous CSI of the links, as well as a high amount of information exchange between the vehicles at each step of resource allocation. To cope with this problem, authors in [193] propose a distributed learning-based CSI compression scheme and a centralized decision-making architecture to maximize the sum rate of all V2V links. The same objective of reducing the signaling overhead while jointly optimizing the V2V and V2N links is also tackled with a centralized approach in [194]. The proposed scheme is based on clustering of base stations and vehicles, and a DQN-based RL technique for resource allocation. Authors in [195] consider fast-varying channels due to vehicle mobility, and exploit recent advances in recurrent neural network (RNN) to tackle this issue without relying on CSI information. Instead, the observation space consists of sensed channel interference measurements, as well as remaining payload size and time budget. Based on this, the vehicular agent decides its own selection of sub-channel and transmission power level to maximize the sum throughput of V2N links while meeting the latency and reliability requirements of V2V links. The developed algorithm shows comparable performance with a CSI-involved version, as well as the NR V2V resource allocation mode 2.

V2V Channel Allocation

Several works consider V2V communication without underlay operation in the V2N bands. Authors in [196] propose a channel allocation scheme with the aim of striking a tradeoff between the queuing delay and the transmit power consumption of vehicles. In the proposed scheme, RSU clusters the vehicle pairs into disjoint groups based on their geographical locations, and makes channel allocation based on the vehicles' local state indicating their channel quality, position information, and queue status. The authors decompose the single-agent problem into a decentralized SARSA algorithm with vehicle pairs as agents. In order to tackle the partial observability and the curse of high dimensionality in the local state-space faced by each vehicle pair, the authors propose LSTM (long short-term memory) and DQN-based techniques. In [197], the same authors target the problem of minimizing the AoI metric in V2V communications. The RSU is proposed to make decisions on frequency band allocation and packet scheduling within an MDP framework using the geographical locations, CSI, packet arrivals, and AoI of transmissions between each vehicular pair as the state information input to a DQN with LSTM layer.

V2V Collision Management

Another line of works proposes contention-based MAC protocols using RL to tackle the challenges of distributed resource allocation in VANETs. In [198], authors come up with a Q-learning based algorithm for unicast V2V traffic, which dynamically adjusts the contention window (CW) size at each vehicle to avoid packet collisions using position information, without relying on RTS/CTS to solve the hidden node problem. The proposed scheme is shown to provide a significantly higher packet delivery ratio, lower end-to-end delay, and higher fairness than the original IEEE 802.11p scheme (with/without RTS/CTS, for different CW sizes). Authors in [199] extend the problem to broadcast V2V communications, utilizing again a Q-learning-based algorithm that takes binary feedback (ACK/NACK) as the reward. Evaluations show an increase in packet delivery ratio of V2V broadcast transmissions by 37.5% as compared to IEEE 802.11p under high-density traffic. The same authors improve their method by utilizing a different reward function to trade throughput and fairness of users for lowering transmission latency or the opposite, in [200]. Authors in [202] propose using state representation that includes CW values received from neighboring vehicles, corresponding success rates, and frequency values to adjust the CW of the ego vehicle. The setting is also extended to multi-channel operation of DSRC standard. Proposed algorithm shows 21% improvement of packet delivery ratio when compared to a simpler algorithm utilizing only ego-CW information for the state representation, however, requiring additional information exchange between the vehicles to establish richer state information. Contrary to the above-mentioned works that increase or decrease the CW exponentially, authors in [201] propose linear adjustment of CW using Q-learning that takes the current CW size, last transmission status, the queue size, and the estimated number of neighboring vehicles as state information. The gain is shown in terms reduced rate of failed transmissions as compared to conventional exponential back-off in IEEE 802.11p. In [203], authors consider cognitive radio (CR) operation, where RSUs periodically sense the channels with regards to their occupancy probability to get information about cognitive channel availability. RSU selects the optimal CR channel after the processing of the sensed data, and provides this to the vehicles upon request. Marginal gains have been shown in terms of average delay and packet delivery ratio as compared to existing CR protocols.

V2N Uplink/Downlink Scheduling

Several other works focus on centralized managing of the resources by applying RL to schedule downlink V2N links. Authors in [204] target maximizing the number of vehicle download requests with an MDP framework using discretized states to establish an optimal RSU scheduling policy. The same authors incorporate the objective of energy consumption of the RSU in [207], and extend the problem to multiple RSUs in [208]. With the aim of serving the vehicles on downlink with minimized waiting time for safety messages, the authors consider a continuous state space and handle this by employing a DQN. In [205], the authors propose a Q-learning algorithm to find a policy that minimizes the number of services requested by vehicles that fail to meet deadlines. Authors in [206] further utilize V2V links to cooperatively relay the downlink data within a cluster of vehicles, with the aid of RL-based scheduling. The study in [209] proposes a Q-learning scheme for BS to choose the UL/DL ratio of time slots in the same frequency band. By considering the predicted future network situation, the scheme is shown to outperform a conventional policy in throughput. The same authors extend the problem to heterogeneous networks involving macro and small cells and different users in terms of mobility (vehicles, VRUs, and UAVs) in [210]. In this study, network performance is improved by considering states of historical time sequence as well as incorporating deep belief neural network to their Q-learning algorithm. The authors in [211] consider the concept of virtual cell formation to serve a user from *multiple* RSUs simultaneously. They evaluate the impact of various RL-based approaches to the problem of user-RSU association and determine downlink beamforming weights based on CSI. The same authors further integrate the minimization of energy consumption to the problem in [212]. Authors in [213] aim at reducing the average packet delivery latency between BS and vehicles in an intersection scenario by adjusting the downlink beam directions considering a MIMO system. The proposed algorithm based on DDPG can achieve optimal average delay performance as compared to traditional methods, especially under non-stationary channels, where vehicles travel at high and time-varying speeds.

User Association, Load Balancing, Vertical Handoff

Authors in [214] target the problem of sharing of unlicensed spectrum by cellular V2N communications and Wi-Fi users. Based on the proposed Q-learning scheme taking the system state as input, the base station adjusts the duty cycle of the unlicensed channel, which is shown to enhance overall communication capacity and ensure fair co-existence between the two technologies. On the other hand, a vertical handover control between cellular V2N and 802.11p V2I links is treated by a fuzzy Q-learning algorithm in [215] to ensure seamless mobility management of vehicular users between RSUs and cellular base stations deployed along the road. In the proposed scheme, RSUs, as learning agents, take various information reported by vehicles into account, which includes average received signal strength level, vehicle velocity, and data type, and also consider the traffic load to make handoff decisions. Along this line, [216] proposes an RL-based algorithm to associate vehicular users to heterogeneous base stations of macro, pico, femto types for the purpose of balancing the network load among them, based on the information of traffic load and received strength of the pilot signal by the vehicles. In [217], authors consider vehicle-cell association in mmWave communication networks to maximize the network-wide sum rate while guaranteeing a minimum threshold service rate for all vehicles. In the proposed distributed RL method, every RSU operates as a local A3C agent that associates users based on the channel observations, experienced rate of vehicles, threshold violation, and the reward function capturing the optimization problem.

Vehicular Cloud Optimization

Some works use RL to manage resources in V2X networks not only for communication but also for computation, storage, etc. in a joint manner, targeting various objectives such as the revenue of network operators per vehicle. The vehicular cloud, which consists of various OBUs, RSUs, and remote cloud servers, brings the concept of different types of resources together. The aim is to provide a pool of processing, sensing, storage, and communication resources that can be dynamically provisioned for vehicular services [234]. The analysis in [218] highlights the drawbacks of traditional approaches to manage the configuration of such resources. Instead, the benefits of an MDP-based approach using policy iteration (PI) are shown considering virtual machines that serve ITS services, which have abstracted resources for processing, sensing, storage, and communication. In [219], networking, caching, and computing for vehicular networks are considered jointly, where vehicles offload their computation tasks to the virtual resources. The proposed Q-learning algorithm optimizes the selection of base stations with the best channel quality. The sum data rates of V2N links are constrained by their backhaul capacity. The same authors improve their method by utilizing CNNs and combining techniques of double and dueling deep Q-networks in [220] and [221], respectively. They generalize their framework to smart city applications in [222]. These works, however, assume fixed statistics of the variation in resources such as wireless communication channels and computing capabilities. The authors in [223] handle the same problem in environments with variations involving diverse scenarios. They tackled this by combining hierarchical RL with meta-learning and adapting the A2C algorithm (advantage actor-critic, the synchronous variant of the A3C algorithm). Authors in [224] employ the

A3C algorithm to determine which base station to assign to the vehicle, and whether the computation task should be offloaded to the edge server. The decisions are made based on the computation capability and communication link capacity of each base station. The proposed method in [225] targets delay-optimal resource allocation for V2N links between software-defined virtual BSs and vehicles, using Bellman equations (BE). The scheme operates at two stages taking large timescale factors (traffic density), and short timescale factors (perfect CSI and queue state information knowledge per vehicle) measured by base stations, respectively. Different from the above-mentioned works, authors in [226] take into account the diversity and difference of services in vehicular networks with respect to their QoS requirements, and further consider two modes (V2N and V2V) for content delivery. The proposed approach utilizes proximal policy optimization (PPO) and soft actor-critic (SAC) algorithms to reduce the action space dimension.

Summary and Remarks

In summary, RL has found successful applications in vehicular networks for resource allocation tasks, majorly in terms of optimizing the coexistence of V2V links with V2I links or different access technologies, as well as orchestrating network resources jointly for computing, storage, and communication. The majority of the works have applied value-based RL methods, including Q-learning for problems with relatively small state spaces, and DQN or its variants such as DDQN for large state spaces. The remaining of the works consider actor-critic-based RL methods such as A3C, SAC, or DDPG to handle the continuous action space in their problems.

Up to our best knowledge, there is no specific RL-based solution targeting the resource allocation problem for V2V communications under intermittent network coverage. Besides, the state information required by the proposed RL-based resource allocation methods in the literature often demands instantaneous and detailed knowledge of the communication channels. Collecting such information by vehicles becomes infeasible due to highly dynamic V2V links, especially when broadcast or multicast communications considered. Further, in the case of centralized coordination of resources that we consider in approach, usage of such collected information may not be possible under intermittent connectivity between the vehicles and the coordinating entity.

In our solutions, we have decided to employ an actor-critic RL method, namely the state-of-the-art A3C algorithm described in Section 2.5.3, due to several reasons. First, despite the popularity of value-based methods, such as DQN, actor-critic methods have theoretical advantages over them, as well as over policy-based methods. Notably, actor-critic methods exhibit lower variance and faster learning, as we described in Section 2.5.3. Next, as we elaborate in the following chapters, our problem setting bears an environment that is difficult to model as an MDP, and the state space we consider is continuous and huge. Therefore, applying traditional or tabular RL methods, e.g., dynamic programming or Q-learning, becomes infeasible, calling for function approximation techniques such as deep learning. When combined with deep learning, A3C has been shown to deliver superior performance over other prominent RL algorithms including DQN [173]. In Chapter 5, we extend the basic A3C algorithm presented in Section 2.5.3 to make it applicable to continuous tasks (as opposed to episodic ones), which we encounter in our problem setting.

Chapter 3

System Model and Simulation Environment

This chapter first presents a precise description of our system model, namely the vehicular network environment, and our assumptions to study the proposed methods in this thesis, in Section 3.1. Section 3.2 defines the metrics we utilize to evaluate the performance of the proposed algorithms. In our evaluations, we have mainly adopted simulation as a research tool. Simulations offer a more viable, flexible, and cost-effective approach, in a more controlled and safer environment as compared to experimenting with real vehicles and wireless networks, which is also taken by the majority of research work on vehicular networks [235]. Section 3.3 describes the simulation environment we have developed to conduct performance evaluations, where we have combined widely-utilized simulation tools and extended with necessary functionalities to support the considered system model.

3.1 Vehicular Network Environment Model

3.1.1 Data Traffic

We consider a vehicular network where vehicles transmit and receive single-hop V2V messages via direct communication. Message traffic is assumed to be of two types: periodic and event-triggered (i.e., aperiodic). A typical example of periodic traffic is the regular broadcast of vehicle information such as position and speed, as in cooperative awareness messages (CAMs) [18], whereas aperiodic traffic is triggered on events to warn vehicles such as of an accident, as in the case of decentralized environmental notification messages (DENMs) [19], as introduced in Section 2.1.

In line with the 3GPP evaluation assumptions [236] and studies in the literature [237], [238], we model[1] the first type of traffic with messages of fixed size S_m and periodicity T_m, and the second type of traffic by a generation of a single message of size S_m upon random events, whose arrivals, assuming statistical independence of such events from each other, follow a Poisson distribution. The Poisson model is a widely-used and acknowledged method for characterizing discrete uncommon events like car accidents [239].

In our evaluations, we primarily consider broadcast type of messages, which are one-to-many transmissions sent from a single vehicle to all other vehicles around, as it constitutes the major part of V2V use cases, enabling situational awareness. Nevertheless, we further consider unicast messages, of periodic type, which are one-to-one transmissions sent from a single vehicle to another specific one, in our evaluations in Chapter 4.

3.1.2 Cellular Network Connectivity

We consider an area comprising a two-way highway segment as shown in Fig. 3.1, with J lanes per direction. Along the road, a cellular network is deployed where vehicles can connect to base stations (BSs). While under the BS coverage, vehicles transmit their V2V messages using the radio resources scheduled by a centralized entity. As discussed in Section 2.3.3, such scheduling entity is deployed at the edge of the cellular network, with access to the BSs. Via BSs, vehicles communicate with the scheduler by signaling the control information required for the resource allocation.

With regards to connectivity between the vehicles and the BSs (hence the scheduler[2]), we distinguish between two main settings:

(i) V2V communications in an area where no communication between the vehicles and the BSs is possible at all. As discussed in Section 2.4.1, such area can be a road tunnel. We consider a road section outside the coverage of BSs located at its two ends, i.e., delimiting it, as illustrated in Fig. 3.1. Thus, we call this area as *delimited out-of-coverage area (DOCA)*. DOCA is assumed to be of length L_{DOCA}. BSs deployed at each end of DOCA are able to serve the vehicles just before (after) they enter (exit) DOCA. We assume that the existence of DOCA, as well as its location, length, etc. are

[1] According to the ETSI specifications [18], [19], while periodic by default, the periodicity of CAM transmissions can be adjusted depending on vehicular mobility and traffic load, and DENM transmissions might contain bursts of several periodic messages. We evaluate the impact of the different modes of traffic separately, by following previous research work (e.g., [80], [117]) and applying the traffic models proposed in 3GPP [236]. This approach makes the analysis easier and enables us to generalize our results to any type of V2V traffic beyond the CAM and DENM applications.

[2] We assume no loss of communication between the BSs and the scheduling entity.

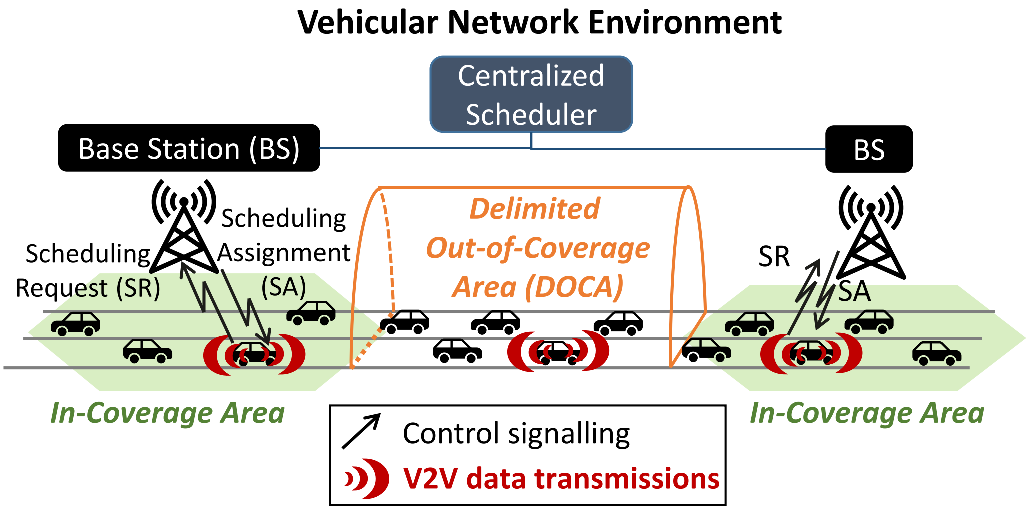


Fig. 3.1 Vehicular network environment containing a delimited out-of-coverage area (DOCA). In coverage, vehicles send scheduling request (SR) to the centralized scheduling entity via base station (BS) to request resources for their V2V transmissions. In turn, vehicles are informed about the allocated resources via scheduling assignment (SA).

known to the network operator, such as via map data. DOCA is the main setting we consider in our evaluations throughout Chapters 4-7.

(ii) V2V communications under the coverage of the BSs, however communication between the BS and vehicles experiencing unexpected interruptions due to path loss and fading characteristics of the environment[3]. Compared to DOCA, this type of coverage losses are much shorter, local, and also unpredictable. We consider this setting in our evaluations in Chapter 8, also in combination with the first setting.

Communication between the vehicles and the BSs takes place over the uplink (UL) and downlink (DL) that use resources different than the ones used for V2V communication. Assuming a frequency division duplex (FDD) operation, UL and DL are both assumed to occupy a 10 MHz of bandwidth, centered at 2.1 GHz, utilizing COST-Hata channel model [240]. Vehicles and BSs are assumed to transmit with fixed-power on UL and DL, respectively, with omni-directional antennas. We utilize the error model for the links introduced in the ns-3 LTE LENA project, which provides an accurate and lightweight abstraction of the errors [241]. This error model accounts for the intermittent connectivity mentioned above, namely the loss of the control messages between the BSs and the vehicles that are used to schedule V2V messages.

[3] Note that we exclude the possibility of an external interference (e.g., due to an attacker or resulting from adjacent frequency bands) similarly blocking the communications, which we mentioned in Section 2.4.1.

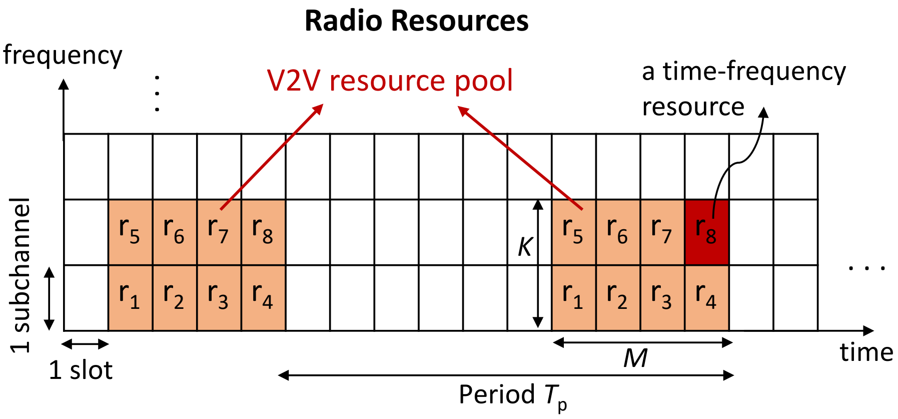


Fig. 3.2 Radio resources and the resource pool configured for V2V communications.

3.1.3 V2V Communication Resources

For V2V communications, we consider a dedicated frequency band of 10 MHz bandwidth at a carrier frequency of 5.9 GHz, namely the ITS band introduced in Section 2.1, hence separate from the bands used for UL/DL communications. In line with the 3GPP specifications [12], V2V communication takes place in a dedicated resource pool configured by the cellular network operator within this band. We denote the pool configuration by $C^{K \times M}$, where K is the number of subchannels and M is the number of slots that the pool contains, over the frequency and time domains, respectively, repeating over time with fixed periodicity T_{p}. Following the 3GPP assumptions [236], a transmission of a single V2V message occupies a single time-frequency resource $r_i \in C^{K \times M}$, which is referred to as a transmission block (TB). A TB consists of a single time slot and a single frequency subchannel containing sufficient number of resource blocks (RBs) to carry the message, given the message size S_m and the employed modulation and coding scheme (MCS). The considered pool, thus, contains a total of $R = K \times M$ TBs, or simply referred to as resources.

As shown in Fig. 3.1, when served by BSs, vehicles can send scheduling request (SR) to the scheduling entity to request resources for their V2V transmissions, which contains information about their V2V traffic, such as S_{m} and T_{m}. In turn, vehicles are informed about the scheduling decision by the scheduling assignments (SAs) sent via BSs. In line with the 3GPP standard, the SA indicates which time-frequency resource to use for transmitting their V2V message in the next available instance of the resource pool. We assume that vehicles are synchronized with each other as well as with the BSs, without consideration of any synchronization errors. To help making the scheduling decisions, the scheduling entity can request vehicles or BSs to report information regarding the mobility of vehicles, such as their location, speed, etc.

3.1.4 V2V Radio Propagation and Interference Model

Vehicles are assumed to have fixed transmission power P_{Tx} to transmit their V2V messages. Transmitted V2V messages are subject to path loss and fading effects of the wireless channel, besides the interference from any other transmission using the same resource. Successful reception of a packet depends on the signal-to-interference-plus-noise ratio (SINR) at the receiver. The SINR of a single transmission at receiver j from transmitter i is:

$$\mathrm{SINR}_{ij} = \frac{P_{\mathrm{Tx}}|h_{ij}|^2}{\sigma^2 + \sum_{l=1,l\neq i}^{L} P_{\mathrm{Tx}}|h_{lj}|^2}, \tag{3.1}$$

where P_{Tx} is the transmit power of the transmitter, and $|h_{ij}|^2$ denotes the channel coefficient between the transmitter i and the receiver j, which accounts for the path loss and fading effects of the wireless channel on the transmitted signal. σ^2 is the noise power, and the summation term in the denominator denotes the interference due to the other vehicles $l = 1,...,L$ using the same TB as i. The SINR depends on the interference level of the other transmissions using the same TB, under the path loss and fading effects of the propagation channel. V2V channel model is assumed to consist of realistic path loss with shadowing fading according to *WINNER+ B1* model [242].

To evaluate whether a received packet can be successfully decoded or not, we use the model developed by the US National Institute of Standards and Technology (NIST) [243], which maps the SINR at the receiver to transport block error rate (TBLER) for the corresponding received packet, given the utilized MCS. Vehicles are further assumed to be equipped with half-duplex (HD) radios for V2V communications, that is, they can either transmit or receive, but can not do both at a given time slot [244]. We therefore distinguish the following types of errors in reception of V2V messages:

- HD error: Vehicles transmitting at the same time are not able to receive each other's message (e.g., resources r_1 and r_5 sharing the same slot in Fig. 3.2). We refer to unsuccessful reception of messages due to HD limitation as *HD error* or *conflict*, and the relation among the TBs in the same slot causing this phenomenon, as *HD constraint*.

- Collision error: Messages transmitted by different vehicles using the same resource, i.e., the same slot *and* the same subchannel, may, depending on the propagation conditions, interfere and lead to decoding errors at the receiver, which we refer to as *collision errors*.

- Propagation error: Further, unsuccessful receptions could also result from the channel effects, e.g., due to path loss and shadowing that considerably reduce the received SINR (or SNR), which we refer to as *propagation errors*.

3.1.5 Vehicular Mobility

For the vehicular mobility, we utilize the realistic Krauss car-following model [245] combined with the lane-changing model [246] implemented by the mobility simulator SUMO [247], which is a popular tool widely used in the vehicular community. The models take several factors into account, such as the desired velocity, following distance, number of lanes, etc., which lead to realistic simulation of mobility conditions, including a stochastic driver behavior. Vehicle arrivals into the considered section of the highway follow a Poisson distribution, which is shown to satisfactorily approximate the free-flowing traffic and has found well-established applications in modeling vehicular mobility (cf. [248], [249], and their references). In the utilized models, vehicles adhere to the maximum speed set for the highway, where a speed value is selected once for each vehicle entering the area from a normal distribution $\mathcal{N}(\mu, \sigma^2)$. By varying the mean μ and the standard deviation σ of this distribution, we adjust the vehicle density and dynamicity in the considered area as desired.

3.1.6 Training Environment

For training the RL models proposed in Chapters 5-8 before their deployment, we utilize a simpler model of the environment, in terms of mobility and radio propagation. As we discuss in the respective chapters, such a simplified environment model enables a computationally efficient, hence faster way of training the RL agent, which we also used for prototyping and developing our RL algorithms. We also utilize this model in Chapter 4, where our evaluations focus on feasibility analysis, neglecting the intricate impacts of the wireless channel and vehicular mobility. In this model, vehicles travel at constant speeds, randomly selected from a normal distribution. To account for a realistic distribution of inter-vehicle gaps, vehicles are initially distributed uniformly random on the considered road section, yet they are assumed to return back from the opposite direction with an exponentially distributed time offset upon leaving the area. Radio propagation in the environment is abstracted with the protocol model [250], assuming fixed circular V2V transmission ranges of R_{Tx} m. With this model, unsuccessful receptions are assumed to result from any transmission whose range intersects with another one using the same resource, at the receiver, besides the errors due to HD radio operation.

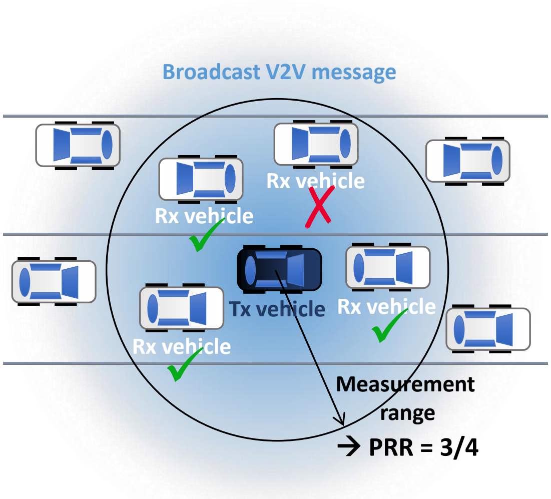


Fig. 3.3 Illustration of packet reception ratio (PRR) calculation.

3.2 Key Evaluation Metrics

As introduced in Chapter 1, ITS use cases have stringent QoS requirements, most prominently in terms of reliability of V2V communications and awareness of vehicular users. Our objective is to provide methods enabling more efficient utilization of radio resources to support V2V communications. We translate such requirements into the following measurable key performance indicators to evaluate our proposed methods in the following chapters.

- Packet reception ratio: We quantify the reliability of V2V transmissions with packet reception ratio (PRR), which is a metric specified by the 3GPP standard [236]. For a single message transmitted from vehicle i, the PRR is calculated by X_i/Y_i, where Y_i is the number of vehicles located within the range (a,b) from the transmitter, and X_i is the number of vehicles with successful reception among Y_i. A simple calculation is illustrated in Fig. 3.3. In the case of unicast messages, this definition adapts $Y = 1$, namely a single targeted receiver [251]. PRR for all vehicles in the environment is then calculated for a series of messages consecutively transmitted by them in a given time interval as $(\sum_{i=1}^{V}\sum_{n=1}^{N_i} X_{i,n})/(\sum_{i=1}^{V}\sum_{n=1}^{N_i} Y_{i,n})$ with N_i denoting the number of generated messages by vehicle i, and V is the number of vehicles [236]. We measure PRR for all vehicles in the environment every 10 s, for a total duration of 1000 s, and show, for a certain transmitter-receiver range in meters, the mean, the median with its 95% confidence interval, as well as the 1st, 25th, 75th, and 99th percentiles of these measurements. We also compute PRR as a function of transmitter-receiver distance, using 20-m bins. In this case, we show the mean and the standard deviation as a function of distance for easier readability of the data.

PRR indicates the quality of the link between the vehicles, yielding the effective or reliable communication range, as well as the maximum communication distance [252]. The effective communication range is the maximum distance between the vehicles at which PRR is equal to or larger than a given threshold (e.g., 80%). Whereas, maximum communication range is the maximum distance where PRR is greater than zero. As introduced in Section 2.2, reliability requirements of V2V applications greatly vary with PRR values ranging between 80 – 99.999% at ranges of 20 – 1000 m depending on the use case involving different scenarios (e.g., highway or urban) and vehicle speeds [47], [253]. As a reference in our evaluations, we consider a target of 80% PRR at 100 m as a baseline requirement, which is crucial in terms of avoiding imminent crashes between the vehicles, taking the braking reaction time into account [254].

- Resource utilization: As introduced in Section 2.2, the vehicular communication system needs to satisfy various reliability requirements specified in terms of PRR and communication range, under *varying traffic loads*, such as impacted by the density and velocity of the vehicles on the road and their V2V message generation rate, all depending on the scenario and the use case(s) to be supported. The reliability performance of a scheduler would highly get impacted by the V2V communication traffic demand and the number of resources available to serve this traffic. Therefore, to account for practical analysis, we evaluate the PRR performance of the algorithms under different V2V data traffic loads. In particular, in Chapters 6 and 8, we go beyond the baseline PRR requirement we consider, and consider PRR targets of 80%, 90%, and 95% for different numbers of available resources in the allocated resource pool (i.e., R). From this data, we also derive the required amount of resources by the algorithms to ensure PRR targets at different communication ranges (e.g., 100, 200, 400 m), which are representative of different use cases introduced in Section 2.2.

  Another resource utilization metric we consider relates to the usage of UL/DL resources (i.e., different than V2V resources), which are used for control signaling between the vehicle and the BS (used to access the scheduling entity) for resource allocation purposes. In the in-coverage setting we consider in Chapter 8, V2V messages are dynamically scheduled as opposed to one-time scheduling assignment per vehicle in the case of a DOCA. Therefore, for this setting, it is desirable to achieve the required V2V reliability with minimum control signaling overhead, hence consuming fewer resources. We therefore further evaluate the impact of the scheduling update rate, namely the frequency of the SAs sent by the scheduler to the vehicles via BSs, on the achievable reliable V2V communication range under different traffic loads in Chapter 8.

In Chapter 4, where we dimension the resource allocation problem for V2V communications, we measure the required amount of resources based on the congestion status of the resources. We first define the probability of overloading of resources as:

$$P[\text{Overload}] = \begin{cases} 1, & \text{if } R < A. \\ 0, & \text{otherwise.} \end{cases} \tag{3.2}$$

where A is the number of resources occupied by the V2V traffic. Assuming aperiodic V2V traffic, which follows a Poisson distribution as introduced in Section 3.1, we have

$$P[k \geq R] = 1 - \sum_{k=0}^{R-1} e^{-\lambda} \frac{\lambda^k}{k!}, \tag{3.3}$$

where λ is the arrival rate of events. The non-overloading probability $1 - P[\text{Overload}]$ would reflect the reliability of V2V transmissions, which is given by

$$\text{Rel} = e^{-\lambda} \sum_{k=0}^{R-1} \frac{\lambda^k}{k!}. \tag{3.4}$$

By solving Eq. 3.4 for R, we determine the required amount of resources needed to achieve a given target reliability $\text{Rel}_{\text{target}}$. In Chapter 4, we evaluate required R to support different reliability targets under different communication loads (varied by the vehicle density) in the environment.

- Latency: An important scheduling metric is the latency of V2V messages. Latency for a single message is measured by the time difference between its generation and succesful reception at the application layer of the transmitter and the receiver, respectively. V2V application requirements set a maximum value on the latency of the messages. For the majority of applications introduced in Section 2.1, the typical requirement is a maximum latency of 100 ms [47]. In our evaluations, we report the mean value of the latency measured for all scheduled packets meeting this deadline.

- Admission rate: In Chapter 4, we consider that messages that can not be scheduled within the allowed maximum latency value are not admitted for transmission, i.e., dropped. The transmissions may not be scheduled such as due to resource conflicts preventing a successful reception. We, therefore, calculate the ratio of the number of scheduled transmissions to the total number of requested transmissions from the scheduler, which we refer to as "admission rate". As mentioned in Section 2.3.1, admission control (or congestion control) targets keeping channel load at or below a target level where the messages can be reliably delivered. Therefore, for the overall goal of maximizing vehicles' awareness of each other, an acceptable admission rate would be the maximum one that results in attaining other communication requirements such as PRR and latency.

- Packet inter-reception time: While PRR indicates the rate of lost V2V messages, it does not tell about the "burstiness" of the losses, i.e., whether or how much these losses are consecutive. Bursty losses may create "blackouts" of awareness at the receivers, which take relatively long times and considerably degrade the performance of the applications. Packet inter-reception (PIR) time is a metric used to evaluate the "situational awareness" of the vehicles in case of periodic V2V message traffic [255]. PIR is defined as the time elapsed between two successive successful receptions at a certain vehicle, transmitted from another one within a given range [236]. V2V applications typically require a maximum value of PIR in the order of a few seconds [256]. Whereas its mean value is ideally desired to be close as much as possible to the message generation periodicity. In our evaluations, we provide the mean and the percentiles of PIR measured for all messages generated in the vehicular environment.

- Fairness: Given that the PRR is calculated for all V2V transmissions belonging to all vehicles in the entire environment, this does not reveal whether all vehicles experience the same (or similar) reliability. Therefore, we additionally use the following metric to evaluate fairness among the vehicular users: the PRR_j will be computed separately for each vehicle j, as described above, and afterwards the standard deviation of the mean values of per-user PRRs will be estimated. We therefore expect the calculated standard deviation to be close to zero as much as possible in ideally fair conditions.

- Mutual awareness: We use the mutual awareness metric [17], [257] to study the impact of PRR on the performance of applications running over V2V links. Authors in [257] propose *awareness probability* P_A as an intermediate metric that relates communication quality of service and application performance. P_A is defined as "probability of successfully receiving at least n packets from a transmitter within the application tolerance time window T", i.e., $P_\text{A} = \sum_n^k \binom{k}{n} p^n (1-p)^{k-n}$, where p is the PRR at the transmitter-receiver range of interest, and k is the number of packets sent during T [257]. Thus, P_A reflects the communication performance in the form of PRR, i.e., reliability, and is used to evaluate its impact on the performance of V2V applications. Each V2V application can set requirements on awareness probability P_A, as well as on n and T. Requirements of several applications are exemplified in [257], which we also provide in our evaluations.

3.3 Simulation Environment

3.3.1 Available Simulation Tools

During the preparation of this thesis work, there were no simulation tools available that can do the following at the same time: i) simulate V2V communications based on the cellular standard; ii) simulate vehicular mobility; and iii) have interface to implement ML algorithms that can run in the network. Majority of the available tools for simulating vehicular communications were developed as modules for well-established network simulators such as *OMNeT++* [258] and *ns-3* [259], and by integrating the existing mobility simulators such as SUMO that is widely used by the vehicular community [247]. Examples include *Veins* [260] and its extension *Artery* [261], which couple the model of WAVE and ETSI ITS-G5 protocols implemented in OMNet++, respectively, together with SUMO. Another example was *iTETRIS* [262], which implemented the WAVE protocol in ns-3, also in combination with SUMO. *VSimRTI* [263] was also offering multiple network simulators combined with different traffic simulators. None of these simulators, however, were providing the support for the cellular standard for V2V communications. With regards to cellular networks, available system-level simulators were able to simulate only uplink/downlink communications, such as the *SimuLTE* module [264] developed in OMNeT++, and the *LENA LTE* module [265] developed in ns-3. More recently, a module supporting cellular D2D communications in ns-3 was released by NIST [266]. The NIST LTE D2D module has extended the LENA LTE module to support sidelink communications. The module also includes the implementation of the above-mentioned error model for the V2V communications, developed by NIST [243].

To implement ML, and in particular RL algorithms, several numerical computational libraries were available, such as the widely used *Tensorflow* [167] and *Keras* [267]. There were also widely-used simulation environments available for developing, testing, and deploying the algorithms, such as *MuJoCo* [268] or *Gym* [269]. These environments simulate physical control tasks or game-playing, for which ML solutions could be applied for. However, none of them are related to communication networks. A recent tool called *ns3-gym* [270] was released, which provided an interface to ns-3 for implementing RL algorithms for networking problems, based on the Gym framework.

To implement our models, we have developed our own simulation environment also by combining and extending the several simulation tools available in the literature. As illustrated in Fig. 3.4, our simulation environment consists of three main parts: i) network simulator implemented in ns-3; mobility simulator SUMO; and iii) RL model and its training environment implemented in *Python*. We describe the details of these in the following.

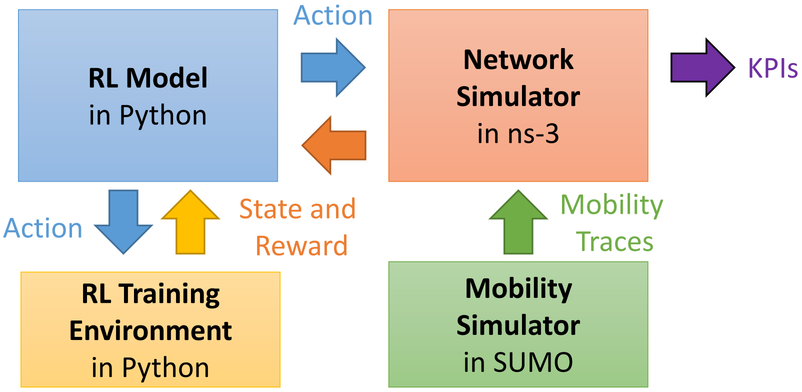


Fig. 3.4 Implemented simulation platform.

3.3.2 Vehicular Network Simulation

To simulate the vehicular network environment model described in Section 3.1, we have utilized the well-established network simulator ns-3. We also use ns-3 for the performance evaluations of the considered algorithms. Different from the rest of the chapters, the analysis and the algorithms we present in Chapter 4 are implemented and evaluated using *MATLAB* [271].

For our simulations, we have employed the above-mentioned LTE D2D module in ns-3, developed by NIST [266]. We have utilized the version *d2d-ns-3.22* that was available during the preparation of this thesis work, in combination with the ns-3 version *ns-3.22*. These software are openly-available in [272] and [273], respectively. We have implemented the following extensions in ns-3 to support full-stack V2V communication protocol based on the *Release 14* of 3GPP LTE specifications [12], which was the available release of the standard during the time of our implementation. Our extensions are summarized in Table 3.1.

At the application (APP) layer of the user nodes, we have extended the BSM application (*BsmApplication*) available in the ns-3 WAVE module, to generate and receive periodic and aperiodic types of V2V messages of certain size, as modeled in Section 3.1.1. We have also implemented the measurement of our evaluation metrics PRR, latency, and PIR, as defined in Section 3.2, by extending the *WaveBsmStats* that is used to collect and manage statistics. The calculations are based on the time stamp and vehicle IDs of the transmitted and received V2V messages at the APP layer.

At the radio resource control (RRC) layer of the user nodes (*LteUeRrc*), we have implemented resource pool configuration as modeled in Section 3.1.3. At the medium access control (MAC) layer of the BS (*LteEnbMac*), we have implemented the scheduling functionality in line with the LTE resource allocation mode 3 as described in Section 2.3.3. While

Table 3.1 Overview of the main changes and extensions we introduced to the NIST LTE D2D module in ns-3.

ns-3 class	**Our extensions to the NIST LTE D2D module in ns-3 [266]**
BsmApplication	- Generation and reception of periodic and aperiodic V2V messages
WaveBsmStats	- Measurements of PRR, PIR, and latency of V2V messages
LteUeRrc	- Configuration and processing of sidelink resource pools
LteEnbMac	- Scheduling of V2V transmissions by connecting to RL model and sending the collected information related to state and reward of the RL model - Indicating the scheduling assignments to vehicles via DCI
LteUeMac	- Processing of DCI in the case of centralized scheduling of V2V messages - Resource selection mechanism for the autonomous scheduling of V2V transmissions based on LTE mode 4
LteUePhy	- Sensing mechanism for the autonomous scheduling of V2V transmissions based on LTE mode 4 - V2V transmission power level
PropagationLossModel	- WINNER+ B1 channel model for V2V communications
IsotropicAntennaModel LteSpectrumPhy	- Antenna related parameters for V2V communications
LteSpectrumValueHelper	- Carrier frequency and bandwidth for V2V communications

we reuse the scheduling requests of vehicles on the UL control channel, the DL control information (DCI) message is modified to indicate the resource allocation from the configured resource pool for V2V messages. Whenever a vehicle sends a resource allocation request, the scheduler calls the RL model in Python via the socket connection we implemented (cf. Section 3.3.4), and transmits the necessary state and reward information based on the information it collected from the network such as past resource allocations, vehicle locations, PRR, etc. In turn, it receives the action selection of the RL model and transmits this info to vehicles in the form of DCI indicating the selected subchannel and time slot in the resource pool.

At the MAC and the physical (PHY) layer of the user nodes (*LteUeMac* and *LteUePhy*), we have implemented the LTE resource allocation mode 4 that consists of resource sensing and selection mechanisms as described in Section 2.3.3, utilizing the resource pool configured at the RRC layer.

To align with the 3GPP evaluation assumptions [236] as described in Section 3.1.4, we have adjusted the parameters of the pathloss and fading model under *PropagationLossModel* for the V2V channel model, *IsotropicAntennaModel* and *LteSpectrumPhy* for the antenna parameters, and *LteUePhy* for the transmission power levels. We have defined a new carrier frequency and operation bandwidth for the V2V communications in *LteSpectrumValueHelper*.

3.3.3 Vehicular Mobility Simulation

To simulate the mobility of vehicles, we have used the openly available realistic road traffic simulator SUMO [247]. We have created the considered highway section described in Section 3.1.5 using the graphical road-network editor of SUMO, called *netedit* [274]. In particular, we created an *edge* (road section) in between two *nodes* (road junctions), as well as the number of lanes per direction and the maximum speed allowed on the road. On this road, we introduce the vehicular traffic via *flow* definitions of SUMO, which create repeated vehicle emissions at each junction. We define a flow per road direction by specifying the vehicle type and the probability of emitting a vehicle each second. The probability determines the rate of vehicle arrivals based on the Poisson distribution in our model. Vehicle type is specified by the length, maximum speed, and maximum acceleration and deceleration of vehicles that control the car-following model. For these parameters, we use the default settings for the vehicle type of *passenger car* provided in SUMO [275]. The speed of each arriving vehicle is selected from a normal distribution, with a mean value that can be specified as a factor of the maximum speed limit, and the standard definition, according to the settings we consider.

The vehicle flows are simulated with a specified duration, from which *floating car data (FCD)* is generated. FCD contains the location and speed of vehicles along with other information for every vehicle at every time step, starting from the time they are generated at the junctions, with a granularity of 1 s and a precision of 1 cm. The built-in *TraceExporter* function of SUMO converts FCD output to a *trace file* format that is readable by ns-3 [276]. We generate the trace files from the time when steady-state road traffic is reached in each simulation, i.e., after excluding the warm-up period. We input the created trace files from SUMO to ns-3 for each network simulation. *Ns2MobilityHelper* function in ns-3 takes the location, speed, and direction of vehicles as input to configure the mobility of "nodes" in the network [277]. While vehicles are created and removed at different times continuously during a simulation in SUMO, all of the network nodes in ns-3 are need to be created before any simulation of the network. We, therefore, create a sufficient number of nodes in ns-3 to accommodate all vehicles generated in SUMO, while turning their radio functionality on or off as they enter or leave the considered simulation area, respectively.

3.3.4 RL Model Implementation

We have implemented our RL models presented in Chapters 5-8 using Python [278]. We have used the openly available TensorFlow library for machine learning purposes, such as to create neural networks or compute gradients, and *NumPy* [279] for other computational purposes. For the A3C algorithm, we have used the openly available implementation in [280] as the baseline. We have modified the neural network structure, state, action, and reward definitions, as well as the training parameters according to our RL model.

To enable the interaction of the RL model with the vehicular network environment, we have coupled the RL model implemented in Python with the network simulator ns-3 written in *C*++ by using a socket programming similar to the implementation in [270]. By establishing a two-way connection, the RL model collects data (such as the information on the state of the network input to its algorithm or the reward it gets from the environment) from the simulated network. In turn, the RL model signals its actions back to the network environment.

For training our RL model, we have implemented the environment model described in Section 3.1.6 in Python. We have established the same two-way connection between the RL model and the training environment, to signal the state, action, and reward of the algorithm correspondingly.

Chapter 4

Feasibility of Centralized Resource Allocation for V2V Communications in Predictable Out-of-coverage Areas

4.1 Motivation and Contribution

In this chapter, we explore a new approach for allocating radio resources for vehicle-to-vehicle (V2V) communications taking place in known out-of-coverage areas that are delimited by the cellular network connectivity, namely the DOCA as defined in Section 3.1.2. As discussed in Chapter 1 and Section 2.4.1, we are motivated by the fact that such coverage gaps would exist invariably during early network deployment or under unpreventable situations even in full deployment due to physical obstructions such as buildings or tunnels. Given that distributed schedulers are inefficient in terms of handling resource utilization as compared to centralized schedulers, and the conventional centralized schedulers are not designed for assigning resources beyond the network coverage, we explore *whether and how a centralized network entity can do in-advance radio resource allocation for vehicles approaching predictable out-of-coverage zones, in which vehicles communicate with each other*.

We begin this chapter by evaluating the required amount of resources that needs to be reserved for V2V services that rely on event-triggered messages. Since events generating this type of traffic, such as emergency braking, crash notifications, etc., occur rather unpredictably, the V2V messages can not be pre-scheduled. Instead, vehicles can utilize the reserved resources for them to achieve certain rate of collision-free transmissions. We evaluate the behavior of the required amount of resources with respect to vehicle density and DOCA size,

Parts of the text and parts of the results in Chapter 4 have been published in [33] ©2018 IEEE.

for different reliability targets. Our results indicate that the required amount of resources depends majorly on the rate of the V2V message traffic, while an increase in the reliability target does not have such a significant impact.

On the other hand, while the same reservation approach can be also applied for the periodic type of V2V data traffic, more efficient allocation of resources becomes possible since characteristics of this type of traffic are rather deterministic and known beforehand. Following this idea, for the periodic type of V2V traffic, we propose a centralized heuristic scheduler that pre-schedules the resources to the vehicles before they leave the coverage and enter the DOCA. The scheduling decisions are based on the predicted future locations of vehicles, which, along with propagation conditions, determine the interference on a specific resource, as well as the HD errors. For this purpose, the centralized scheduler collects information about the vehicles such as their velocity, density, and message traffic. We evaluate how imperfections in predicted locations impact the performance of V2V communications, such as in terms of the rate of successful receptions, in comparison to an ideal case where the perfect knowledge of vehicle locations in DOCA is assumed to be known. We evaluate the performance of V2V communications under varying densities of vehicular traffic, different distributions of vehicle speeds, various transmission powers, and DOCA sizes. Our results show that the rate of successful transmissions gets highly impacted by the prediction errors when combined with varying conditions in the out-of-coverage area.

This chapter aims at analyzing the boundaries of our resource allocation problem, where we consider reservation of resources for aperiodic V2V traffic, and pre-scheduling of resources for periodic V2V traffic, assuming perfect and imperfect predictions of vehicle locations. For our evaluations in this chapter, we utilize the model defined in Section 3.1.6 that has simple mobility and radio propagation, as our main focus is a feasibility analysis, neglecting the intricate impacts of the wireless environment. Our results indicate that while the idea of reserving and pre-scheduling resources for V2V communications taking place in known out-of-coverage areas is feasible, the resource allocation task should carefully consider the mobility conditions as well as the data traffic load of the vehicles, which serves us as a guideline for the remaining chapters.

In the rest of this chapter, we present our proposed methods for resource reservation and pre-scheduling in Section 4.2. Section 4.3 presents our evaluations results. Finally, Section 4.4 concludes the chapter.

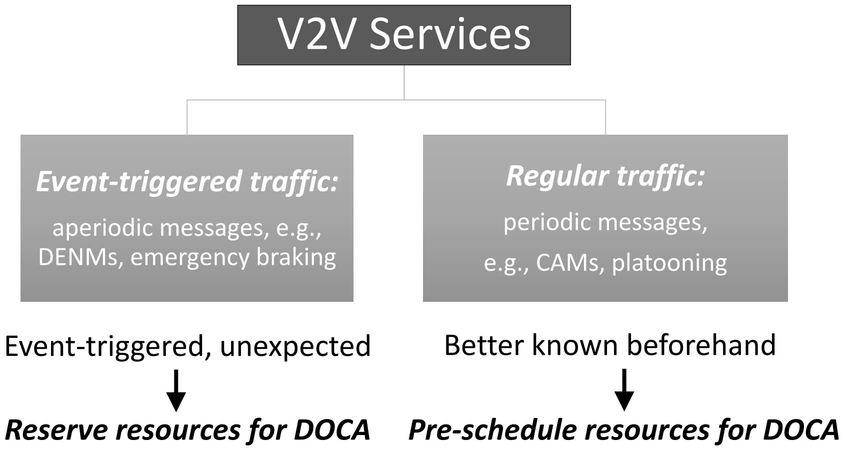

V2V Services
Event-triggered traffic:
aperiodic messages, e.g., DENMs, emergency braking
Regular traffic:
periodic messages, e.g., CAMs, platooning
Event-triggered, unexpected
Better known beforehand
Reserve resources for DOCA
Pre-schedule resources for DOCA

Fig. 4.1 Proposed approach to handle different V2V services.

4.2 Proposed Method

We propose a centralized entity to manage the radio resources for V2V communications, which particularly requires access to the road and the message traffic information. The BSs delimiting the DOCA are proposed to collect this information from the vehicles entering (exiting) the DOCA. The collected information is then used to make decisions by the centralized scheduler. Our solution regarding the radio resource management comprises of two main parts, as summarized in Fig. 4.1: i) resource reservation for the event-triggered services with aperiodic V2V traffic in DOCA; and ii) pre-scheduling the regular services having periodic V2V traffic in DOCA.

4.2.1 Resource Reservation for Event-Triggered Services in DOCA

As event-triggered messages can not be pre-scheduled before the vehicles enter the DOCA, we propose the centralized entity to reserve a portion of the available radio resources for such services, in order to still reliably support them. In order to calculate the amount of resources that needs to be reserved, we use the following formulation.

Recall that we have defined the probability of non-overloading of resources in Section 3.2 (Eq. 3.4) as:

$$\mathrm{Rel} = e^{-\lambda} \sum_{k=0}^{R-1} \frac{\lambda^k}{k!}. \tag{4.1}$$

where R is the number of available resources and λ is the arrival rate of services. For aperiodic V2V traffic,

$$\lambda = \gamma \times min(2R_{\mathrm{Tx}}, L_{\mathrm{DOCA}}) \times \lambda_{\mathrm{evt}}, \tag{4.2}$$

assuming a single, one-dimensional collision domain, where γ is the vehicle density within DOCA given by number of vehicles per unit distance, and λ_{evt} is the probability of an event per unit distance leading a vehicle to generate an event-triggered V2V message. Depending on its size L_{DOCA} and the transmission range of the vehicles, the DOCA may contain one or more collision domains. As described in Section 3.1.6, we model the transmission range of vehicles as fixed circular range with radius R_{Tx} in this chapter, assuming the protocol model. Based on this, we distinguish between two cases:

Case I: DOCA is a single collision domain

In this case, we have $L_{\mathrm{DOCA}} \leq 2R_{\mathrm{Tx}}$, hence Eq. 4.2 becomes $\lambda = \gamma \times L_{\mathrm{DOCA}} \times \lambda_{\mathrm{evt}}$. Within the DOCA, the transmissions will interfere with each other if they use the same TB.

Case II: DOCA is not a single collision domain

In this case $L_{\mathrm{DOCA}} > 2R_{\mathrm{Tx}}$, and λ does not grow above $2R_{\mathrm{Tx}}$, i.e., $\lambda = \gamma \times 2R_{\mathrm{Tx}} \times \lambda_{\mathrm{evt}}$. Instead, Poisson arrivals follow a memory-less property for each collision domain, and different V2V transmissions within DOCA can use the same TB, if they are taking place far enough from each other (i.e., at different collision domains). In other words, in Case II, the spatial reuse of radio resources is possible.

If we solve Eq. 4.1 for R, we determine the required amount of resources needed to achieve a given target reliability $\mathrm{Rel}_{\mathrm{target}}$ for the event-triggered services.

4.2.2 Pre-scheduling the Periodic Services in DOCA

In the case of regular periodic V2V messages, instead of static reservation of resources, the vehicles can be provided with a pre-scheduled resource assignment that they dynamically utilize, as characteristics of these transmissions (such as timing and periodicity) are pre-known. For this purpose, we propose that the centralized entity determines the pre-schedule by utilizing the information sent by the vehicles along with their scheduling requests (SRs). Such information is requested and collected by the BSs before vehicles enter the DOCA.

SR contains the identifications (IDs) of the transmitter and the receiver vehicles (in the case of one-to-one transmissions), their current position and velocities, as well as T_{m} of the V2V messages to be transmitted in DOCA. Based on the collected information, the centralized entity predicts the future trajectories of the vehicles for the time they will be inside DOCA. Predicted location information is then used to determine the pre-schedule for each vehicle, as we describe in the following.

Regarding Case I, the pre-scheduling task is trivial. Namely, for each requested transmission, the scheduler can only assign a new TB in order to avoid any collision with the other transmissions taking place inside the DOCA. Considering Case II, reuse of the TBs is possible among different collision domains within the DOCA, which requires a decision mechanism reliably assigning them. We elaborate on the latter, as follows.

For each incoming SR, the scheduler goes through the requested transmissions starting from the first arrived one, and attempts to assign each transmission to a TB that does not violate the constraints of reliability and half-duplex, based on the protocol model. Specifically, starting from the first among K subchannels at the requested time to transmit, a TB is assigned if all of the following apply: i) the targeted receiver vehicle is within the transmission range R_{Tx} of the transmitter vehicle; ii) both vehicles are not previously scheduled for any other reception or transmission (half-duplex constraint); and iii) no other vehicle scheduled for a transmission in that TB is closer than R_{Tx} to the receiver vehicle; and iv) the transmitter vehicle is not within R_{Tx} of a vehicle that was previously scheduled for another reception in that TB. In case none of the K subchannels are available at the requested time to transmit, the scheduler continues by checking the TBs in the next time slot, and repeats the checking process until a TB satisfying the all conditions is found. Note that the outcome of the schedule would delay each transmission by the number of time slots that had to be skipped during the check.

An example schedule is shown in Fig. 4.2, considering a simple case with a DOCA having a single-collision domain. Vehicles A, B and C send SRs to the BS respectively, requesting transmissions with different T_{m}, which are all assumed to collide if assigned to the same TB. Therefore, the second message of vehicle B, B_2, is scheduled in the next available subchannel at f_1. C_2, requested for t_4, could only be scheduled in the next slot (t_5, f_0) since all subchannels are occupied at the requested time slot, hence it experiences a delay of 1 time slot.

To account for the the maximum amount of tolerable latency $T_{\max}$ for V2V messages, we introduce another constraint to the pre-schedule. If a message has to be delayed for a duration larger than $T_{\max}$, then the message is dropped, i.e., not admitted to the schedule. Such a situation may happen when there is a high demand on the radio resources among the vehicles, e.g., due to a larger γ.

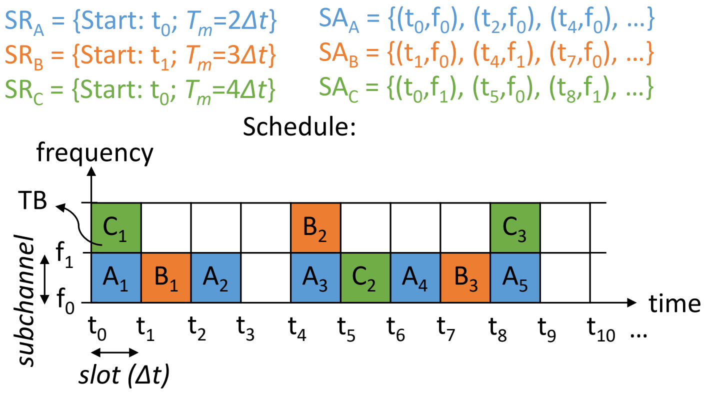

SR_A = {Start: t_0; T_m=2Δt}
SR_B = {Start: t_1; T_m=3Δt}
SR_C = {Start: t_0; T_m=4Δt}
SA_A = {(t_0,f_0), (t_2,f_0), (t_4,f_0), ...}
SA_B = {(t_1,f_0), (t_4,f_1), (t_7,f_0), ...}
SA_C = {(t_0,f_1), (t_5,f_0), (t_8,f_1), ...}
Schedule:
frequency
TB
subchannel
f_1
f_0
C_1
B_2
C_3
A_1
B_1
A_2
A_3
C_2
A_4
B_3
A_5
time
t_0 t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} ...
slot (Δt)

Fig. 4.2 An example schedule on the radio resource grid according to scheduling requests (SRs) sent by vehicles A, B and C. Vehicles are informed about their schedule by the scheduling assignments (SAs) sent via the base stations (adapted from [33] ©2018 IEEE).

The scheduler informs the vehicles about the schedule by sending scheduling assignments (SAs) timely before they enter the DOCA. SA is an array of values indicating the allocated subchannel and slots, as exemplified in Fig. 4.2, where vehicles look up the TBs to transmit the messages during their traversal of DOCA. On the other hand, they inform the BSs about their exit from DOCA, so that the scheduler can better adapt the schedule for future transmissions, e.g., by re-allocating the emptied resources.

4.3 Evaluation

In our evaluations, we utilize the system model as described in Sections 3.1 and 3.1.6 with the parameters provided in Table 4.1 with their default values. We vary the value of the parameters in our analysis on resource reservation as well as to evaluate their impact on the scheduling performance in the following subsections.

Except for our results in Section 4.3.1 where we analytically solve Eq. 4.1, evaluations in the following are based on the measurements collected from DOCA for a sufficiently large duration to have statistically meaningful results, starting after which a steady-state level of the road and the message traffic are reached in the simulated setting.

4.3.1 Resource reservation for event-triggered V2V services

Figure 4.3 shows the result of numerical simulations for the required number of resources *R* to support event-triggered V2V services under given reliability requirements, by solving

Table 4.1 Simulation parameters (adapted from [33] ©2018 IEEE).

Length of the DOCA, L_{DOCA}	1000 m
Probability of events triggering V2V messages, λ_{evt}	0.05 events/vehicle/m
V2V transmission range, R_{Tx}	75 m
Arrival rate of vehicles at the DOCA, λ_{arr}	3 vehicles/s/direction
Slot duration	25 ms
Number of subchannels, K	5
Message periodicity of each vehicle, T_{m}	$\{25, 50, 75\}$ ms with equal probability
Maximum allowed latency, $T_{\max}$	100 ms

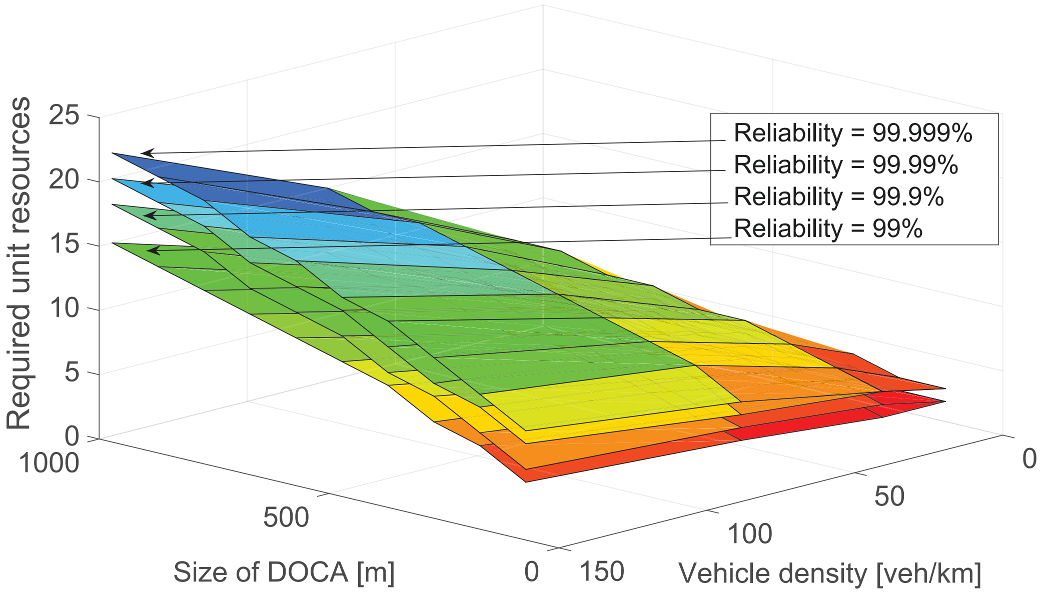


Fig. 4.3 Required unit resources R as a function of DOCA size L_{DOCA} and vehicle density γ, with respect to different reliability targets [33] ©2018 IEEE.

Eq. 4.1. Specifically, we assume a perfect resource allocation: one that assigns the V2V messages in non-overlapping resources without any scheduling overhead.

The results indicate that the increase in reliability does not penalize the system prohibitively. This is in contrast with the efficiency penalty on the physical layer, where increase in reliable transmissions would be costlier in terms of the spectral efficiency [281]. Furthermore, Fig. 4.3 shows that γ, as well as L_{DOCA}, have a more significant effect on the required resources than the target reliability. The results provide design guidelines for a DOCA resource allocation, which should be sensitive to vehicle density changes and adapt both the amount of resources reserved as well as the schedule according to the vehicle density and mobility in DOCA.

4.3.2 Impact of the Predictions and Vehicle Velocities on the Pre-Scheduling Performance

In this subsection, we evaluate the performance of the proposed pre-scheduling algorithm, impacted by the imperfections of the predicted vehicle velocities used to determine the interference conditions on the resources. Rather than concentrating on how the predictions are made, we analyze the consequences of different types of predictions on the scheduling performance, where the vehicle velocities are predicted to be less than, equal to or over the actual values.

In our evaluations, we consider two mobility scenarios: a) all vehicles travel with a constant speed of 30 m/s; and b) vehicles have random constant speeds uniformly distributed between 20 and 30 m/s. For the first scenario, we evaluate the performance of the scheduler when the vehicles are predicted to have the same constant speed of 5, 15, 30, 35 and 45 m/s, as well as random constant speeds uniformly distributed between $5-15$, $15-25$, $25-35$, $35-45$ and $45-55$ m/s. For the second case, we evaluate the effect of vehicles being predicted to have the same constant speed of 5, 15, 25, 35 and 45 m/s, together with the predictions of random constant speeds distributed uniformly between $5-15$, $20-30$, $25-35$, $35-45$ and $45-55$ m/s.

In the considered setting, all pre-scheduled services are assumed to be periodic unicast (one-to-one) messages. In particular, each vehicle has a message traffic with random T_{m}, to be transmitted to the vehicle following behind it at the time it is entering the DOCA, and desires to maintain this communication for the rest of the time they are inside the DOCA together. Such transmissions represent the typical use case of platooning vehicles, where each member of a group vehicles driving together sends a unicast message to inform the one behind [136].

We evaluate the performance of the scheduler in terms of the rate of successful receptions, i.e., PRR within R_{Tx}, and the rate of unsuccessful receptions classified with respect to type of errors, as well as other KPIs, which we list in Table 4.2. The results are provided in Fig. 4.4 and 4.5, respectively for the scenarios (a) and (b), for different cases of predicted and actual velocities of the vehicles as described above.

As expected and can be seen from Fig. 4.4 and 4.5, correct predictions achieve the largest *Sch'd & Successful*, hence *Successful Transmission Rate*. Accordingly, both KPIs decrease with the predicted speeds deviating from the actual values. To illustrate, when vehicles are predicted to be all traveling at 35 m/s instead of their actual speeds of 30 m/s, *Successful Transmission Rate* decreases around 40%.

Note that even with correct predictions not all transmissions could be scheduled (i.e., *Admission Rate* is less than 1). This is because for some Rx-vehicles, it is not possible to

Table 4.2 Scheduling KPIs (adapted from [33] ©2018 IEEE).

Percentage of transmissions classified as:
Sch'd & Successful: scheduled, transmitted and successfully received, i.e., PRR within R_{Tx}
Sch'd but RxIsFar: scheduled and transmitted, however the Rx-vehicle is actually outside the transmission range of the Tx-vehicle, hence not successfully received, i.e., propagation error
Sch'd but RxRecInterf: scheduled and transmitted, however the Rx-vehicle is actually subject to interference resulting in unsuccessful reception, i.e., collision error
Drop'd & RxIsFarIndeed: not admitted to the schedule since the Rx-vehicle is predicted to be outside the transmission range of the Tx-vehicle, and this turns out to be true
Drop'd dueRxIsFar butNot: not admitted to the schedule due to the previous reason, however the Rx-vehicle is actually traveling within the transmission range of the Tx-vehicle
Drop'd Else: not admitted to the schedule due to any other reason, e.g., Rx-vehicle is predicted to receive interference at that time instance, or due to HD constraint
Other KPIs:
Admission Rate: the ratio of the number of scheduled transmissions to the total number of requested transmissions
Successful Transmission Rate: the ratio of the number of successful transmissions that were requested, to the number of transmissions admitted in the case of a correct predictor (correctly predicting the actual velocities of the vehicles)
Average Latency: the mean value of the latency experienced among all scheduled transmissions, in ms

schedule them given the system constraints K and T_{max}, without any interference during at least some part of their time within DOCA, or they might not be within the transmission rage of the Tx-vehicle. It can be observed for the correct predictions in Fig. 4.5 that both occasions rise, as the relative speeds of the vehicles increased.

Considering the cases where vehicles are predicted to be slower, the percentage of *Drop'd Else* considerably increases, besides the transmissions *Sch'd but RxRecInterf*, all due to the errors in the predicted positions of the interferers. Consequently, *Admission Rate* can drop below 0.5 if the velocities are predicted as low as 5 m/s.

On the other hand, when the vehicles are predicted to be faster, *Sch'd but RxRecInterf* are present with larger percentages, in addition to the occurrences of *Drop'd & RxIsFarIndeed* and *Drop'd dueRxIsFar butNot*. This can be explained by our assumption that each vehicle transmits to the vehicle following itself. If the vehicle entered the DOCA is predicted to be faster, then the corresponding Rx-vehicle is thought as being left far behind it, hence the messages are (erroneously) dropped. Similarly, interferers are also thought to be away from the Rx-vehicles, resulting in higher *Admission Rates*.

For the cases of vehicles having different relative speeds, as provided in Fig. 4.5, the percentage of *Drop'd & RxIsFarIndeed* is more pronounced than *Drop'd dueRxIsFar butNot*, due to Rx-vehicles now being able to overtake their Tx-vehicles, and even moving farther

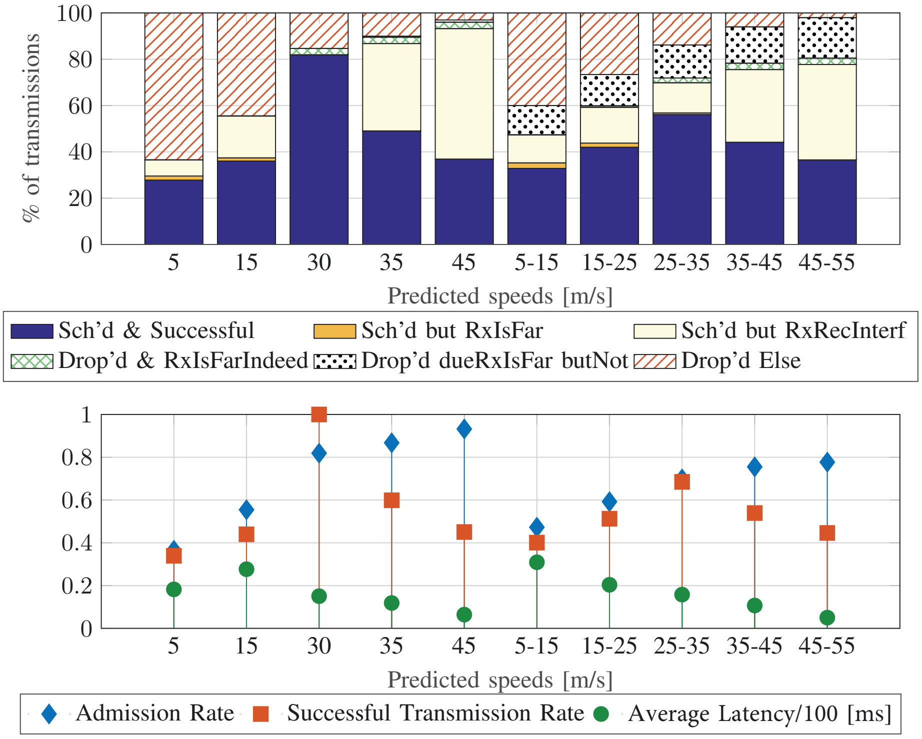


Fig. 4.4 Impact of speed predictions on the scheduling performance. All vehicles have the same speed: 30 m/s (adapted from [33] ©2018 IEEE).

than their transmission rage apart. This also results in considerable percentage of *Sch'd but RxIsFar*, especially if the vehicles are all predicted as having the same speed.

Regarding *Average Latency*, it is interesting to observe the trend where it decreases by predicting the vehicles to be faster. Such predictions assume less collisions, resulting in more admissions to the schedule, hence the transmissions experience less delay (although they eventually collide).

4.3.3 Impact of the V2V transmission range

In this subsection, we evaluate the impact of the V2V transmission range R_{Tx} on the schedule. The range is practically determined according to many factors in the network, such as the transmit power of the vehicle antenna, propagation losses and fading on the radio channel. Thus, it is possible for the system to operate under many different R_{Tx}. Here, we consider the cases of $R_{\mathrm{Tx}} = 45$ m and $R_{\mathrm{Tx}} = 100$ m, reported in and Fig. 4.6 and Fig. 4.7, respectively, in addition to the case of $R_{\mathrm{Tx}} = 75$ m in the previous section (Figures 4.4 and 4.5), and compare the measured KPIs of the schedules based on them.

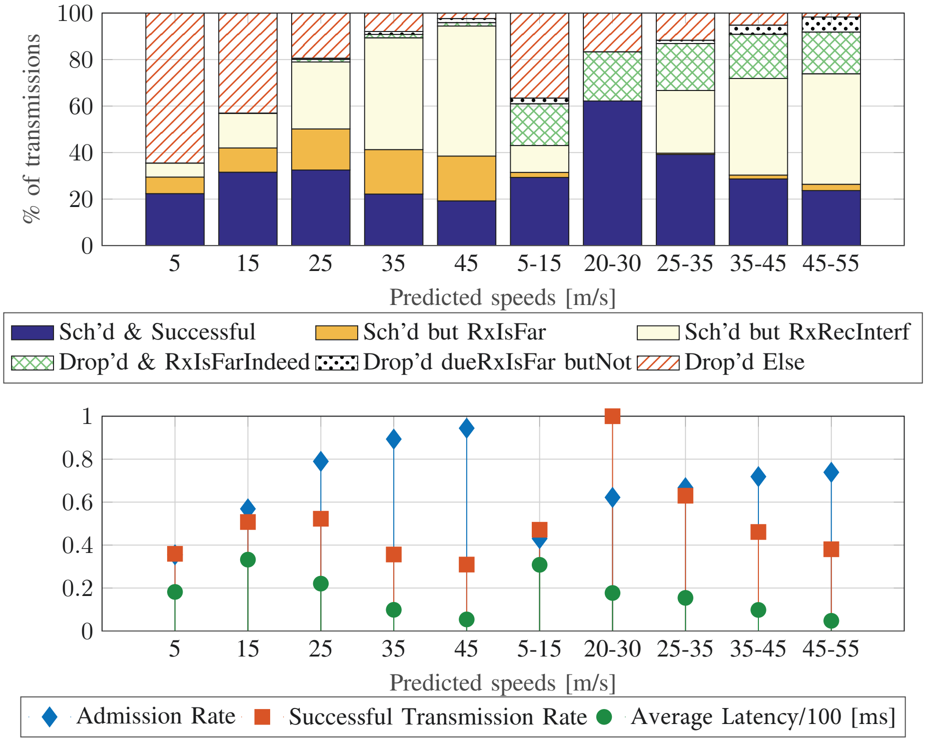


Fig. 4.5 Impact of speed predictions on the scheduling performance. Vehicles have random constant speeds, uniformly distributed between 20 and 30 m/s (adapted from [33] ©2018 IEEE).

The transmission range is also referred to as “interference” range, since it determines the size of the area where any vehicle inside will receive interference from the transmitter vehicle, if it is not the intended receiver. Accordingly, increasing this range increases the number of vehicles receiving interference, given the same vehicle density on the road. Increased size of the transmission range considerably increases the ratio of dropped transmissions due to predicted interference (*Drop'd Else*), in addition to yielding higher average latency in the scheduled transmissions. Similarly, the ratio of scheduled transmissions suffering from unforeseen interference (*Sch'd but RxRecInterf*) also increases.

A smaller V2V transmission range also results in larger ratio of transmissions that can not be received by the Rx-vehicles that are further away from the Tx-vehicles. In case of vehicles traveling with the same speed, predicting them to have different speeds increases the ratio of transmissions that are dropped (*Drop'd dueRxIsFar butNot*) due to predicting the Rx-vehicle likely to move away from the receiver. Contrarily, when vehicles have different speeds, predicting them to have the same speed increases the ratio of scheduled transmissions being not received due to Rx-vehicle actually moving away from the Tx-vehicle (*Sch'd but RxIsFar*). In the case of correct predictions, larger ratio of transmissions are dropped due to

Rx-vehicles residing outside the V2V transmission range (*Drop'd & RxIsFarIndeed*) when the range is smaller.

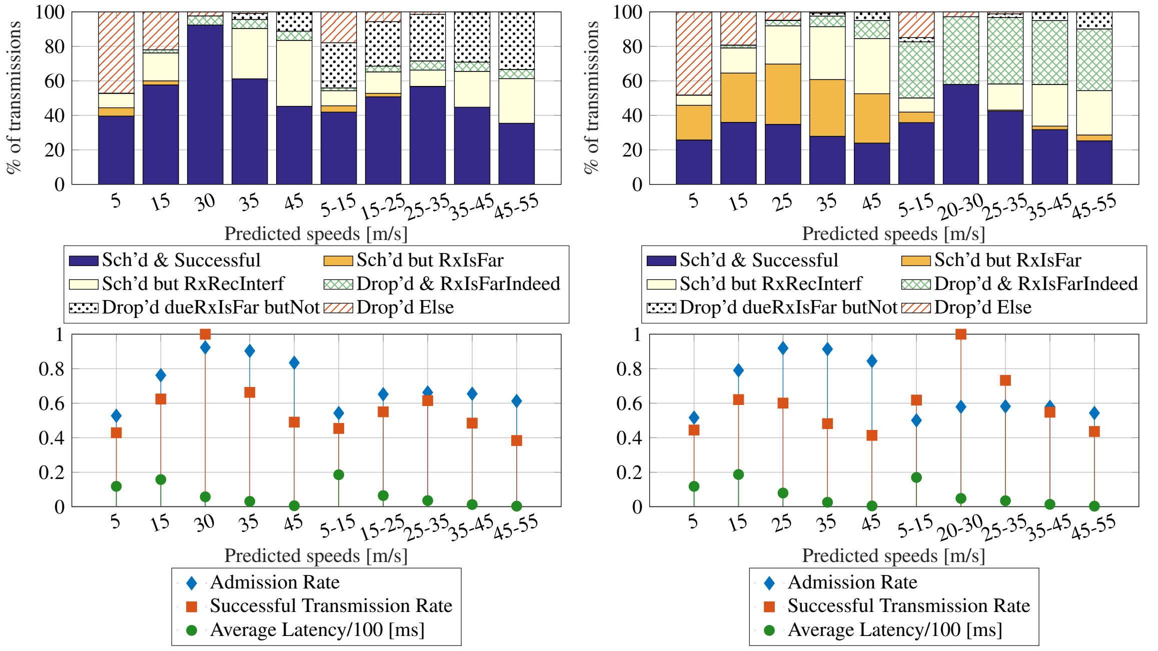
4.3.4 Impact of the Vehicle Density

In this subsection, we evaluate the impact of the rate λ_{arr}, the number of vehicles arriving in both directions per second, on the schedule. From this rate, it is possible to determine the vehicle density γ on the road, namely the two-way traffic volume in terms of the number of vehicles per a unit section of the highway, given a constant flow. Traffic volume is an important parameter in the design of transportation systems, regarding, e.g., the capacity of the roads. Even for the same road, it takes different values on an hourly or a seasonal basis.

In our case, we are concerned about the radio resource usage by the vehicles on a given highway segment, which is mainly based on the requested number of transmissions. In fact, demand on the radio resources is proportionally related to λ_{arr}. Correspondingly, for a fixed value of λ_{arr}, increasing the frequency of V2V message traffic would also create the same effect on the system. We are interested in how the schedule gets affected by different vehicle densities. For this, KPIs are provided for the values of $\lambda_{\text{arr}} = 1$ and $\lambda_{\text{arr}} = 5$, in Fig. 4.8 and 4.9, respectively, besides our previous results for $\lambda_{\text{arr}} = 3$ in Figures 4.4 and 4.5. Other system-level parameters are kept constant, i.e., $L_{\text{DOCA}} = 1000$ m and $R_{\text{Tx}} = 75$ m.

It is clear that, with the increased density of vehicles, the ratio of the transmissions not admitted to the schedule increase (i.e, *Drop'd Else*), mostly due to the predicted interference, in addition to the larger average latency incurred in the schedule. Erroneous predictions in case of higher density of vehicles result in increased rate of scheduled transmissions that are not successfully received (i.e., *Sch'd but RxRecInterf*). The proportion of vehicles suffering from interference due to any transmission taking place essentially increase as in the case of increased size of the V2V transmission range, however this time due to larger number of vehicles within the same transmission range.

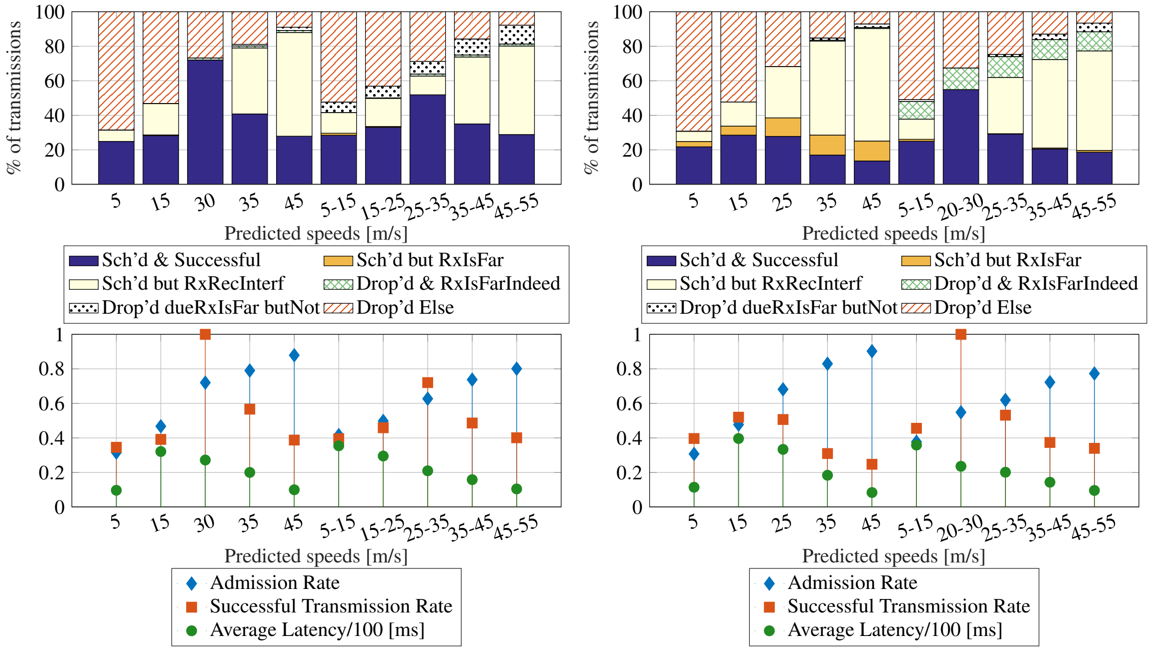
On the other hand, lower road traffic density might result in larger proportion of intended Rx-vehicles residing outside the transmission range of Tx-vehicles, thus not necessitating scheduling such transmissions. Fig. 4.8 shows a considerable proportion of such transmissions (i.e., *Drop'd & RxIsFarIndeed*) in case of $\lambda_{\text{arr}} = 1$. Similarly, predicting Tx-vehicles to be slower or faster result in increased ratio of transmissions that are scheduled however not received by the Rx-vehicle or dropped unnecessarily, respectively (i.e., *Sch'd but RxIsFar* and *Drop'd dueRxIsFar butNot*).



(a) All vehicles having the same speed of 30 m/s.

(b) Vehicles having different speeds uniformly-random distributed between 20 and 30 m/s.

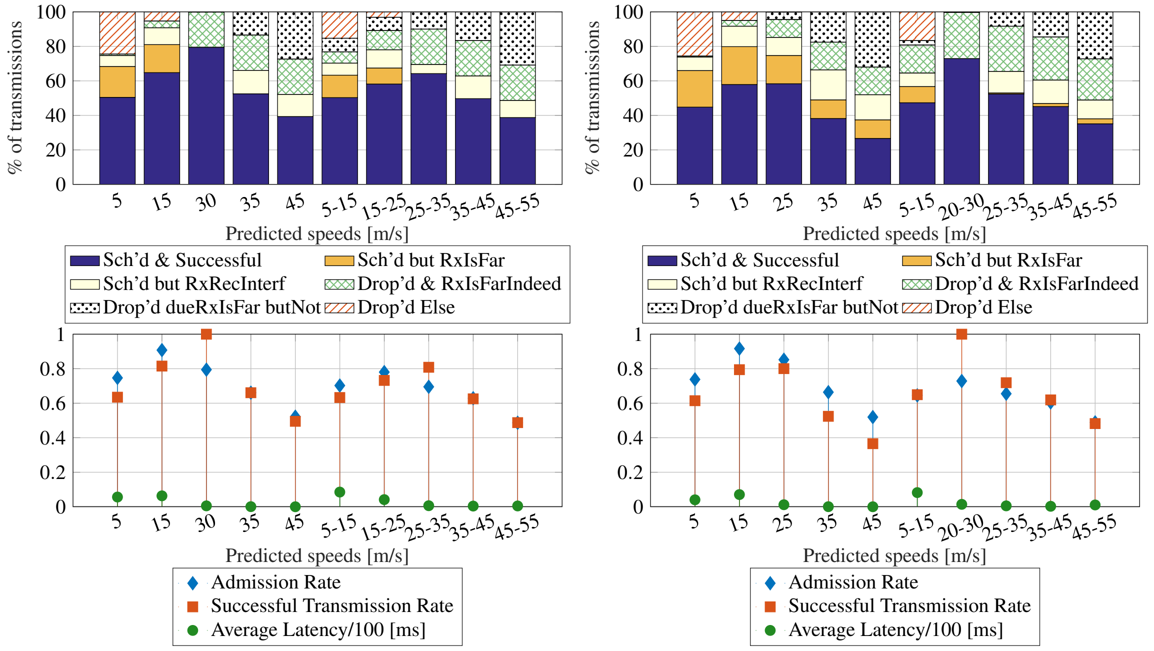
Fig. 4.6 Impact of speed predictions on the scheduling performance with V2V transmission range of $R_{\mathrm{Tx}} = 45$ m.



(a) All vehicles having the same speed of 30 m/s.

(b) Vehicles having different speeds uniformly-random distributed between 20 and 30 m/s.

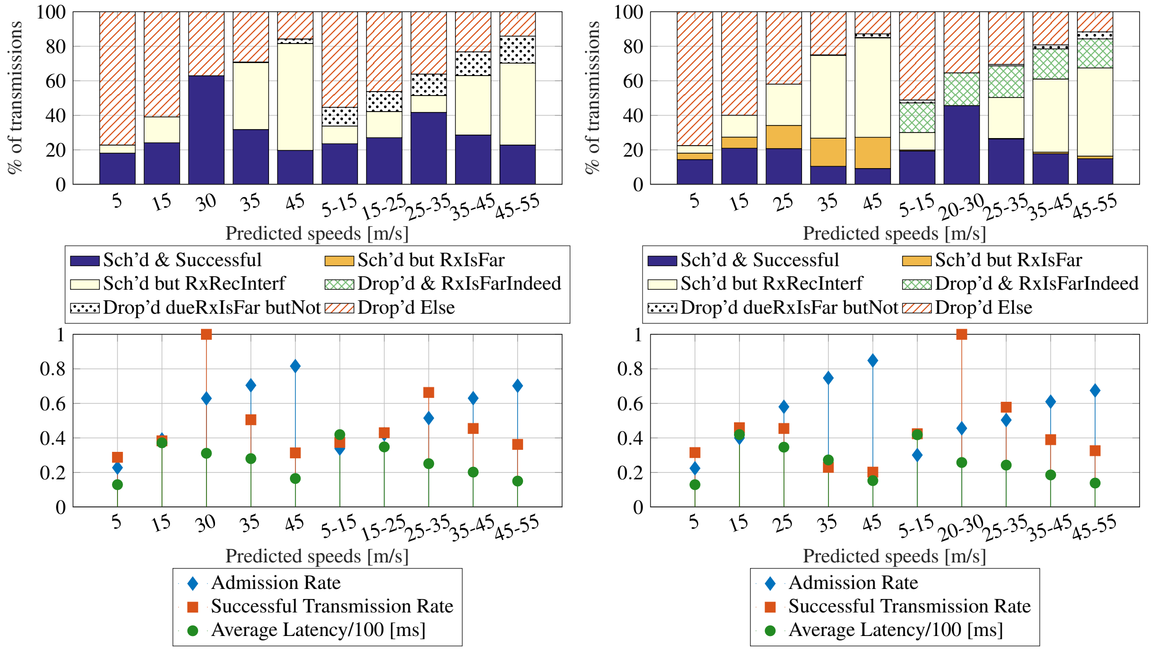
Fig. 4.7 Impact of speed predictions on the scheduling performance with V2V transmission range of $R_{\mathrm{Tx}} = 100$ m.



(a) All vehicles having the same speed of 30 m/s.

(b) Vehicles having different speeds uniformly-random distributed between 20 and 30 m/s.

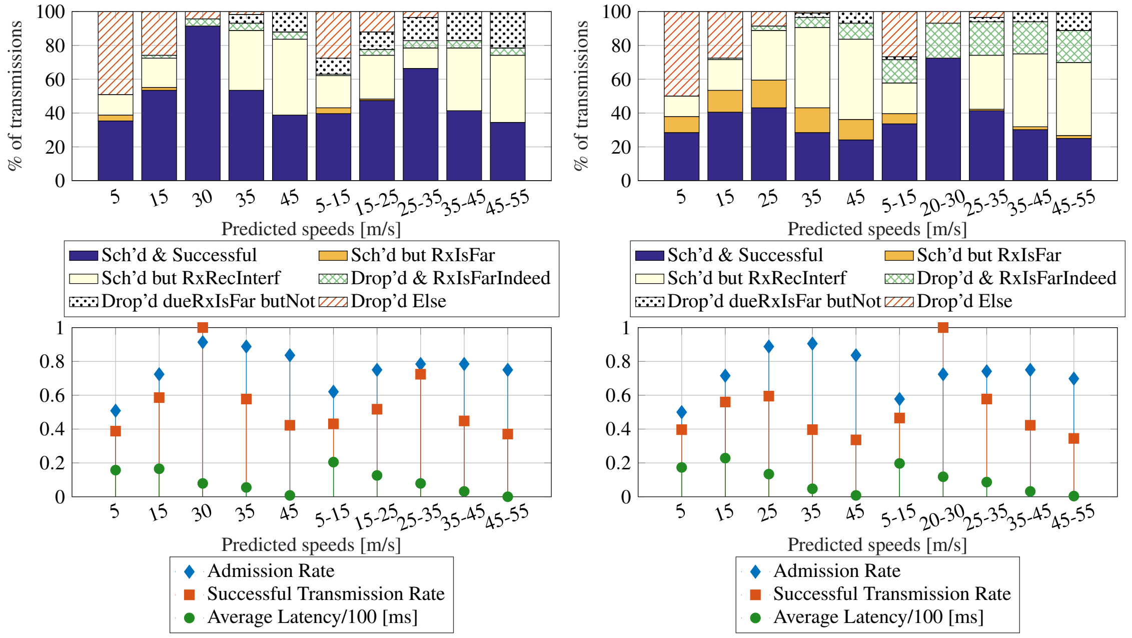
Fig. 4.8 Impact of speed predictions on the scheduling performance with the rate of arriving vehicles $\lambda_{\text{arr}} = 1$ vehicle/second.



(a) All vehicles having the same speed of 30 m/s.

(b) Vehicles having different speeds uniformly-random distributed between 20 and 30 m/s.

Fig. 4.9 Impact of speed predictions on the scheduling performance with the rate of arriving vehicles $\lambda_{\text{arr}} = 5$ vehicles/second.



(a) All vehicles having the same speed of 30 m/s.

(b) Vehicles having different speeds uniformly-random distributed between 20 and 30 m/s.

Fig. 4.10 Impact of speed predictions on the scheduling performance for DOCA size $L_{\text{DOCA}} =$ 500 m.

4.3.5 Impact of the DOCA Size

The size of the DOCA determines the duration for which the vehicles travel inside it. The proposed scheduler estimates the future positions of the vehicles based on their predicted velocities, which is assumed to be constant over time. Accordingly, in case of wrong predictions, the deviation between the actual and the estimated positions of the vehicles increases with time, i.e., as the DOCA gets larger.

In Fig. 4.10, we report the results for the scenario with a DOCA of length $L_{\text{DOCA}} = 500$ m, which is smaller than the scenario with $L_{\text{DOCA}} = 1000$ m reported in Figures 4.4 and 4.5. We observe that a larger size of DOCA results in lower rate of successfully received transmissions, where ratio of dropped transmissions are increased in the case of wrong predictions on the positions of the vehicles. In the case of perfect predictions, a larger DOCA size still results in lower rate of successful transmissions, which is due to the increased proportion of vehicles receiving higher level of interference within the DOCA.

4.4 Conclusions

In this chapter, we conducted an exploratory study on allocating resources for V2V communications in a known out-of-coverage area, namely for a DOCA. Differing from the state of the art, we proposed the resources for V2V transmissions in DOCA to be pre-allocated by a centralized network entity, whereby allocations are communicated to the vehicles via BSs before they enter DOCA. For the aperiodic type of V2V traffic (e.g., emergency braking, crash notifications, etc.), we proposed to reserve resources, since such messages can not be scheduled beforehand. We analyzed the required amount of resources to achieve the target reliability of V2V applications. Our preliminary analysis showed that the amount of reserved resources needs to be adapted with respect to vehicle density changes. Following, for the periodic type of V2V traffic, which can be pre-scheduled (e.g., CAM transmissions, platooning, etc.), we have proposed a centralized scheduler that provides a pre-schedule to each vehicle for its transmissions throughout the DOCA, before entering it. The scheduler is proposed to make predictions, e.g., regarding the vehicle positions, to allocate resources for successful receptions, based on the information it collects from the vehicles. We analyzed how imperfect predictions and variations in vehicular velocities impact the pre-scheduling performance in terms of reliability, admission rate, and latency, under varied vehicular mobility, density, wireless channel conditions, and DOCA size. Overall, our results indicate that the proposed idea of pre-allocating resources for V2V communications in expected out-of-coverage areas is feasible. However, the rate of successful transmissions gets highly impacted by the prediction errors, when combined with the varying conditions in the vehicular environment. Thus, resource allocation for out-of-coverage V2V communications in practice calls for a flexible scheduler design.

Chapter 5

Learning to Schedule V2V Communications in Predictable Out-of-coverage Areas

5.1 Motivation and Contribution

In Chapter 4, we explored the potential performance of centrally allocating the resources for V2V communications in areas outside the cellular network coverage. Our preliminary analysis showed that efficient prediction mechanisms would be necessary to make resource allocation, considering the vehicular mobility, density, traffic load, and wireless channel characteristics. In this chapter, we propose a reinforcement learning (RL) based approach to predictively schedule the resources for V2V communications outside the coverage. We are motivated by the recent success of RL in similar decision-making problems under uncertainty, as discussed in Section 2.6.

The proposed scheduler learns to allocate the resources by using the information available from the vehicular environment, such as the occupancy of radio resources, and the reward signal that we have designed to maximize the reliability of V2V transmissions in DOCA. We have utilized the state-of-the-art asynchronous advantage actor-critic (A3C) algorithm introduced in Section 2.5.3, and extended it for our continuous-task setting to train the scheduling policy that we represent by a deep neural network. The trained policy schedules the available resources for periodic V2V transmissions in DOCA for the vehicles before they exit the network coverage.

Parts of the text, and the results in Chapter 5 have been published in [34] ©2018 IEEE.

As covered in Section 2.6, application of machine learning (ML) to any resource allocation problem targeting vehicular communications is in its infancy. To the best of our knowledge, ML for a centralized scheduler managing the resources of V2V communications has not been treated in the literature yet. To exploit V2X-specific information for resource pre-allocation, we resort to RL, which was shown to apply well to a wide range of problems, such as games involving large combinatorial space, image recognition, and robot movement [164], and was recently applied to resource scheduling in vehicular networks [29] (cf. Section 2.6). We are interested to observe if and how it could be useful to satisfy the stringent requirements of V2V use cases outside coverage, such as in terms of reliability and latency. Towards this goal, in this chapter, we perform an exploratory study using several relevant V2V scenarios to investigate *if a centralized scheduler can learn to perform resource (pre-)scheduling reliably*. In particular, we evaluate the reliability performance of the proposed RL scheduler in specifically-designed sanity-check environments, such as with a certain number of resources and vehicles in a single collision domain, for which the optimal schedule is easily computable, and compare its performance with the existing resource allocation schemes. To observe the scheduling strategy it develops, we further trace and analyze the individual actions of the scheduler, as well as the convergence behavior during its training. Our results show that the proposed RL-based scheduler can achieve performance as good as or better than the state-of-art distributed scheduler, often outperforming it. Besides, the learning process completes within a reasonable time (ranging from a few hundred to a few thousand epochs), thus making the RL-based approach a promising solution for scheduling V2V communications outside the network coverage.

The rest of this chapter is organized as follows. In Section 5.2, we present our RL-based scheduler design. Section 5.3 presents the results of our evaluations. Finally, Section 5.4 concludes the chapter.

5.2 Deep Reinforcement Learning Scheduler

We design a learning scheduler that manages the V2V radio resources for DOCA, whose model is as described in Section 3.1. The scheduler assigns resources to each vehicle *before* it enters DOCA; the resources will be used by that vehicle throughout its travel in DOCA.

5.2.1 RL Model and the Training Algorithm

We apply RL to determine the scheduling policy. As introduced in Section 2.5.2, RL considers a setting where an agent is interacting with its environment by applying a policy

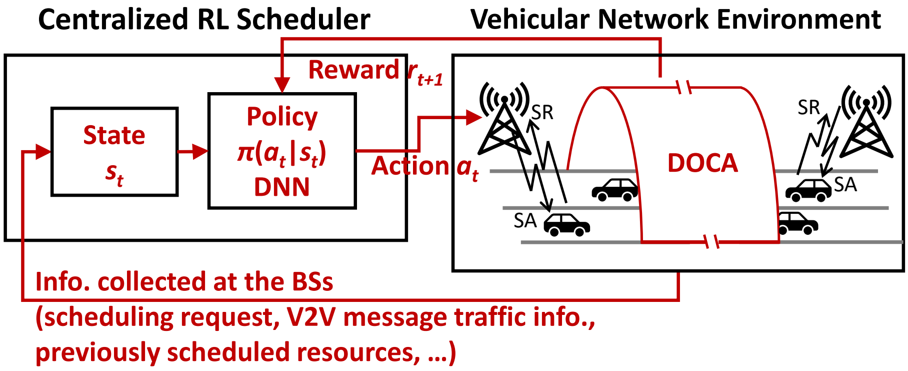

Centralized RL Scheduler
Vehicular Network Environment
Reward r_{t+1}
State s_t
Policy $\pi(a_t|s_t)$ DNN
Action a_t
SR
SR
SA
SA
DOCA
Info. collected at the BSs
(scheduling request, V2V message traffic info.,
previously scheduled resources, ...)

Fig. 5.1 RL framework applied to our scheduling problem (adapted from [34] ©2018 IEEE).

that determines the agent's behavior on selecting from the available actions, based on the available information, or perceived states of the environment.

Figure 5.1 depicts how we apply this framework to our scheduling problem. Whenever a new vehicle is about to enter the DOCA, a new action should be taken by the agent. The action consists of assigning a single time-frequency resource, i.e., a TB, to the vehicle. The assignment is performed according to a policy $\pi : \pi(a_t|s_t) \rightarrow [0,1]$, which defines a probability distribution over the set of available actions, namely selecting one of the R TBs.

Given the possible number of resources and vehicles (both possibly in thousands or more), there are many potential pairs of $(state, action)$, making the tabular solutions infeasible for this problem [164]. We therefore propose to apply approximate solutions, where the policy is represented by a deep neural network (DNN) with a set of adjustable policy parameters θ, i.e., $\pi(a|s,\theta)$. As discussed in Section 2.5, the benefits of applying such solution are twofold: i) it makes the learning process much faster, as the number of policy parameters are typically much smaller than the number of $(state, action)$ pairs; and ii) it learns through raw observations and requires no prior information about the task in hand and the model of the environment.

To train the policy parameters, we make use of the state-of-the-art A3C algorithm defined in Section 2.5.3, which applies an actor-critic RL method. The actor-critic algorithm used in our solution involves training two DNNs, one which is used to represent the policy, referred to as the actor network, and the other one which is used to represent state values, referred to as the critic network (see Fig.5.2). The value of a state under the policy π is defined as the expected rewards received by that state in a long run. We denote by $v(s_t, w)$ the value of state s_t while following π, represented by a critic network with parameters w. The state values are used as a critic when training the policy parameters. Similar to [173], we apply policy gradient method to train the parameters of the actor and the critic networks, i.e. θ and w.

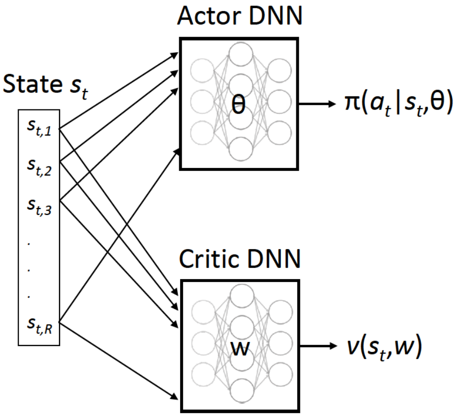


Fig. 5.2 Components of the A3C algorithm (adapted from [34] ©2018 IEEE).

Thanks to the policy gradient theorem [164], an exact expression on how the performance is affected by the policy parameters can be driven for such methods. This ensures performance improvement at each step and hence provides strong convergence properties for policy gradient methods. Besides, using separate networks to represent the state values and the policy removes the possible bias and dependencies introduced when applying policy gradient methods, which in turn accelerates the learning.

Our resource allocation problem requires a continuing task: the agent needs to allocate a resource whenever a new vehicles arrives at the DOCA. This is in contrast to the episodic tasks, where the sequence of actions could be broken down into natural episodes, such as plays of a chess game, each having a starting and a terminating state. We therefore extend the basic version of the A3C algorithm by replacing its discounted reward setting, which is suitable for episodic tasks, with the average reward formulation that is suitable for continuing problems as described in Section 2.5.3. The extended algorithm is provided as Algorithm 3.

We utilize Algorithm 3 with multiple learning actors, each interacting with a different random instance of the environment. Each instance starts with a random assignment of resources to the vehicles, and a random action taken. After a certain period of interaction and experience with the environment, called an *epoch*, each actor updates the parameters of the DNNs used for learning the policy and the state values. These parameters are then globally shared by all actors.

5.2.2 State Information, Reward, and Deep Neural Networks

We have designed the state information provided to the RL agent considering two different vehicular environments:

Algorithm 3 Extended A3C Algorithm for Continuing Problems (per actor-learner)

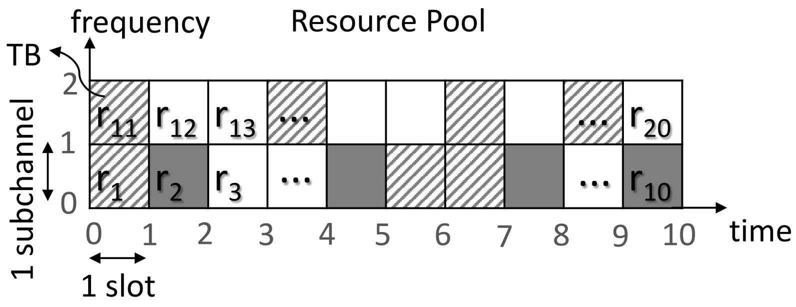
```
Initialize global shared policy parameters $\theta$ and state-value weights $w$
Initialize actor-learner-specific parameters $\theta'$ and weights $w'$, and counter $t = 1$
loop until convergence
    Synchronize actor-learner-specific parameters, $\theta' \leftarrow \theta$ and $w' \leftarrow w$
    Generate an epoch of length $T$, $\{s_0, a_0, r_1, \dots, s_{T-1}, a_{T-1}, r_T\}$ following $\pi(\cdot|\cdot, \theta')$
    for $t = 0, 1, ..., T-1$ do
        $n \leftarrow T - 1 - t$ (number of steps until end of epoch)
        Calculate $n$-step differential return, bootstrapped from the estimated value of the last reached state, $R_t = r_{t+1} + r_{t+2} + \cdots + r_{t+n} + v(s_{t+n+1}, w') - n\bar{r}$
        Calculate TD error, i.e., advantage, $\delta = R_t - v(s_t, w')$
        Update estimate of the average reward, $\bar{r} \leftarrow \bar{r} + \beta\delta$
        Accumulate gradients wrt. $w'$, $dw \leftarrow dw + \alpha\delta\nabla v(s, w')$
        Accumulate gradients wrt. $\theta'$, $d\theta \leftarrow d\theta + \alpha\delta\nabla \ln \pi(a|s, \theta')$
    end for
    Perform asynchronous update of $\theta$ and $w$ using $d\theta$ and $dw$, respectively.
end loop
```

E1) A DOCA of a single collision domain, assuming all vehicles are within the transmission range of each other. Given these conditions, reception of a message is successful if no other transmission takes place on the same radio resource scheduled (i.e., no collision), and the receiver is not scheduled to transmit at the same time, as imposed by the half duplex (HD) constraint.

E2) A DOCA of multiple collision domains, where pathloss and fading effects are taken into account. Hence, successful reception of a message requires the signal-to-interference-plus-noise ratio (SINR) at the receiver to be larger than a certain target level, which depends on the distance between the transmitter and the receiver, as well as the interference level from other transmissions using the same radio resource, besides the half-duplex constraint. Reusing the same radio resource is possible, when the transmitters are sufficiently far from each other so that the SINR does not drop below the target level.

E1 helps us to identify the scheduler performance, and the ability of RL to avoid HD constraint and assigning interfering resources, while abstracting the effects of channel conditions. Whereas, E2 enables a realistic evaluation of our proposed RL scheduler.

For E1, we design the state information to represent the number of vehicles each resource is assigned to. Given this input, the actor DNN provides the policy determining which resources to be assigned to the vehicle entering DOCA. Therefore, both state information and the policy output have the same size equal to the number of total resources $R = K \times M$ in



(a) Radio resource pool with its occupancy status represented with colors

$$s_t = [\overbrace{0, 1, -1, -1, 1, 0, \dots, 0, -1}^{R \text{ resources (TBs)}}] \longrightarrow \text{Occupancy of each TB (-1, 0 or 1)}$$

(b) State information used in E1

$$s_t = \begin{array}{c} \begin{array}{cc} \overbrace{\hphantom{1 \; \cdots \; 3 \;\; 1 \;\; 2}}^{N\text{-1 past actions}} & \overset{\text{Current vehicle requesting resource}}{\uparrow} \end{array} \\ \begin{bmatrix} 1 & \cdots & 3 & 1 & 2 & 0 \\ -1 & \cdots & -1 & -1 & 1 & -1 \\ 6 & \cdots & 2 & 12 & 7 & -1 \end{bmatrix} \end{array} \begin{array}{l} \longrightarrow \text{Time passed since last action (in s)} \\ \longrightarrow \text{Direction of the vehicle (1 or -1)} \\ \longrightarrow \text{Index } i \text{ of TB } r_i \text{ assigned to the vehicle} \end{array}$$

(c) State information used in E2

Fig. 5.3 Example representations of different state information (adapted from [34] ©2018 IEEE).

the resource pool having K subchannels and M slots. An example of state representation is provided in Fig. 5.3(a). A resource pool consisting of $K = 2$ subchannels and $M = 10$ slots (i.e., containing a total of 20 TBs $r_1, r_2, ...r_{20}$) is utilized by the scheduler. In Fig. 5.3(a), we represent the resource occupancy of the pool with colors. White colored TBs (e.g., r_3 and r_{12}) indicate that they are not assigned to any vehicles inside DOCA, striped ones (e.g., r_1 and r_{11}) indicate that a TB is assigned to a single vehicle, and dark-gray-colored TBs (e.g., TB r_2 and r_5) are assigned to more than a single vehicle. This way, we quantize the number of vehicles each TB is assigned to, which reduces the state-space considerably, and consequently accelerates the learning process. The quantized state information is still sufficient as the resources used by any number of vehicles greater than one vehicle will result in collisions, due to the assumption that all vehicles are in the transmission range of each other in E1. We represent each element of the state vector with -1 if a resource is not scheduled to any vehicle, with 0 if it is scheduled to a single vehicle, and with 1 if that TB is scheduled to more than one vehicle inside DOCA. Accordingly, the state vector of the example in Fig. 5.3(a) is $s_t = [0, 1, -1, -1, 1, 0, ..., 0, -1]$, as provided in Fig. 5.3(b). Given such s_t, one preferable action would be $a_t = r_3$, namely the scheduler assigning the TB r_3 to a new vehicle entering DOCA, which will not result in any collisions and half-duplex errors with the vehicles already traversing the DOCA.

For E2, we consider a more realistic and complex environment, where the reuse of the radio resources is possible. For such decisions to be given by the scheduler, a state information containing quantized counts of resource occupancy as in E1 would not be sufficient. It is critical to know which resource was scheduled, to which and how many vehicles, and also when it was scheduled, as vehicles travel across the DOCA using the same resource. Therefore, we utilize the following structure. The state information of E2 has a matrix structure of size 3 by N, as illustrated in Fig. 5.3(c), where each of the first $N-1$ columns contains information corresponding to each of the $N-1$ previous actions taken, and the last column representing the information about the current vehicle requesting resource from the scheduler, just before entering the DOCA. For each action or column, the first row represents the time passed since the previous action was taken, rounded to the closest integer number of seconds (e.g., 0, 1, 2, etc.). The second row is the direction of the vehicle for which the action is taken (i.e., 1 for from west to east, and -1 for from east to west). Finally, the third row is the index i of TB r_i scheduled by that action (e.g., 7, 12, etc.). For the last column, i.e., corresponding to the current vehicle requesting resource, 0 is put in the first row as the time passed, it's direction is entered in the second row, and a dummy variable -1 is inserted into the third row, as it's resource is yet to be assigned by the current action to be taken. Following, in the next state, the elements of the matrix will be shifted to left by one, with the entries of the last vehicle being updated with actual values, and the information of the next vehicle entering the DOCA that needs to be assigned a resource is appended to the right end of the matrix. Such a state representation contains all the necessary information from the environment in a compact form, whose size is independent of the number of vehicles and resources available in the network.

The goal of the learning agent is set to maximize the reliability of transmissions taking place in DOCA. We use the reliability metric PRR as defined in Section 3.2. Specifically, after each action, the reward collected from the environment is defined to be $+10$ in case $PRR \geq 90\%$ for all transmissions, and $-10 \times (1 - min(PRR))$, otherwise, where minimum PRR of any of the transmissions is used. For E2, in order to avoid any under-utilization of any resources by the scheduler, we have also modified the reward definition as $-10 \times (1 - min(PRR)) - R_0$, where R_0 is the number of resources that are not assigned to any vehicle in that state.

The implemented DNN for both actor and critic consists of 2 convolutional layers followed by 2 fully connected ones. All layers have tanh as the activation function, except the last one using linear function in the case of critic network, and softmax function in the case of actor network, to output the value function and the action probabilities respectively. When processing the state information of E2, which has 2D structure as compared to 1D state information for E1, rows of the 2D state information are separately fed into different convolutional layers as the input. Output of each layer are then merged and fed to the second convolutional layer together, followed by a single fully connected layer.

5.3 Evaluation

5.3.1 Simulation Setup

In our evaluations, the DOCA is assumed to be of length $L_{\text{DOCA}} = 500$ m, having a single lane per direction, each 4 m wide. We consider a constant density of vehicles in the environment, in order to simulate a constant load of V2V communication traffic for a more tractable analysis of the algorithms. To achieve this, similar to the mobility model described in Section 3.1.6, a certain number of vehicles are initially assumed to be within DOCA, traveling at the same constant speeds, and upon their exit, they are assumed to return back to DOCA from the opposite direction after a random time offset exponentially distributed with a mean of 2.5 s, to account for realistic Poisson distribution of inter-vehicle gaps [236]. Vehicles are assumed to generate broadcast V2V messages of size $S_{\text{m}} = 190$ Bytes, each occupying a single TB, with a periodicity of $T_{\text{m}} = 100$ ms.

In the case of E1, we consider three scenarios, designated E1-A, E1-B, and E1-C, which differ with respect to vehicle densities and the number of resources. In scenario E1-A, 10 vehicles reside in DOCA, all traveling at 140 km/h, where a resource pool consisting of 1 subchannel and 10 slots is utilized by the scheduler. In scenario E1-B, 12 vehicles travel at 140 km/h, this time having a resource pool of 2 subchannels, and 10 slots. In the latest scenario, E1-C, there are 24 vehicles traveling at 70 km/h residing in DOCA, utilizing a resource pool with the same size of 2 subchannels, and 10 slots. Our choice of the scenarios is motivated by the goal of representing the following three cases of network condition; E1-A: loaded, without half-duplex (HD) constraint, E1-B: under-loaded, with HD constraint; and E1-C: over-loaded, with HD constraint. The transmit power of the vehicles are set to its allowed maximum value of 23 dBm [282] in E1, without consideration of any path loss, in order to simulate a single collision domain.

In E2, we consider a single scenario where 30 vehicles are traveling at the speed of 50 km/h across the DOCA, and where a resource pool of 2 subchannels by 10 slots is available. This scenario is used to evaluate the potential of our RL solution on reusing resources which will overcome the drawbacks of the overloaded situation. In order to enable

Table 5.1 Simulation parameters (adapted from [34] ©2018 IEEE).

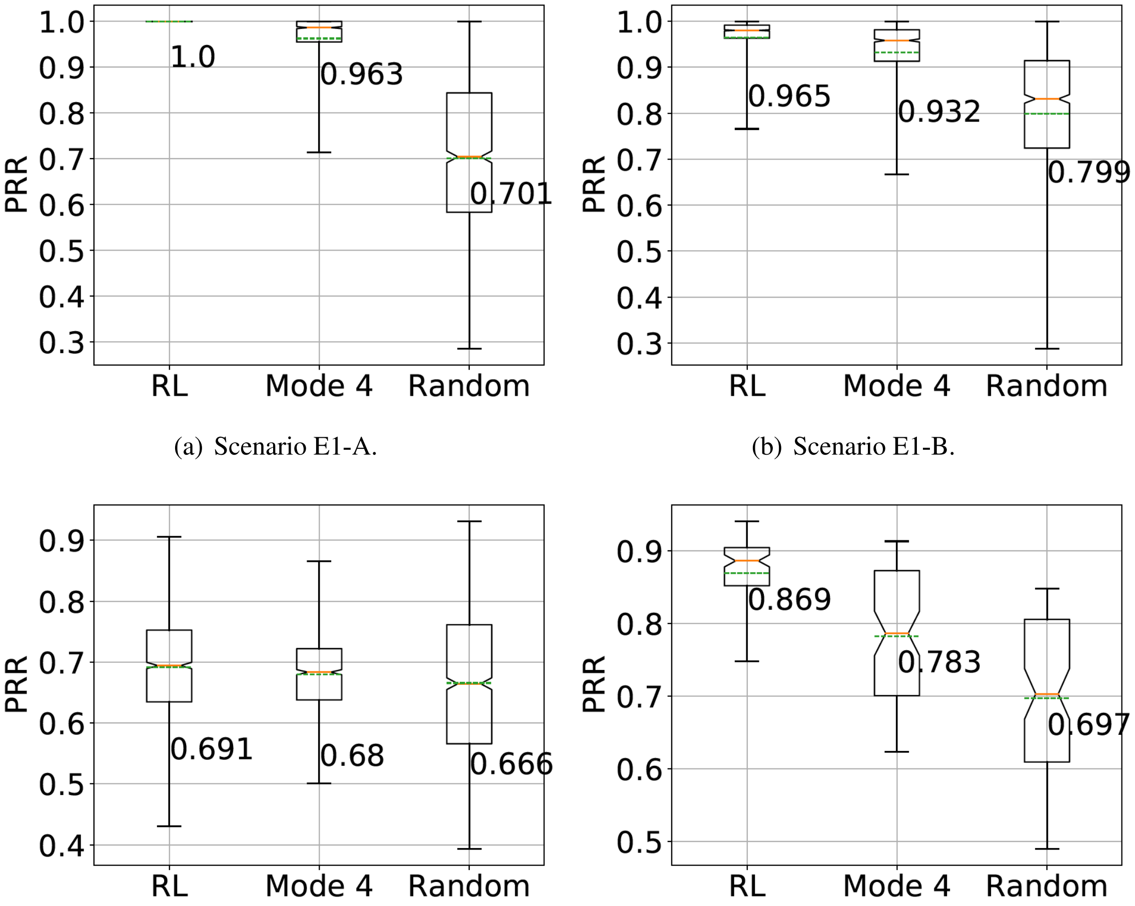
	E1-A	**E1-B**	**E1-C**	**E2**
Number of vehicles	10	12	24	30
Vehicle speed	140 km/h	140 km/h	70 km/h	50 km/h
Resource pool	1 subchannel 10 slots	2 subchannels 10 slots		
DOCA size	500 m of a straight highway, 1 lane per direction, 4 m lane width			
Vehicle spatial distribution	Poisson with 2.5-s inter-vehicle distance on average [236]			
Transmission power	23 dBm (the maximum allowed value)			–5 dBm
V2V message size and period	190 B, 100 ms			
Slot duration	1 ms			
V2V channel model in E2 [236]				
Pathloss model	LOS in WINNER+B1 with antenna height = 1.5 m; pathloss at 3 m is used for distance < 3 m			
Shadowing fading	Log-normal distributed with 3 dB standard deviation, and decorrelation distance of 25 m			

resource reuse within the considered DOCA size of 500 m, transmission powers of the vehicles are reduced to -5 dBm (as opposed to transmitting with the maximum power of 23 dBm in E1). This way, the power received beyond 100 m away from the transmitter is reduced to around noise power level, which in turn enables reusing the same resource at around a distance of 200 m. Moreover, the realistic channel model according to 3GPP evaluation assumptions is adopted, with details provided in Table 5.1.

As described in Section 3.1.6, the RL agents for each scenario are initially trained in an environment simplified in terms of the propagation model, assuming a V2V transmission range of $R_{\mathrm{Tx}} = 120$ m. For training the A3C algorithm, each instance of the environment is generated using a different random seeds of the simulation. For the reward calculation, PRR is measured at $0-100$ m Tx-Rx distance for the transmissions taking place between each action, as described in Section 3.2. The evaluation environments and scenarios are summarized in Table 5.1, together with the corresponding values of the parameters utilized in each of them.

5.3.2 Comparison

In this section, we evaluate and compare the performance of the trained centralized RL scheduler with two baselines: the sensing-based distributed scheduling mode 4 from the 3GPP standard [12] as described in Section 2.3.3, and a centralized scheduler assigning random resources to the vehicles entering DOCA. Simulation of the vehicular environment is carried out using the realistic network simulator ns-3 [259], and the vehicular mobility



(a) Scenario E1-A.

(b) Scenario E1-B.

(c) Scenario E1-C.

(d) Scenario E2.

Fig. 5.4 Mean (green, dashed, denoted), median (red) with 95% confidence interval around (notches), 25^{th} and 75^{th} percentiles (box), and 1^{st} and 99^{th} percentiles (whiskers) of PRR for the proposed centralized RL scheduler, distributed mode 4 algorithm [236], and a centralized scheduler assigning random resources [34] ©2018 IEEE.

simulator SUMO [247] as described in Section 3.3. The key performance indicator (KPI) we are interested in is the PRR as defined in Section 3.2. We report the results in terms of the mean, median (with 95% confidence interval), and percentiles of PRR measured for the range $0-100$ m between the transmitters and the receivers. Fig. 5.4 shows the results for the considered scenarios.

For E1-A, RL scheduler is able to perform at 100% PRR (after eliminating the transient phase that starts from the state of randomly assigned resources), which is achieved by learning to allocate time-orthogonal resources to each vehicle in DOCA. As the number of vehicles inside the DOCA is equal to the number of resources, no collision would occur, and all the vehicles can hear each other all the time. Mode 4 is able to achieve a mean PRR of 96.3%, where the performance degradation comes from the randomness in its resource selection algorithm. After each sensing period, vehicles select the resource to transmit randomly among the best 20% resources according to their sensing results (as described in Section 2.3.3). In our case, each vehicle selects one of the two best resources out of 10 at random, which results in collisions if an occupied resource is selected. As one of the two selected resources will always be occupied for the case of the last (10^{th}) vehicle selecting a resource, collision happens with a probability of 5% ($1/2 \times 1/10$) on average, which is in line with our simulation results. The scheduler assigning random resources acts as a reference for the remaining two algorithms, as it performs the worst with a mean PRR of 70.1%. The optimal performance, however, could be also achieved using a round-robin scheduler assigning time-orthogonal resources to the vehicles entering DOCA. In that sense, scenario E1-A serves as a sanity-check, where RL scheduler is shown to perform optimally.

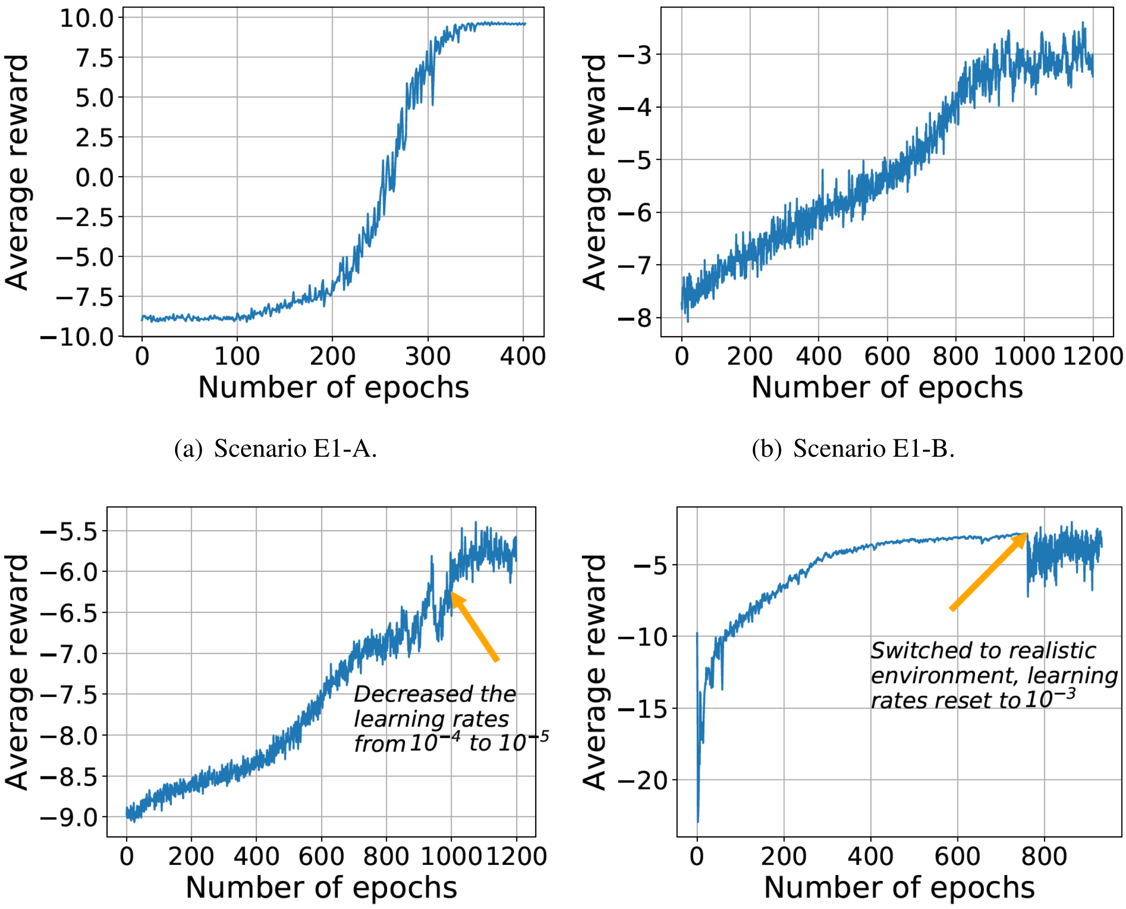
In E1-B, performance of both RL scheduler and mode 4 is degraded, due to introduced HD constraint in the environment. Whenever a vehicle transmits, it does not hear the other transmissions taking place at the same slot on the next subchannel. Nevertheless, RL can achieve a performance of 96.5% average PRR as compared to mode 4 (93.2% average PRR). The strategy that the RL scheduler learns in this scenario is to allocate resources orthogonal both in time and frequency as much as possible. As there are 2 more vehicles than the number of slots, RL scheduler tries to assign them to different subchannels, rather than assigning to the occupied subchannel at each slot, hence most of the time resulting only in half-duplex reception errors among 2 vehicles instead of any collision error affecting the reception of all vehicles. Hence, again, the RL scheduler manages to find the near-optimal solution. On the other hand, random resource allocation performs better than in Scenario A, as the network is in an under-loaded condition with higher probability of assigning non-colliding resources, compared to a loaded one.

Scenario E1-C represents the overloaded network conditions, in addition to the HD constraint. Therefore, collisions are unavoidable in any case since all vehicles are assigned resources (i.e., no admission control), which results in a considerable amount of performance degradation in case of all algorithms. In this scenario, RL scheduler develops a strategy where it tries to maximize the number of non-colliding resources, namely assigning them orthogonally in time and frequency as much as possible, as in E1-B, this time scheduling all the remaining vehicles onto one or two resources where they collide. In the best case, 19 vehicles in DOCA are scheduled to orthogonal resources, and the remaining ones are

"Best" in this context is defined as the lowest energy sensed on the resource.

all being assigned the single resource left, which results in a mean PRR of about 75%. RL scheduler performs slightly better than mode 4, and provides a mean PRR of 69.1%.

Scenario E2 allows for the reuse of the resources, which results in an overall better performance compared to the overloaded case of scenario E1-C. PRRs up to 94% are achievable by RL, even in the case of a higher number of vehicles in DOCA. Compared to E1, RL makes use of additional state information obtained from the environment, as explained in Section 5.2.2. Looking at specific state-action pairs, we observe that most of the time RL (re)uses the same resource in either of the directions while allowing some time gap between each reassignment. Moreover, thanks to the modified reward definition for E2, it yields a very low number of unused resources.



(a) Scenario E1-A.

(b) Scenario E1-B.

(c) Scenario E1-C.

(d) Scenario E2.

Fig. 5.5 Learning curve for each of the environments and scenarios [34] ©2018 IEEE.

Table 5.2 RL model parameters (adapted from [34] ©2018 IEEE).

	E1-A	**E1-B**	**E1-C**	**E2**
State information size	10×1	20×1	20×1	$(N = 30) \times 3$
Number of actions per epoch	20	30	48	120
Number of training epochs	400	1400	1200	930
Learning rates of actor-critic	10^{-4}	10^{-4}	10^{-4}, and 10^{-5} for $\#ep > 1000$	$\frac{10^{-3}}{\lfloor 1+0.01\times\#ep^{1.1}\rfloor}$
Layers of actor-critic DNNs	2 convolutional + 2 fully connected			2 conv. + 1 FC
Number of learning actors	16			

5.3.3 Learning Performance

To analyze the convergence of the RL scheduler, we show the learning curves in terms of the collected average reward by the RL agent with respect to the number of training epochs, in Fig. 5.5, for all four scenarios. Detailed training parameters for each scenario are provided in Table 5.2. Note that we are more interested in the convergence behavior, rather than the actual value of the average reward that the agent has been converged to.

As can be observed from Fig. 5.5, it takes around 350 epochs for the RL agent to converge for scenario E1-A to an average reward of around 9.6. On the other hand, scenarios E1-B, E1-C, and E2 require more epochs for the algorithm to converge to a certain level of average reward, mainly due to larger state-space they contain. Particularly, the algorithm converges to an average reward of around -3.1 after around 900 epochs in E1-B. In E1-C, the algorithm converges to an average reward of around -5.7 at around 1200th epoch. In order to assure convergence for scenario E1-C, we further tuned the learning rates of the actor-critic DNNs, which is shown to have an impact on the learning performance of the A3C algorithm [173]. Specifically, we reduced both from 10^{-4} to 10^{-5} after the 1000^{th} epoch, as we started to observe oscillations on the average reward that are also visible from Fig. 5.5(c). For E2, the agents were able to converge to an average reward of around -2.7 on the simple environment they were trained, in around 760 epochs. The agents were then continued to be trained in the actual environment, in also which their performance was evaluated in Section 5.3.2. Due to longer simulation times, number of training epochs in the realistic environment were limited to around 170 epochs. However, it is expected to have better performance with an extended training. Learning rates are set to an initial value of 10^{-3} and reduced exponentially with the number of training epochs ($\#ep$) to enable better convergence.

5.4 Conclusions

In this chapter, we proposed an RL-based approach for scheduling the resources for V2V communications inside a DOCA in a centralized way. We were motivated by the necessity of an efficient and flexible scheduler to predictively allocate resources, which we concluded in Chapter 4, and recent successful applications of ML techniques in resource allocation problems. The proposed RL scheduler learns how to assign resources to vehicles solely through interaction with the vehicular environment. In particular, we designed the state information of the environment and the reward signal for the RL model, which are processed using deep neural networks, and we extended the state-of-the-art A3C algorithm to train our model.

In specifically-designed DOCA environments varying from simple to complex, and from under-loaded to over-loaded, including half-duplex and realistic channel conditions, we investigated the performance of our solution through simulations. In the considered environments, the proposed RL-based centralized scheduler learned to develop strategies that allowed it to: i) assign fully orthogonal resources in scenario E1-A; ii) avoid HD constraint to the extent possible in E1-B; and iii) to group excess transmissions in case of network overload to a small set of resources in E1-C, thus allowing remaining transmissions to have no collisions. Furthermore, in a more realistic environment, E2, it achieved the reuse of resources by taking the direction and arrival time of the vehicles into account, which lead to success in dealing with an overloaded scenario. Moreover, insights from evaluating a simple environment, E1, helped us to better design the RL agent for a realistic environment, E2. In comparison to existing resource allocation schemes, namely the distributed scheduling algorithm mode 4 from the 3GPP standard and a centralized scheduler allocating resources randomly, the proposed RL-based scheduler achieved as good as or better performance in terms of reliability of V2V transmissions, often outperforming them in the considered settings. Furthermore, the learning process takes a reasonable time, within a few hundred to thousand epochs, thus making the RL-based approach a promising solution for scheduling V2V communications under intermittent network coverage.

Chapter 6

VRLS: Vehicular Reinforcement Learning Scheduler

6.1 Motivation and Contribution

In Chapters 4 and 5, we scoped the problem of scheduling V2V communications in expected delimited out-of-coverage (OOC) areas, and explored the ability of a centralized reinforcement learning (RL) scheduler to "pre-schedule" the V2V transmissions for OOC, respectively. We showed that there lies a strong promise in using RL to efficiently schedule periodic V2V transmissions for OOC areas on highway that experience different vehicular and data traffic. Our encouraging results in Chapter 5 motivate us for further work, in particular, to consider more complex vehicular environments outside the network coverage. On the other hand, RL has its own domain-specific challenges that require careful consideration, especially in pursuance of a practical solution.

We observe that, in Chapter 5, each time we consider a different vehicular environment, we were required to redesign (hence also retrain) our RL model to reach a desirable performance. Specifically, we modified the state representation, reward definition, as well as the structure of the underlying DNN when the environment changed considerably (e.g., in terms of vehicle density, speeds, or the structure of the resource pool), in order to guarantee convergence to a "good" policy through learning. Such an approach becomes impractical for arbitrarily different new environments that V2V communications need to support, and the policy learned in one environment cannot be used as a starting point in another environment. In real-world problems, it is likely that the conditions in a given environment, such as road traffic mobility or data traffic load in a V2V communication network, would change over time.

Parts of Chapter 6 including the results have been published in [35] ©2019 IEEE and [39] ©2022 IEEE.

It would be impractical to redesign, retrain, and reevaluate a new RL solution every time the environment changes, even if this change is substantial. Therefore, a single RL-based solution should be applicable to varying conditions in the environment.

In addition, considering that the RL model is trained “off-line”, i.e., before its deployment, it is also desirable to learn a policy that is applicable to different environments of interest without further training. This would eliminate the need of training a new agent from scratch every time an unseen (yet similar) condition arises in the deployed environment. Furthermore, it would offer the possibility to train the agent in a simpler, simulated environment, saving from the burdens of real-world training. In particular, training and deploying RL solutions in the real world is costly due to following reasons:

- Training an RL agent in a real-world setting is considerably slower than training it in a simulated environment because of the limited availability of data samples.

- Collecting data from an actual vehicular network is expensive, or might not be even possible considering the additional signaling and processing overhead it incurs.

- Any undesirable outcomes of an RL agent still under training might threaten the safety-critical V2V use cases.

Existing literature indicates that the above challenges of RL is not specific to our application in vehicular networks, but rather a general problem of deep learning encountered in different fields. Typically, RL solutions in the literature are designed, trained, and evaluated in the same environment that has a specific distribution of (or even fixed) parameters. However, particular design choices tailored for a specific setting may not work as well or even be applicable when the parameters of the environment change significantly. Specifically, recent studies report the challenge of deep RL, where standard algorithms and architectures are shown to perform poorly in case changes (e.g., noise) applied to the environment [283]. For example, authors of [284] show that the famous deep RL algorithm [171], which is trained to play Atari games and shown to outperform humans, fails completely when simple modifications are applied to the environment (e.g., adding pixels to the screen). Various approaches are proposed by these recent works, such as changing the state representation, applying a different DNN architecture, or training the agent from scratch, in order to achieve applicability and the desired performance of the RL agent in different environments, as fine-tuning is not always effective (cf. [283], [284], and their references).

In this chapter, we propose Vehicular Reinforcement Learning Scheduler (VRLS), a unified RL approach to overcome the above-mentioned challenges for scheduling V2V communications. To achieve this, our VRLS design most importantly focuses on having a *unified* state representation of the vehicular environment, along with the other RL components whose structures remain the same irrespective of the setting they are applied to. Further, the state representation contains relevant information for the resource allocation problem in a condensed manner. In particular, the input to the RL model accommodates variables that convey the resource utilization status within the road section outside the network coverage, which can represent any number of vehicles, size of the road, resource pool configuration, etc. This allows the applicability of the proposed RL model to various environments of interest, as well as facilitates training over simplified ones. The proposed state representation captures the information about the traffic load on each resource, potential interference to the vehicle entering the area, and the vehicle density, per the direction of the traffic, while accounting for the half-duplex constraint among the resources. Similarly, the reward provided to VRLS is unified in a way to reflect our overall goal of maximizing the reliability of V2V transmissions irrespective of the number of resources or vehicles.

Our evaluations in this chapter yield the following contributions:

- VRLS improves the performance of scheduling V2V communications, in terms of reduced packet error rates achievable in DOCA, compared to state-of-the-art algorithms. We evaluate VRLS in several sanity-check scenarios, and show that it can learn near optimal policies in all these cases.

- We train VRLS in simplified and simulated vehicular environments, and show that it can be deployed with limited or no retraining, in realistic, complex environments, varying in terms of mobility, wireless channel characteristics, OOC area size, network load, and traffic.

- We evaluate the performance of VRLS in terms of reliability, resource utilization efficiency, user fairness, latency, and packet inter-reception time, as well as the impact of network quality of service on the V2V applications, in comparison to the state-of-the-art distributed scheduler mode 4. In terms of reliability, VRLS reduces by half the packet loss of mode 4 in highly loaded conditions, and performs close to the theoretical maximum in low-load scenarios. VRLS requires much less number of resources to achieve reliability targets of the V2V applications, as compared to mode 4, especially at higher reliability targets that needs to be satisfied for relatively larger transmitter-receiver distances. Further, VRLS does not compromise on fairness across the vehicular users, while achieving similar latency and higher mutual awareness as compared to mode 4.

- Considering that the network might need to operate differently configured resource pools in terms of the number of resources in time and frequency, e.g., to support different V2V services, we show that VRLS can be trained across multiple predetermined resource configurations at once to support any of them by learning a single policy.

In the rest of this chapter, Section 6.2 describes VRLS, Section 6.3 presents the evaluation results, and Section 6.4 delivers our conclusions.

6.2 VRLS: Vehicular Reinforcement Learning Scheduler

As in Chapter 5, we formulate the centralized resource pre-allocation problem for the DOCA with a single-agent RL problem, where VRLS acts as the agent on the vehicular network environment. Based on the observed *state* s_t of the environment at each discrete instant t in which a new vehicle arrives at the DOCA, VRLS takes an *action* a_t, which is to assign a single time-frequency resource to that vehicle. The actions of VRLS are based on its trained *policy* π, which we model as a *deep neural network* (DNN). The agent is trained with a *reward* signal r_{t+1} provided upon each action, indicating how "good" the action was. In turn, the training goal of the agent is to maximize the total reward it receives in the long run. To train VRLS, we use the A3C algorithm we extended in Algorithm 3 we provided in Chapter 5. We utilize N_{actor} actor-learners in parallel. Each actor-learner updates the shared global parameters of the DNNs that represent the policy and the value function, after collecting epochs of experience, where each epoch consists of a sequence of state-action-reward tuples of length L_{epoch}.

6.2.1 State Information

We devise the state s_t to provide the agent with information on *how* the resources are utilized at each instant t a vehicle is entering the DOCA. Formally, s_t is a matrix (shown in Fig. 6.1) with each row representing a resource (TB) in the resource pool, and the columns providing the following information:

- C: number of vehicles each resource is assigned, normalized to the maximum number of vehicles that the DOCA can accommodate (derived by $J \times L_{\text{DOCA}}/L_{\text{veh}}$, considering the case where all lanes are fully occupied). C represents whether the resources are free and, if not, how much loaded. The vehicle density in the DOCA could also be obtained from C by accounting for the sum of the allocated resources in proportion to the calculated maximum number of vehicles.
- Δx: distance from the entrance point of the DOCA to the latest vehicle the resource was assigned to, normalized to L_{DOCA}. The distance is estimated by multiplying the

amount of time passed since the vehicle went out of coverage by the average speed v_{avg} of the vehicles in the DOCA, as their speed might vary over time. Δx represents how far the potential interferers are, hence facilitating spatial reuse of each resource.

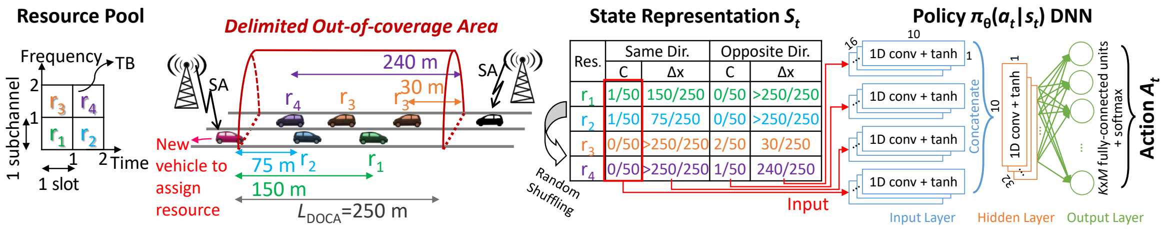
- The *order* of the columns represents the direction of the vehicle entering the DOCA. The first pair of columns provides C and Δx for the vehicles traveling in the same direction as the vehicle entering the DOCA, while the second pair provides the information from the opposite direction of the DOCA.

Algorithm 4 details how s_t is calculated. The following variables are input for each vehicle i inside the DOCA: time of its entry t_i, assigned resource r_i, traveling direction (i.e., east or west) d_i; as well as the average speed of the vehicle traffic v_{avg} and the current time t_{now}. The algorithm first updates the distance Δx_i of vehicles by multiplying the time passed since their entry, with the average speed. The vehicles that exited the DOCA are excluded from the list. Then, for each resource r in the resource pool and for each road direction d, the algorithm finds the vehicles using that resource and traveling in that direction. The number of found vehicles is normalized and entered in s_t as the value of C for the respective resource r and the direction d. Next, among the found vehicles, the algorithm finds the closest one to the entry point of the DOCA entry at the given direction, i.e., with $\min(\Delta x_i)$, and updates the variable Δx in s_t with the normalized value of $\min(\Delta x_i)$ for the respective d and r. Finally, the pairs of the columns in s_t are ordered with respect to the entering vehicle's direction.

Algorithm 4 Calculation of the state representation s_t input to VRLS (adapted from [39] ©2022 IEEE)

```
Require: $t_i, r_i, d_i \ \forall i, v_{\text{avg}}, t_{\text{now}}$
Update vehicle distances: $\Delta x_i \leftarrow v_{\text{avg}}(t_{\text{now}} - t_i) \ \forall i$
Remove vehicle if it left the DOCA ($\Delta x_i > L_{\text{DOCA}}$)
for each resource in the pool $r = 1, 2, ..., K \times M$ do
    for each road direction $d = \{\text{east}, \text{west}\}$ do
        Find vehicles using the same resource in the same direction (check if $d_i == d$ && $r_i == r$)
        Update $C$ of the respective $d$ and $r$ with the number of found vehicles
        Sort distances of found vehicles to find $\min(\Delta x_i)$
        Update $\Delta x$ of the respective $d$ and $r$ with $\min(\Delta x_i)$
    end for
end for
Order the columns of $s_t$ w.r.t. the direction of the entering vehicle
return $s_t$
```

s_t is applicable to any number of resources and vehicles, and any DOCA size, thanks to the normalized state variables. An example s_t is illustrated in Fig. 6.1 for a simple scenario



Res.	Same Dir.		Opposite Dir.	
	C	Δx	C	Δx
r_1	1/50	150/250	0/50	>250/250
r_2	1/50	75/250	0/50	>250/250
r_3	0/50	>250/250	2/50	30/250
r_4	0/50	>250/250	1/50	240/250

Fig. 6.1 State representation of a simple exemplary scenario in the DOCA provided to the DNN of VRLS [39] ©2022 IEEE.

with 4 resources and a DOCA of $L_{\text{DOCA}} = 250$ m, where a maximum of 50 vehicles of $L_{\text{veh}} = 5$ m can fit per lane.

6.2.2 Action Definition

The agent takes an action a_t, at each instant t a vehicle is about to enter the DOCA. In case multiple vehicles enter at the same instant, the corresponding actions are taken in random order. Action denotes assigning a single time-frequency resource, i.e., a TB, which the vehicle uses for its V2V transmissions through the DOCA. Accordingly, the action-space is a vector of $K \times M$ TBs in the resource pool configured in the network. VRLS gives the decision on which TB to be assigned at time t by its policy π. The policy is a mapping $\pi(a_t|s_t) \rightarrow [0,1]_{K\times M}$ from the state s_t of the environment at t, to a probability distribution over the set of possible actions (the TBs in the resource pool). The TB to be assigned is selected at random according to this distribution.

6.2.3 Deep Neural Network Architecture

The large space of possible combinations of vehicles and resources makes tabular RL methods infeasible for this problem [164] (cf. Section 2.5.2). This leads us to apply approximate solution methods by utilizing a DNN to represent the policy. DNN consists of a set of adjustable parameters θ, i.e., $\pi(a_t|s_t,\theta)$ that maps a given state to action probabilities.

We utilize a convolutional neural network (CNN) to model π_θ. At the input layer, we utilize 4 sets of convolutional filters, each processing a different column of s_t, as illustrated in Fig. 6.1. Each set contains 16 1D convolutional filters of length 10 and applies a tanh nonlinearity. The output of these filters is then concatenated and input to the hidden layer of the CNN, which is another convolutional layer with 32 1D filters of length 10. The output layer of the CNN is a fully-connected layer with the number of units equal to the number of actions, i.e., $K \times M$ TBs available in the configured resource pool. The softmax activation function is applied at the output to produce a probability distribution over the actions, from which the TB to be assigned is selected at random.

6.2.4 Data Augmentation

The output of convolutional layers is variant to the *order* of the input data they process, due to the convolution operation. Although this is useful for their most common applications, such as processing images or audio that present naturally ordered data (e.g., ordered pixels in space), this feature poses a limitation in our case. The policy of VRLS should not depend on the order of resources presented in s_t, but rather on the information provided about them. That said, the HD constraint depends on the order of resources in the resource pool, as the HD error is caused by using the resources in the same slot (cf. Section 3.1.4). For example, the two pool configurations $C^{4\times5}$ and $C^{2\times10}$ have different HD constraints, although both have the same number of resources, and there is no information in the state representation to differentiate between them.

To address this challenge, we resort to data augmentation methods. Data augmentation is commonly utilized in deep learning, e.g., for image classification tasks, where the agent is made to *learn* becoming invariant to the modifications of the data, e.g., image rotation, clipping, etc., by providing the agent with such modified inputs during the training, as applied in [285] and [286]. In our case, we apply data augmentation by randomly shuffling the order of resources first in time and then in frequency. The rows of s_t and the resource selection probabilities at the output layer of the DNN follow this order. To illustrate our method, consider the example in Fig. 6.1. A raster-scan ordering of the resources in the resource pool is $[r_1, r_2, r_3, r_4]$. We first group the resources sharing each slot (corresponding to "columns" of the resource pool), and randomize the order of these groups. This yields a raster-scan ordering of, e.g., $[r_2, r_1, r_4, r_3]$. Then, we group the resources sharing each subchannel, i.e., the "rows" of the pool, and randomly shuffle the order of the "rows". This way, the convolutional network becomes invariant to the order of resources in time or frequency, while being able to infer the HD constraints among the resources.

In Appendix A, we provide several other design options for the DNN architecture and data augmentation we have considered for VRLS, along with the one proposed here. We compared their performance in terms of learning performance and reliability of V2V messages. Our results showed that the proposed design here achieves the best performance in both terms.

6.2.5 Reward Definition

We impart our main goal of maximizing the reliability of transmissions taking place in DOCA to the reward signal r_{t+1} provided to the agent upon its each action a_t. Specifically, we define the reward as a linear function of the reliability metric PRR: $r_{t+1} = -10 \times (1 - \text{PRR})$. PRR is computed at a certain range of interest for all transmissions within the DOCA *since the last action*, i.e., in between each vehicle arrival to the DOCA. The range at which the PRR is measured for the reward could be determined by several factors, such as the distance at which a target PRR value needs to be satisfied; additionally, it can also be limited by the transmission power of the vehicles. In case no transmissions take place between consequent actions, e.g., when two vehicles enter the DOCA almost at the same time, we provide the reward of the previous action to the agent.

6.3 Evaluation

In this section, we first compare the performance of VRLS with the state of the art, including the RL scheduler we designed in Chapter 5. We then demonstrate the ability of VRLS to handle resource conflicts including the half-duplex (HD) constraint under various resource pool configurations and settings that enable a tractable analysis, where the optimal schedule is computable. Later, we evaluate the performance VRLS achieves in case of more complex, realistic scenarios, under varying mobility, density, wireless channel, and message traffic inside the DOCA. Following, we show how VRLS can handle multiple differently configured resource pools in parallel. In the last two subsections, we analyze the learning performance of VRLS and elaborate on its practical aspects including a complexity analysis, respectively, for implementing VRLS in the real world.

6.3.1 Training Environment Model and Methodology

The training environment (denoted as "E0") has basic vehicular mobility and wireless channel characteristics, as described in Section 3.1.6, which enables an efficient training thanks to reduced simulation time. The communications in the training environment is abstracted by the *protocol model* [250], using a transmission range of $R_{\text{Tx}} = 120$ m. The mobility is simple with 30 vehicles having the same constant speed of 50 km/h, initially placed uniformly at random inside a DOCA of length $L_{\text{DOCA}} = 500$ m with $J = 1$ lane/direction of 4 m width. Upon exiting the DOCA, the vehicles are returned back from the opposite direction after a time offset $\sim \text{Exp}(0.4)$, leading to an average inter-vehicle gap of 2.5 s [236]. The V2V resource pool in the network is assumed to be configured with $C^{2\times 10}$, i.e., 2 subchannels by

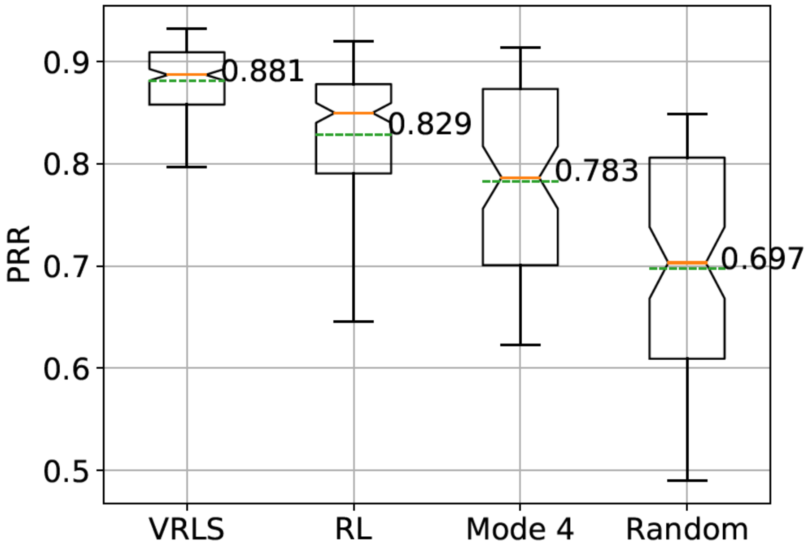


Fig. 6.2 Comparison of VRLS to the state of the art (our solution proposed in Chapter 5 denoted with “RL”) in a multi-collision-domain (MCD) DOCA. Mean (green, dashed, denoted), median (red) with 95% confidence interval around (notches), 25^{th} and 75^{th} percentiles (box), and 1^{st} and 99^{th} percentiles (whiskers) of PRR (adapted from [35] ©2019 IEEE).

10 slots to generate loaded conditions with a V2V message traffic that has a fixed periodicity of $T_{\mathrm{m}} = 100$ ms.

The training is conducted using $N_{\mathrm{actor}} = 16$ actors in parallel, each interacting with a different instance of the environment in epochs of length $L_{\mathrm{epoch}} = 60$. For the computation of the reward during the training, PRR is measured at $0 - 100$ m Tx-Rx distance for the transmissions taking place in between each action, as described in Section 3.2.

6.3.2 Comparison of VRLS and State-of-the-art Algorithms

We first compare the performance of VRLS with our solution proposed in Chapter 5, as well as the distributed scheduling algorithm mode 4 from the 3GPP standard [12] (as described in Section 2.3.3), and the random resource allocation performed by a centralized scheduler.

We evaluate the performance of the algorithms in an overloaded network scenario with multiple collision domains inside DOCA, denoted as “MCD”. To achieve this condition, 30 vehicles are assumed to be traveling in DOCA (of 500-m length with single lane per direction), with an available resource pool configured with 2 subchannels by 10 slots, i.e., $C^{2\times10}$, where their transmission ranges are limited to around 120 m by adjusting the transmission powers. As in the training environment, vehicular density in DOCA is constant where vehicles return

Table 6.1 Simulation parameters (adapted from [35] ©2019 IEEE).

	MCD	**SCD-i**	**SCD-ii**	**SCD-iii**
Maximum number of vehicles	30	10	4	5
Resource pool configuration	$C^{2\times10}$	$C^{2\times10}$	$C^{10\times2}$	$C^{4\times5}$
DOCA	$L_{\text{DOCA}} = 500$ m of a straight highway, $J = 1$ lane per direction, 4 m lane width			
Vehicle speed	50 km/h			
Vehicle distribution	Poisson with mean of 2.5-s distance [236]			
V2V transmission power P_{Tx}	–5 dBm	23 dBm (the maximum value)		
V2V message size and periodicity	$S_{\text{m}} = 190$ Bytes, $T_{\text{m}} = 100$ ms			
Mode 4 P_{keep}	0			
Number of actions per epoch	60			
Actor-critic learning rates	$10^{-3}/(1+0.01\times\#ep^{1.1})$			
V2V channel model [236]				
Pathloss model	LOS in WINNER+B1 with antenna height = 1.5 m; pathloss at 3 m is used for distance $<$ 3 m			
Shadowing fading	Log-normal distributed with 3 dB standard deviation, and decorrelation distance of 25 m			

back to DOCA from the opposite direction once they leave it, and assumed to travel at constant speeds. Different than the training environment, path loss and fading effects on the wireless channel are introduced into the evaluation environment with parameters in Table 6.1. We report the performance of the RL agents after a limited re-training (for 200 epochs) on the new environment. Further details of the evaluation assumptions are provided in Table 6.1.

We present the results of the algorithms in Fig. 6.2 in terms of the PRR measured at $0-100$ m range between the transmitters and receivers, as described in Section 3.2. Under the considered settings, VRLS outperforms the other schedulers by reaching up to 93% PRR. Specifically, we observe a significant improvement on low percentiles. The performance gain with respect to our previous RL scheduler is mainly achieved by the difference in our design of the state representation described in Section 6.2.1.

In terms of the developed policy, VRLS learns to divide the resource pool dynamically into two directions of the highway, proportional to the density of each direction, while performing resource reuse per direction. This way, resources are efficiently utilized while aiming to minimize the collisions, with the trade-off controlled by the received award. On the other hand, HD errors occur due to agent's allocations, which in this scenario is unavoidable given the overloaded conditions of the network. In majority of such cases, subchannels sharing the same slot are assigned to vehicles moving in *opposite* directions. Such vehicles would not be able to listen to each other when passing each other for a short duration of time. However, this type of allocation degrades the PRR to a lesser extent compared to the

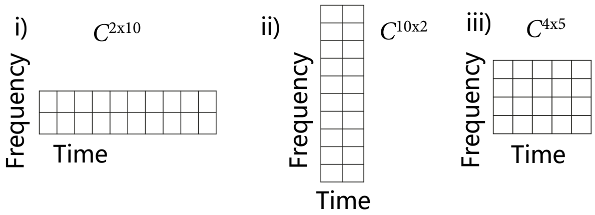


Fig. 6.3 Different configurations of resource pools considered for evaluations in Section 6.3.3 (adapted from [35] ©2019 IEEE).

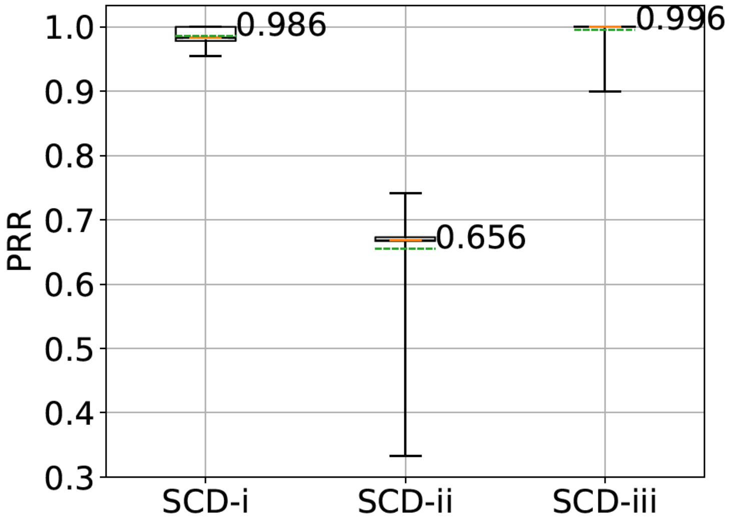


Fig. 6.4 Performance of VRLS on a single-collision-domain (SCD) DOCA, with added complexity to mobility, in scenarios SCD-i, SCD-ii, and SCD-iii. Mean (green, dashed, denoted), median (red) with 95% confidence interval around (notches), 25^{th} and 75^{th} percentiles (box), and 1^{st} and 99^{th} percentiles (whiskers) of PRR [35] ©2019 IEEE.

impact of alternative policies, e.g., HD errors or collisions that would otherwise occur more persistently in the same direction.

6.3.3 Learning the Half-duplex Constraint

Our evaluation in the previous subsection demonstrated VRLS capability to reuse TBs, and prevent collisions in an overloaded MCD scenario. In this subsection, we evaluate the performance of VRLS in scenarios that specifically require its capability of learning and solving the HD constraint, given different configurations of the resource pool, as shown in Fig. 6.3. In particular, we consider a single-collision-domain (SCD) DOCA, where all vehicles inside are able to sense each other's transmissions, and any resource reuse leads to collision. Fig. 6.3 shows the three resource pool configurations. The first two scenarios represent the two extremes of a resource pool configuration: SCD-i) $C^{2\times10}$: 2 subchannels by 10 slots, and SCD-ii) $C^{10\times2}$: 10 subchannels by 2 slots. For SCD-i and SCD-ii, we simulate a maximum number of 10 and 4 vehicles, respectively. The third scenario (SCD-iii) lies in between: 4 subchannels by 5 slots (i.e., $C^{4\times4}$), and we simulate 5 vehicles. These settings are chosen such that any HD errors would decrease the PRR considerably, and the optimal resource allocation is possible only if the HD relation among the resources is learned by the scheduler. Differing from the previous evaluations, we also introduce an added complexity to the vehicular mobility, where after leaving, vehicles return to DOCA after a time offset distributed exponentially at random with a 2.5 s mean, which introduces time-varying vehicular density inside DOCA.

The performance of the RL agent for each scenario is shown in Fig. 6.4 in terms of PRR.We observe that VRLS can easily adapt to each of the settings, and performs near optimal in all scenarios (i.e., close to the analytical maximum), after a training of around 500 epochs. In SCD-i and SCD-iii, 100% PRR is achievable analytically, if the TBs assigned to vehicles are all orthogonal in time (i.e., chosen from different slots). In SCD-ii, in case all four vehicles are inside the DOCA, then in the best case, two vehicles are assigned different subchannels in one slot and two in another, yielding a PRR of 66.7% (limited by the HD constraint). Higher PRRs are achievable in the case of fewer number of vehicles traveling through DOCA.

In SCD-i, a single HD error due to an assignment of two resources non-orthogonal in time would analytically lead to 97.7% PRR, which can be observed in around 25% of the cases. In SCD-ii, a single HD conflict would result in a PRR of 50%, observed in less than 25% of the cases. In case of SCD-iii, the agent is able to achieve a similar performance, having a single HD conflict in 1% of the cases, which results in a PRR of 90%. Moreover, occasionally in SCD-ii, there are fewer than four vehicles traveling in DOCA, where a single HD error between two vehicles would yield a PRR of 0%. The trained RL agent is successfully able to yield non-zero PRRs more than 99% of the time.

On the other hand, compared to HD errors, any collision error (due to assignment of the same TB to more than a single vehicle) would reduce the PRR to a greater extent. As an example, in SCD-i, assigning the same TB to a single pair of vehicles in DOCA would result in an analytically derived PRR of 80%. Such cases were only observed in less than 1% of the time, which shows the success of the RL agent on avoiding the collisions. Overall, the results show the ability of VRLS to learn and deal with the HD constraint, in addition to avoiding the collisions, achieved in three different resource pool configurations.

Table 6.2 Simulation parameters (adapted from [39] ©2022 IEEE).

	Realistic Evaluation Environments			
	E1-L	**E1-HL**	**E2-L**	**E2-HL**
Mobility model	Straight highway section with 4-m lane-width; vehicle length $L_{\text{veh}} = 5$ m			
– Highway scenario	$J = 1$ lane/direction, no overtaking $L_{\text{DOCA}} = \{500, 1000\}$ m		$J = 2$ lanes/direction, with overtaking $L_{\text{DOCA}} = \{500, 1000\}$ m	
– Vehicle speeds	$\sim \mathcal{N}(120, 12)$ km/h	$\sim \mathcal{N}(50, 5)$ km/h	$\sim \mathcal{N}(120, 36)$ km/h	$\sim \mathcal{N}(50, 15)$ km/h
– Dynamics	Poisson arrival per direction with $\sim$ Exp(0.4) [236] Realistic SUMO mobility [247]			
Network model	Complete LTE V2X protocol stack in ns-3 [259] [266] [265] Bandwidth = 10 MHz (50 RBs) with 32 RBs active; Carrier frequency = 5.9 GHz 1 subchannel = 16 RBs, 1 slot = 1 ms, MCS index = 9			
V2V channel model	3GPP Channel Model [236] with Path loss: LOS model in WINNER+B1 with antenna height = 1.5 m; path loss at 3 m is used for distances < 3 m Shadowing fading: log-normal distr. with 3 dB std. dev. and 25 m decorr. distance $P_{\text{Tx}} = \{–5, 23\}$ dBm; Thermal noise level = –174 dBm/Hz 1 Tx and 2 Rx omni-directional antennae with 3 dBi gain and 9 dB Rx noise figure			
Message traffic model	$S_{\text{m}} = 190$ Bytes; $T_{\text{m}} = 100$ ms for periodic [236], and $X_{evt} = 1/$ s for aperiodic traffic.			
Resource pool	$C^{2\times10}$ (2 subchannels by 10 slots) and $C^{2\times50}$ (2 subchannels by 50 slots), periodically repeating with 100 ms			
Mode 4 Configuration Parameters				
$T_1 = 4$ ms [236], $T_2 = \{14, 54\}$ ms, $C_{\text{resel}} \sim \text{Unif}[5, 15]$ [100], $P_{\text{keep}} = 0$, $Thr_{\text{sense}} = –120$ dBm				
VRLS Training Parameters				
$N_{\text{actor}} = 16$; $L_{\text{epoch}} = 60$; $\alpha = 10^{-3}/(1 + 0.01 \times \#ep^{1.1})$				

6.3.4 Realistic Evaluation Environment Models and Methodology

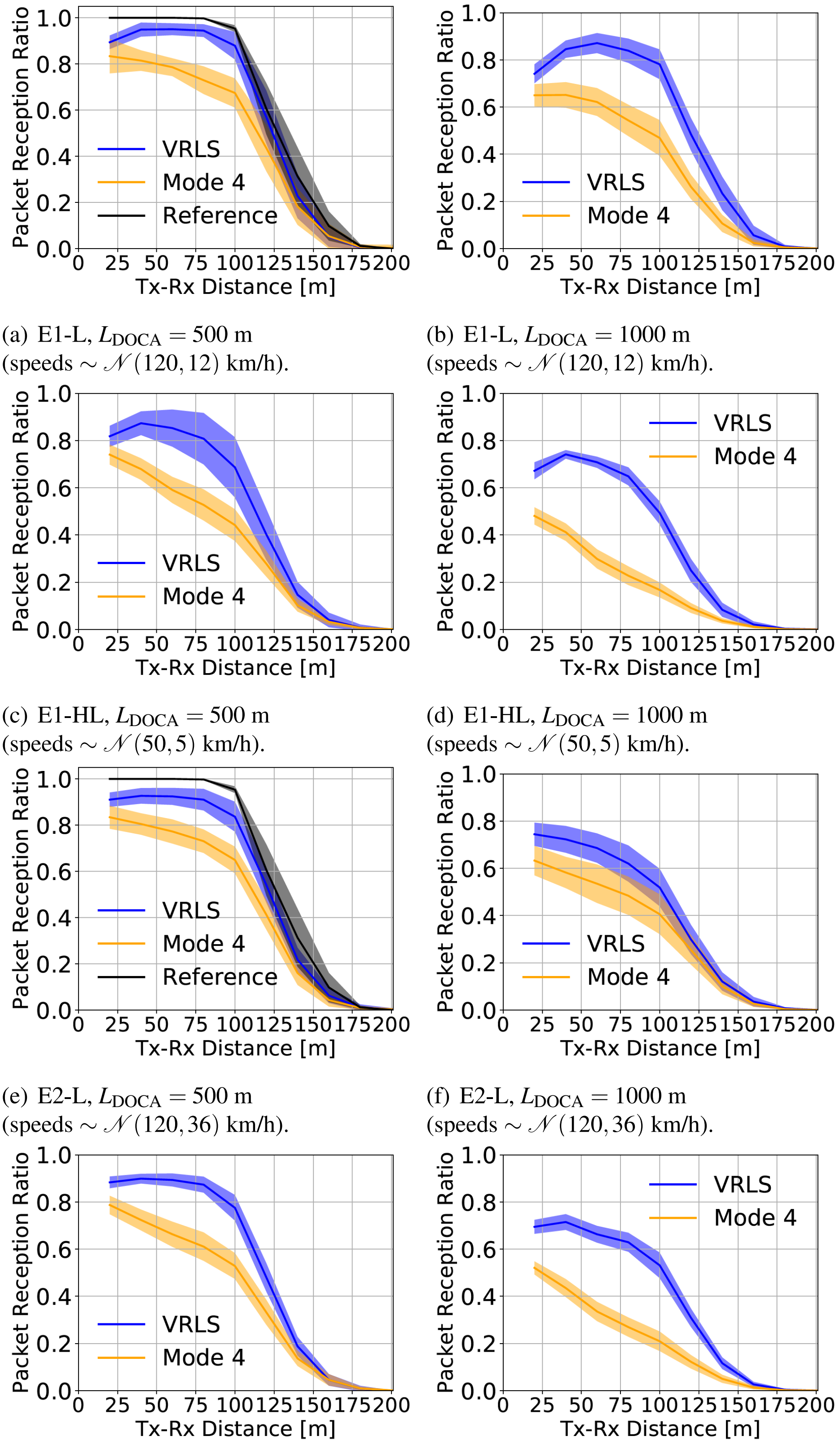
We now evaluate the performance of VRLS trained on the simple environment E0, over realistic environments accommodating various mobility, density, wireless channel conditions, and message traffic in the DOCA. Table 6.2 provides an overview of the considered environments and their parameters.

We consider two realistic evaluation environments denoted as “E1” and “E2”. E1 has a single lane per direction, which obliges vehicles to drive in an ordered manner, thus representing a use case similar to platooning. Whereas, E2 has two lanes per direction, which yields more dynamic mobility due to the second lane allowing overtaking. We consider two DOCA lengths of $L_{\text{DOCA}} = 500$ m and $L_{\text{DOCA}} = 1000$ m for both environments. The vehicle arrivals to the DOCA follow a Poisson distribution with rate 0.4/s (mean of 2.5 s inter-arrival time) per direction as per the 3GPP evaluation assumptions [236]. The vehicles follow a stochastic driving behavior by randomly varying their speeds based on the utilized car-following and lane-changing models [245], [246], which depend on, e.g., average speed, road length, etc., hence making the mobility even more realistic in the evaluation environments. We vary the mean and variance of the vehicle speeds in both environments to create different loads of the vehicular traffic over time and space. Specifically, we consider two scenarios in terms of vehicle density, denoted as loaded ("L") and highly loaded ("HL"), both in E1 and E2, where the mean speed of the vehicles is set to 120 km/h and 50 km/h, respectively (i.e., the slower, the denser). Further, the speeds among the vehicles are normally distributed, where we set the variance to 10% and 30% of the mean values in E1 and E2, respectively. The higher variance of speeds in E2 increases the occurrence of vehicle take-overs across the two lanes.

Unless otherwise stated, the vehicles generate a periodic V2V traffic with $T_{\text{m}} = 100$ ms and $S_{\text{m}} = 190$ B (as common to CAMs [236]). We set the MCS index as 9 and the number of RBs per subchannel as 16 to fit the transmission of a single message of 190 B into a single subchannel. In order to simulate loaded (and highly loaded) channel conditions in our evaluations, we assume that the resource pool consists of 2 subchannels in the frequency domain (within an overall V2V bandwidth of 10 MHz) and 10 slots in the time domain (hence denoted by $C^{2\times 10}$) unless otherwise stated, considering the number of vehicles and their V2V message generation rate. We accordingly set the length of the resource selection window of the mode 4 algorithm to 10 ms with $T_1 = 4$ [236] and $T_2 = 14$ ms. T_1 is to allow a processing time for the vehicles before they transmit their V2V messages, and T_2 sets a limit on the maximum latency of the transmissions. P_{keep} is set to 0, which leads to a dynamic re-selection of resources as much as possible, and has been shown to improve the reliability by avoiding persistent collisions especially under (highly) loaded channel conditions as in [287]. To enable multiple collision domains within smaller DOCA lengths (500 m), we set the V2V transmit power as -5 dBm, which yields a maximum communication range of around 200 m. This allows us to simulate environments with fewer vehicles, thus taking shorter simulation times. Nevertheless, we evaluate the performance of the algorithms also with the transmission power set to its allowed maximum value of 23 dBm [282] in Section 6.3.8. The further parameters related to the environment models, training of VRLS, and configuration of mode 4 are as listed in Table 6.2.

6.3.5 Reliability Performance

In Fig. 6.5, we compare the reliability of VRLS and mode 4 in E1 and E2 with loaded (L) and highly-loaded (HL) traffic with two DOCA sizes of $L_{\text{DOCA}} = 500$ m and 1000 m, using different subfigures. The plots provide the mean (solid curve) and the standard deviation (shaded region) of the average PRRs calculated in 10 s intervals, for a simulation duration of



(a) E1-L, $L_{\text{DOCA}} = 500$ m (speeds $\sim \mathcal{N}(120,12)$ km/h).

(b) E1-L, $L_{\text{DOCA}} = 1000$ m (speeds $\sim \mathcal{N}(120,12)$ km/h).

(c) E1-HL, $L_{\text{DOCA}} = 500$ m (speeds $\sim \mathcal{N}(50,5)$ km/h).

(d) E1-HL, $L_{\text{DOCA}} = 1000$ m (speeds $\sim \mathcal{N}(50,5)$ km/h).

(e) E2-L, $L_{\text{DOCA}} = 500$ m (speeds $\sim \mathcal{N}(120,36)$ km/h).

(f) E2-L, $L_{\text{DOCA}} = 1000$ m (speeds $\sim \mathcal{N}(120,36)$ km/h).

(g) E2-HL, $L_{\text{DOCA}} = 500$ m (speeds $\sim \mathcal{N}(50,15)$ km/h).

(h) E2-HL, $L_{\text{DOCA}} = 1000$ m (speeds $\sim \mathcal{N}(50,15)$ km/h).

Fig. 6.5 Performance of VRLS and distributed scheduling algorithm mode 4 in the DOCA environments E1 and E2 with different vehicular mobility scenarios. PRR vs Tx-Rx distance shown with mean (solid curve) and standard deviation (shade) [39] ©2022 IEEE.

Table 6.3 Percentages of packet loss due to scheduling, and mean latency in E1 and E2 [39] ©2022 IEEE.

	E1-L 500m		**E1-L 1000m**		**E1-HL 500m**		**E1-HL 1000m**		**E2-L 500m**		**E2-L 1000m**		**E2-HL 500m**		**E2-HL 1000m**	
Tx-Rx[m]	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4
0-20	10.6	16.6	25.9	35.0	18.2	26.0	32.8	51.9	9.0	16.6	25.6	36.7	11.7	21.3	30.5	47.9
20-40	5.1	18.6	15.4	34.8	12.6	32.1	25.8	58.8	7.3	19.5	27.7	41.8	10.1	27.6	28.5	56.5
40-60	5.0	21.4	12.8	37.8	14.7	40.9	29.2	70.1	7.6	22.7	31.4	46.5	10.6	33.7	33.7	66.4
60-80	5.3	26.8	15.8	45.4	18.9	47.0	34.9	77.0	8.7	26.7	37.7	51.4	12.4	38.7	36.8	72.8
80-100	7.4	27.9	17.2	48.4	26.7	51.1	46.0	78.6	11.7	30.4	43.4	54.8	17.8	42.5	42.2	74.4
100-120	3.6	17.1	12.2	34.5	20.4	32.8	35.6	51.6	6.5	19.4	30.8	35.2	12.6	27.4	29.7	48.4
120-140	8.5	13.1	7.9	20.6	16.6	21.0	22.9	27.6	9.5	15.3	19.2	21.2	12.5	17.6	19.6	26.3
Latency [ms]	9.43	7.80	9.40	7.89	9.40	8.79	9.41	7.96	9.39	8.25	9.30	8.04	9.44	8.80	9.36	7.92

1000 s (excluding the initial warm-up phase of 200 s due to the initial random assignment of resources).

Fig. 6.5 shows that VRLS achieves better performance than mode 4 in all of the considered scenarios. VRLS is typically able to maintain a higher PRR over larger transmission ranges in both E1 and E2. The performance of mode 4 degrades more with the increasing distance between the transmitters and the receivers, mainly caused by the hidden-node problem leading to packet collisions. Beyond 100 m, the path loss effect of the wireless channel becomes dominant, and inevitably reduces the PRR of both algorithms.

To isolate the errors due to scheduling, in Table 6.3, we numerically show the percentage of the packet losses due to the scheduling of the algorithms. The percentages at each transmitter-receiver (Tx-Rx) range are calculated as the difference between the achieved mean PRR and a reference value giving the maximum possible mean PRR in the environment. The reference values represent an ideal scenario, which assumes that there are always sufficient resources for all transmissions, and the packet losses are only due to propagation errors. For convenience, we also plot the maximum possible PRR as a *reference curve* in Fig. 6.5(a), 6.5(e), and in Fig. 6.7.

From Table 6.3, we observe that VRLS has superior performance compared to mode 4 in all scenarios. Within a 100 m of Tx-Rx range, VRLS maintains a higher rate of successful packets. Beyond this range, the packet losses are predominantly caused by the propagation loss rather than the scheduling, given the low transmit power. In scenarios E1-L and E2-L with $L_{\text{DOCA}} = 500$ m, VRLS shows a performance close to the ideal scenario. The PRR for both algorithms is degraded considerably with the increased vehicular density, as well as the increased DOCA size that impact the interference conditions. In the highly-loaded scenarios, the collisions increase due to the allocation of the same resources to different vehicles. In such cases, although both algorithms perform sub-optimally given the limited number of resources, VRLS results in half the packet losses compared to mode 4.

We examined the policy that VRLS learned by observing the course of states and actions. VRLS develops a strategy to divide the resource pool dynamically into two directions of the highway, in proportion to the data traffic demand. Simultaneously, VRLS performs *resource reuse per direction*, hence mitigating the hidden-node problem. Given the loaded conditions, HD errors in the network become inevitable even though the collisions could be avoided. Namely, vehicles can be allocated to different subchannels, yet sharing the same time slot. To illustrate, with $C^{2\times 10}$, when there are 20 vehicles in the DOCA, each of the 10 slots would be shared by two vehicles using different subchannels in order to avoid any resource collisions. Yet, such an allocation would result in HD errors among these vehicles when they enter within each other's communication range. In fact, in such loaded scenarios, the probability of unsuccessful transmissions due to HD errors would be even larger if the pool consisted of fewer slots and more subchannels (e.g., $C^{4\times 5}$) as more vehicles would be required to use the different subchannels sharing each slot to avoid any collision errors. We evaluate and observe such different resource pool configurations in Section 6.3.12. Yet, VRLS learned to assign the resources with HD conflicts, i.e., the subchannels sharing the same time slot, to vehicles in the *opposite* directions rather than to the nearby vehicles in the same direction. Such assignment strategy results in comparatively fewer HD errors, as those vehicles pass by each other for a shorter duration of time. The outcome is especially observable in E1. The single-lane traffic in E1 results in a minimum inter-vehicle distance of about 40 m within a lane. The Tx-Rx distances below 40 m have a reduced PRR, which occurs only as a result of the vehicles from opposite directions passing by each other. Compared to E1, the second lane in E2 enables the vehicles to overtake each other, hence resulting in a more dynamic environment. Subsequently, the distance between the vehicles in the same direction can take any value, resulting in a smoother decrease of the PRR with the increasing Tx-Rx range, as compared to E1. The trained VRLS policy is efficiently deployable in such a dynamic environment with varying vehicular density and network load over time and space, where it can deliver higher reliability than mode 4.

6.3.6 Impact of Communication Quality of Service on V2V Applications

We evaluate the impact of PRR on the performance of V2V applications, by utilizing the awareness probability P_{A} introduced in Section 3.2. As an illustrative example, the lane-change warning application requires at least $n = 3$ messages to be received within $T = 1$ s with $P_{\mathrm{A}} = 99\%$ to make the neighboring vehicles aware of the intended maneuver [257]. Following our assumption of 10 Hz message frequency, i.e., $k = 10$, and assuming independent message errors, this translates into a PRR requirement (i.e., p given P_{A}) of 61.12%. In our multi-lane environment E2, VRLS can achieve such a PRR at up to a 120 m of range for

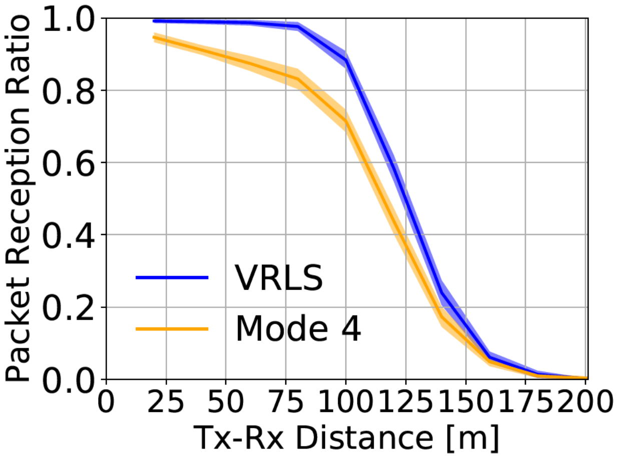


Fig. 6.6 PRR evaluation of VRLS and mode 4 with aperiodic V2V traffic in the environment E2-HL with $L_{\mathrm{DOCA}} = 1000$ m. PRR vs Tx-Rx distance shown with mean (solid curve) and standard deviation (shade) [39] ©2022 IEEE.

$L_{\mathrm{DOCA}} = 500$ m (see Fig. 6.5(e) and 6.5(g)), and up to around 80 m in $L_{\mathrm{DOCA}} = 1000$ m (Fig. 6.5(f) and 6.5(h)). In comparison, mode 4 achieves around 100 m and 80 m of an awareness range in $L_{\mathrm{DOCA}} = 500$ m under the loaded and highly-loaded traffic, respectively. In the case of $L_{\mathrm{DOCA}} = 1000$ m, mode 4 yields an awareness range of 30 m for the loaded scenario, and cannot satisfy the requirement at all for the highly-loaded scenario.

6.3.7 Performance under Aperiodic V2V Traffic

We further evaluate the performance of VRLS under event-triggered V2V traffic. The vehicles are assumed to generate a message upon each event, where the event arrivals for each vehicle follow a Poisson distribution with a rate of 1 event/s ($X_{\mathrm{evt}} = 1/\mathrm{s}$). In Fig. 6.6, we report the PRR performance of the algorithms in scenario E2-HL with $L_{\mathrm{DOCA}} = 1000$ m by considering aperiodic traffic only (not coexisting with periodic traffic). The event-triggered traffic results in less frequent V2V message generation as compared to the periodic traffic, which effectively creates a lower network load. Accordingly, the performance of both algorithms is increased (observed also in other scenarios), with VRLS achieving a PRR very close to 100% up to a range of 80 m. The results show that the policy learned by VRLS for the periodic traffic is applicable to the aperiodic type of traffic as well. On the other hand, mode 4, which is a solution primarily designed for periodic V2V traffic, underperforms in this setting.

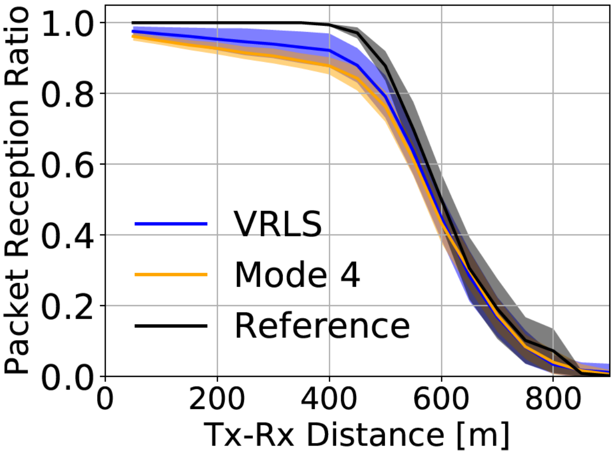


Fig. 6.7 PRR evaluation of VRLS and mode 4 in comparison to maximum possible value (reference) with $P_{\text{Tx}} = 23$ dBm and $C^{2\times 50}$ in E2-L with $L_{\text{DOCA}} = 1000$ m. PRR vs Tx-Rx distance shown with mean (solid curve) and standard deviation (shade) [39] ©2022 IEEE.

6.3.8 Performance under High Transmit Power and Larger Resource Pool

For all of the results above, the vehicle transmit powers are set to $P_{\text{Tx}} = -5$ dBm, whereas the allowed maximum for V2V transmissions is 23 dBm [282]. We selected -5 dBm to enable multiple collision domains for smaller DOCA lengths (500 m). This allowed us to simulate environments with fewer vehicles, thus taking shorter simulation times. To ensure that the performance of VRLS holds for larger and arguably more realistic communication ranges, we evaluate the performance of the algorithms with the transmission power set to 23 dBm for all vehicles in E2-L with $L_{\text{DOCA}} = 1000$ m. In order to compensate for the increased interference caused by the high-powered transmissions, we consider a pool that consists of 2 subchannels and 50 slots, i.e., $C^{2\times 50}$, which is five times larger than the resource pool configuration $C^{2\times 10}$ we have considered so far. For this scenario, VRLS is trained in E0 as well, but utilizing the resource pool $C^{2\times 50}$.

Results of the algorithms are provided in Fig. 6.7, where we see that both algorithms achieve very high reliability, close to 100% PRR at shorter Tx-Rx distances, owing to sufficiently provisioned resources. VRLS delivers marginally higher PRR than mode 4 at almost all Tx-Rx ranges. The results demonstrate that VRLS is trainable on environments having different resource pool configurations, and that the learned policy is applicable to scenarios with different transmission ranges. The small percentage of packet losses in the environment results mainly from propagation errors, but also due to HD or even collisions to small extent. In case of mode 4, vehicles cannot sense the slots they transmit on (due to HD constraint), thus there exists a probability of selecting resources used by other vehicles that might interfere or collide. In case of VRLS, although the learned policy avoids allocating the same resource to more than a single vehicle, it is challenging for the agent to learn the HD constraints in such a large and sparse state-space, where some of its assignments lead to HD errors.

6.3.9 Performance under Larger DOCA Size

We extend our evaluations to consider larger sizes of DOCA, especially to see the impact of longer periods of intermittent coverage on the reliability of resources scheduled by VRLS. After a vehicle arrives at the DOCA, interference conditions it experiences throughout the DOCA evolve over time, where the increase in DOCA size would make the task of efficient resource reuse across the DOCA even more difficult. In Table 6.4, similar to Table 6.3, we provide the percentages of packet losses due to scheduling of VRLS and mode 4, for the Tx-Rx range of 80-100 m, for different environments varying in terms of the DOCA size.

In the case of E1, since vehicles travel on a single lane per direction, interference conditions among the vehicles reusing the same resource in the same direction are not expected to change considerably over time. Our results in Table 6.4 for the environment E1-HL verify this hypothesis by showing that a larger DOCA size does not considerably impact the PRR of the algorithms in case of having only one-way traffic. On the other hand, with the addition of vehicle traffic from the opposite direction, the rate of resource collisions in E1 considerably increases with the size of the DOCA, since vehicles get longer exposure to the interference they experience from the vehicles reusing the same resource on the opposite direction. In the case of VRLS, where vehicles use their assigned resources persistently throughout the DOCA, the rate of resource collisions shows a larger increase with the size of DOCA, as compared to mode 4, where vehicles continuously re-select their resources based on the updated sensing results. In the case of E2, due to the extra lane in each direction offering an additional degree of freedom to the vehicles, further mobility deviations lead to diverse interference conditions within the DOCA over time, as its size gets larger. Our results in Table 6.4 show that larger DOCA in the case of E2 also yields a considerable increase in the rate of resource collisions for both algorithms.

6.3.10 Resource Efficiency

We observe in the above evaluations that the PRR highly depends on the network load and interference conditions in the environment, where the reliability requirements of V2V applications may not be satisfied under a given amount of traffic demand and available

Table 6.4 Percentages of packet loss due to scheduling under different DOCA sizes.

Environment	L_{DOCA}	**VRLS**	**Mode 4**
E1-L	500 m	7.4%	27.9%
E1-L	1000 m	17.2%	48.4%
E1-L	2000 m	30.0%	65.4%
E1-HL	500 m	26.7%	51.1%
E1-HL	1000 m	46.0%	78.6%
E1-HL one-way traffic	1000 m	22.1%	65.4%
E1-HL one-way traffic	2000 m	25.2%	68.2%
E2-L ($P_{\text{Tx}} = 23$ dBm, $C^{2\times10}$)	1000 m	20.3%	28.6%
E2-L ($P_{\text{Tx}} = 23$ dBm, $C^{2\times10}$)	5000 m	37.4%	55.4%

resources. Therefore, for more practical conclusions, we evaluate the resource efficiency of the algorithms as defined in Section 3.2. Namely, we first evaluate the maximum V2V distance at which different target PRR requirements can be satisfied by different algorithms for the given communication traffic load under different numbers of resources available in the network. We then evaluate the minimum number of resources required to achieve a given reliability target.

In Table 6.5, we provide the reliable V2V communication distance in meters, calculated as the maximum distance at which given PRR targets of 80%, 90%, and 95% can be achieved for different resource configurations of $C^{2\times10}$, $C^{2\times20}$, $C^{2\times50}$, thus, for a total number of 20, 40, and 100 resources, respectively. We provide the results considering the realistic setting with vehicle transmit powers set to 23 dBm in scenario E2-L with $L_{\text{DOCA}} = 1000$ m. Results in Table 6.5 show that VRLS outperforms mode 4 in terms of better making use of the available resources by achieving the given target reliability at larger distances. The performance gain is especially larger in cases of fewer resources available or higher reliability targets, where VRLS can guarantee the target reliability at V2V distances up to four times larger than that of mode 4.

Using the results in Table 6.5, we then derive the minimum required number of resources for each algorithm to achieve a reliability target at different distances between vehicles. We provide the results in Table 6.6 for reliability targets of 80%, 90%, and 95% mean PRR at distances of 50, 100, 200, and 400 m. We use least-squares fitting in case no data points are available from the measurements in Table 6.5. Results in Table 6.6 show that VRLS requires fewer resources to achieve a reliability target at all distances. While mode 4 shows a quadratic increase in the required number of resources to achieve a reliability target at larger distances, the number of resources required by VRLS shows rather a logarithmic increase.

Table 6.5 Reliable communication range for different PRR targets under different number of resources.

Number of	**80% PRR**		**90% PRR**		**95% PRR**	
Resources, *R*	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4
20	200 m	75 m	100 m	50 m	50 m	0 m
40	330 m	270 m	250 m	150 m	200 m	50 m
100	500 m	470 m	450 m	300 m	400 m	200 m

Table 6.6 Required number of resources to achieve different PRR targets.

Reliable	**80% PRR**		**90% PRR**		**95% PRR**	
Comm. Range	VRLS	Mode 4	VRLS	Mode 4	VRLS	Mode 4
50 m	9	13	16	20	20	40
100 m	12	20	20	30	25	59
200 m	20	35	32	61	40	100
400 m	59	75	80	140	100	200

This behavior of the algorithms becomes especially visible in the case of larger reliability targets. VRLS requires at most half of the resources required by mode 4 to achieve a 95% mean PRR for all ranges under consideration.

6.3.11 User Fairness, Packet Inter Reception, and Latency

Although our solution gives an equal opportunity to all vehicles to transmit in the DOCA by allocating resources, the PRR results do not provide the information on whether fairness is ensured, i.e., the PRR of a certain group of users is not sacrificed in favor of system-wide performance. In Fig. 6.8, we provide the mean and the standard deviation of the per-user average PRRs, to evaluate the variation of reliability across the users. The results are presented for the highly-loaded environment E1-HL with $L_{\text{DOCA}} = 500$ m. We observe that VRLS is able to deliver its performance without sacrificing user fairness. Both VRLS and mode 4 achieve a similar variation of mean PRR across the users, where the standard deviation is around 0.025 considering all Tx-Rx ranges.

In Fig. 6.9, we report another *per-user* performance metric, PIR (cf. Section 3.2), in the environment E2-HL with $L_{\text{DOCA}} = 500$ m, measured within a 50 m of Tx-Rx range, in terms of mean and percentiles. We observe that VRLS achieves better performance than mode 4, the latter resulting in at least 25% larger PIR on average. Note that for both algorithms, all quartiles of the PIR are at 100 ms, which is equal to T_{m}. We have observed that the

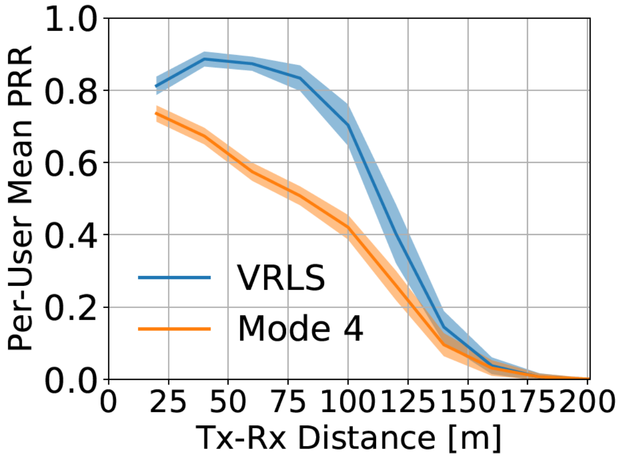


Fig. 6.8 Per-user mean PRR of VRLS and distributed scheduling algorithm mode 4 on the environment E1-HL with $L_{\text{DOCA}} = 500$ m; shown with mean (solid curve) and standard deviation (shade) [39] ©2022 IEEE.

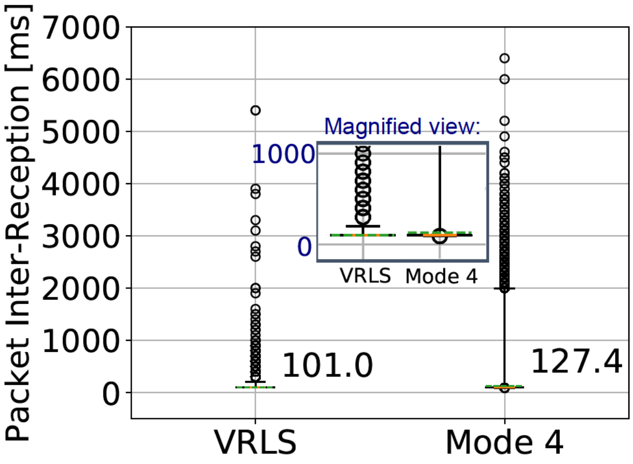


Fig. 6.9 PIR performance of the algorithms in E2-HL with $L_{\text{DOCA}} = 500$ m, shown with mean (green, dashed, denoted): $101.0^V, 127.4^M$; median (orange): $100.0^V, 100.0^M$; 25th and 75th percentiles (black): both $100.0^V, 100.0^M$; 0.1st and 99.9th percentiles (whiskers): $100.0^V, 100.0^M$ and $200.0^V, 2000.0^M$; and outliers (rings), measured at Tx-Rx distances of $0-50$ m (V: VRLS, M: Mode 4, values in ms) [39] ©2022 IEEE.

relative PIR performance of the algorithms in the other scenarios are also similar, where VRLS achieves a mean PIR close to 100 ms, at most reaching 106 ms, and mode 4 resulting in mean values ranging from around 115 ms up to 200 ms that mainly increase with the network load. For the high-transmit-power scenario in Fig. 6.7, as the best case, VRLS and mode 4 can achieve a mean PIR of 100.4 and 100.6 ms, respectively. Combined with our analysis in Section 6.3.6, it is evident that VRLS is able to provide superior awareness to the vehicular users while maintaining fairness, which benefits the V2V applications.

We report the mean latency measured in all evaluation scenarios in the last row of Table 6.3. Note that the messages are at least delayed by the processing time across the communication layers, which is assumed to be 4 ms [236], and at most delayed by the time-length of the resource pool, which is 10 ms, plus the processing time, as all messages are scheduled within the utilized resource pool by both algorithms. Both VRLS and mode 4 yield a similar latency of around 9 ms, on average.

6.3.12 Handling Different Resource Pool Configurations

The network operator might need to configure different resource pools that may vary in terms of the number of resources in time and frequency, i.e., K and M, e.g., to serve different V2V services with different traffic requirements or depending on the availability of resources. Therefore, we are interested in studying whether and how VRLS can handle a *set* of different resource pools configured with different number of resources in time and frequency, with a *single* training. Note that the pool configurations are usually a part of network planning; thus the configuration information would be available before operating the scheduler. Given this information, having a single policy that can achieve an appropriate performance level under all different configurations would be desirable, instead of training multiple ones that can only operate on a specific configuration. That said, it is a challenge to learn and solve the HD and collision constraints of different pools at once by a single policy, as each has a different impact on the performance. Such an approach requires careful consideration of the state and the other RL components.

We describe our method to train a single policy for multiple resource pool configurations as follows. Consider a set of different resource pool configurations $\{C_1^{K_1 \times M_1}, C_2^{K_2 \times M_2}, ...\}$ to be operated by the network, where each pool C_i consists of a different number of subchannels K_i and slots M_i. We first determine a superset ("master") pool configuration $C_{\mathrm{ms}}^{K_{\mathrm{ms}} \times M_{\mathrm{ms}}}$, which can accommodate any configuration in the set. Accordingly, the dimensions of C_{ms} are selected as $K_{\mathrm{ms}} = \max(K_1, K_2, ...)$ and $M_{\mathrm{ms}} = \max(M_1, M_2, ...)$. VRLS is provided with a state- and an action-space having the same number of resources as in C_{ms}, i.e., $K_{\mathrm{ms}} \times M_{\mathrm{ms}}$. As illustrated in Fig. 6.10, considering a case where the network operates four different resource configurations $\{C_1^{2\times 10}, C_2^{4\times 5}, C_3^{5\times 4}, C_4^{10\times 2}\}$, a master pool of $C_{\mathrm{ms}}^{10\times 10}$ accommodates all four; thus the state- and the action-space of VRLS consist of 100 resources. We provide the different pool configurations to different groups of actors training the VRLS policy in parallel. Each group of actors i is trained with the pool configuration C_i, where we only "disclose" the resources of C_i within s_t by replacing the rows corresponding to other resources

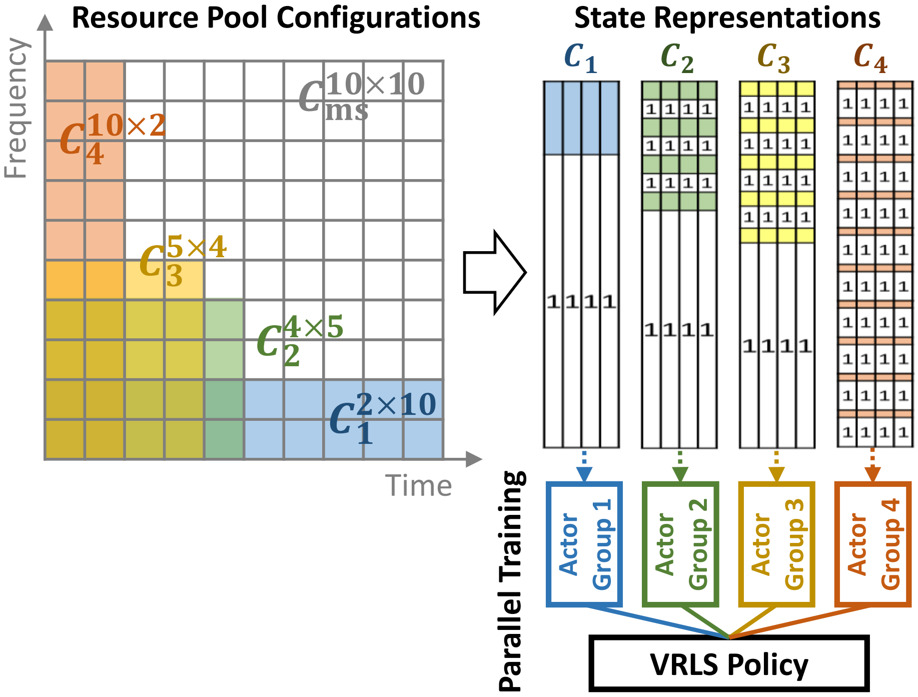


Fig. 6.10 Training VRLS with multiple resource pool configurations in parallel (adapted from [39] ©2022 IEEE).

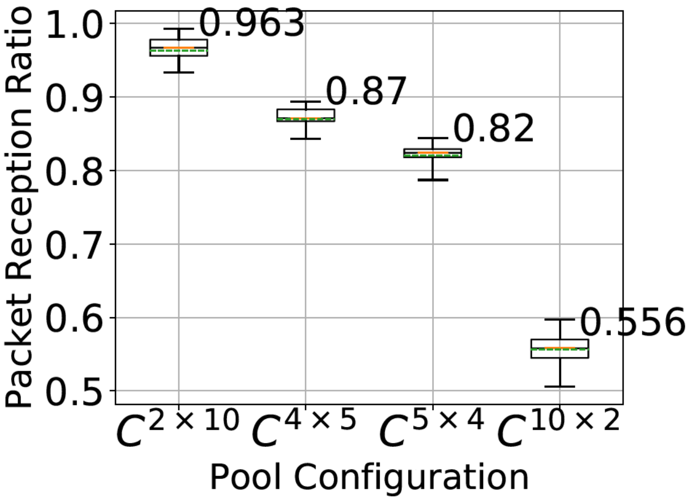


Fig. 6.11 PRR performance of VRLS on different resource pool configurations $C^{K\times M}$ having K subchannels and M slots, with mean (green, dashed, denoted), median (red), 25th and 75th percentiles (box), and 5th and 95th percentiles (whiskers) [39] ©2022 IEEE.

with the row vector $[1,1,1,1]$. Further, if the actor selects such a resource, we provide a large negative reward and execute no actions until the actor selects a resource within its own pool. With such training, we aim at limiting the action selection of the policy only to the represented subset of the resources in s_t.

In the following, we evaluate our solution for four different configurations $\{C_1^{2\times10}, C_2^{4\times5}, C_3^{5\times4}, C_4^{10\times2}\}$. VRLS is trained from scratch with a total of 40 actors in parallel, in four groups of 10 actors. Each group is provided with one of the four different configurations. If a actors selects a resource outside its configuration, a reward of $r_{t+1} = -10$ is provided. In turn, to compensate for the higher variance in the rewards, the training epoch length is increased to 200 actions. We evaluate VRLS in a DOCA similar to E0, with 10 vehicles initially placed on the highway with transmission range $R_{\text{Tx}} = 500$ m, resulting in a single collision domain (transmissions using the same TB are assumed to collide). Such a simple setting enables a deterministic calculation of performance bounds and better evaluation of whether the learned policy can deal with the constraints of the different resource pools. Namely, in the case of $C_1^{2\times10}$, when all 10 vehicles reside in the DOCA, a 100% PRR would be achievable only if vehicles were assigned to TBs in different slots. With $C_2^{4\times5}$, the allocation for all 10 vehicles would be optimal if all TBs were assigned orthogonally first in time, then in frequency. Every transmission would be received by all other vehicles except the one transmitting in the same slot, due to the HD constraint. Thus, the best assignment of TBs would result in eight successful receptions out of nine receiving vehicles, yielding an $88.\bar{8}\%$ PRR. Similarly, in the case of $C_3^{5\times4}$ and $C_4^{10\times2}$, when all of 10 vehicles exist in the DOCA at the same time, orthogonal assignment of TBs first in time and then in frequency would yield $82.\bar{2}\%$ and $55.\bar{5}\%$ PRR, respectively.

In Fig. 6.11, we report the performance of VRLS when applied to the network with different resource pool configurations, in terms of the PRR measured up to a 500 m of Tx-Rx range. We observe that VRLS yields a mean PRR almost equal to the calculated bounds of 1.0, $0.\bar{8}$, $0.8\bar{2}$, and $0.\bar{5}$ for the configurations $C_1^{2\times10}$, $C_2^{4\times5}$, $C_3^{5\times4}$, and $C_4^{10\times2}$, respectively. Note that the larger PRRs are reached when fewer than the maximum of 10 vehicles reside in the DOCA. VRLS is able to achieve such performance by learning a single policy that can handle distinct constraints of HD and collisions for different pools simultaneously.

6.3.13 Learning Performance

We provide the learning curves of VRLS in the training environment E0 with different resource pool configurations that we considered throughout our evaluations, in Fig. 6.12. The curves represent the average reward collected by the trained actors versus the number of training epochs. VRLS converged to a stable performance level after around 1000 epochs when trained with a single resource configuration $C^{2\times10}$ as per Section 6.3.5. With the larger resource pool of $C^{2\times50}$ as per Section 6.3.8, it took around two times longer for the agent to converge. This is because of additional exploration required by the increased state and the action space. VRLS obtained a larger average reward in the case of $C^{2\times50}$ owing to the

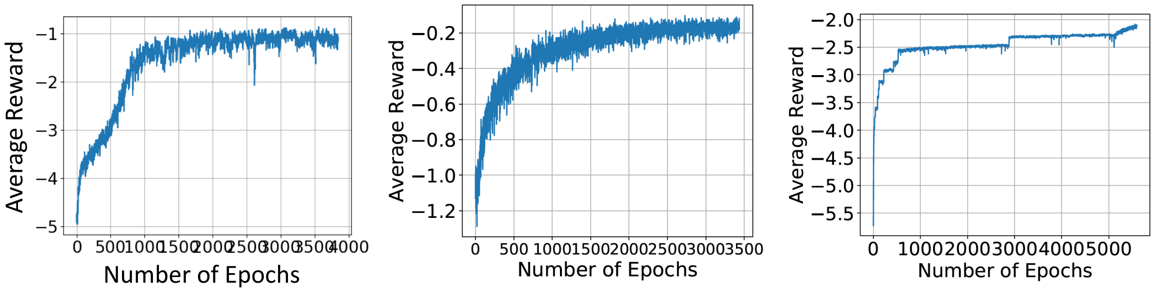


Fig. 6.12 Learning curves of VRLS on the training environment E0 with resource pool configuration $C^{2\times10}$ (left), $C^{2\times50}$ (center), and with multiple configurations $\{C_1^{2\times10}, C_2^{4\times5}, C_3^{5\times4}, C_4^{10\times2}\}$ (right) [39] ©2022 IEEE.

sufficiently provisioned resources in the network. When VRLS is trained with four resource pool configurations in parallel, i.e., with $\{C_1^{2\times10}, C_2^{4\times5}, C_3^{5\times4}, C_4^{10\times2}\}$ as per Section 6.3.12, it took a longer time for the algorithm to converge as compared to the training with a single pool configuration. This is due to the different collision and HD constraints posed by each different resource pool configuration that VRLS needs to learn, as well as the larger state space, which results in slower convergence. The collected reward is smaller as it represents the average of dissimilar performance levels on the different pools reported in Fig. 6.11. The overall performance is largely converged, which could be yet further optimized, such as via exhaustive training on the desired configuration, or with a larger number of actors, however, calling for increased training time and resources.

6.3.14 On Real-world Implementation of VRLS

We train and evaluate the performance of VRLS in simulative environments. Yet, the proposed methods might as well be implemented in a real-world vehicular network. In a real-world scenario, network vendors or operators would implement VRLS as an intelligent controller deployed at the edge of the network and integrated into the radio access network (RAN), thanks to the enabling architecture envisioned for 5G and beyond networks [98], which we have discussed in Section 2.3.3. Within this architecture, BSs deployed at the entrance/exit of the DOCA can be realized as remote radio units. While these radio units serve physical layer functions, they are connected to a centralized entity that is responsible for resource allocation and other higher-layer functionalities, where VRLS can be implemented. By implementing VRLS, operators would aim at ensuring seamless quality of V2V communications when vehicular users experience coverage losses. This would in turn ensure safer and more efficient road traffic. With regards to deployment and operation costs of VRLS, since being a logical entity, it can be implemented as software and can make use of the processing hardware available at the network infrastructure. Additional processing power is necessary to train the RL agent (with high processing requirements and possible need to pre-train in the simulated environments), and to operate it (with relatively low processing requirements). VRLS can again benefit from the fact that in 5G RAN, there are more computational resources deployed at the network edge to train learning algorithms [98].

To train and operate VRLS, the BSs delimiting the DOCA would collect and report the required data constituting the state information input to VRLS as described in Section 6.2.1. The BSs can easily keep track of the time of entry t_i and assigned resource r_i for each vehicle, autonomously, thus not requiring any additional signaling between the vehicles and the BS. Further, the BSs can obtain the information pertaining to d_i and v_{avg} from the regular V2V traffic such as CAMs that vehicles transmit, thus again not requiring any additional signaling. The collected information at the BSs is forwarded to the centralized agent when an action is required. In turn, the actions of VRLS, namely the resource allocation, will be signaled to the vehicles via the BSs, before they enter the DOCA from the respective direction. Considering a pool configuration of $C^{K\times M}$, signaling of an assigned resource would consist of $\log(K\times M)$ bits of information. Assuming a vehicle traffic of 0.4 vehicles/s/lane arriving at a DOCA with 3 lanes per direction [236], and a pool of $C^{2\times 100}$, this would correspond to $\sim$ 8 bits/s of downlink data traffic per BS.

In Table 6.7, we provide the algorithmic complexity of VRLS during its real-time operation (i.e., online inference phase), by decomposing it into two stages: i) the calculation of s_t as given by Algorithm 4; and ii) the processing of s_t by the trained CNN to select the resource as described in Section 6.2.3. In stage (i), assuming V vehicles inside DOCA, the Step 1 of the Algorithm 4 would take V multiplications, followed by the search of length V in Step 2. At Step 5, for each resource and direction, i.e., $2KM$ times, the algorithm makes a search again of length V. Sorting operation at Step 7 can be implemented with $\mathscr{O}(V\log V)$ complexity. Altogether, stage (i) yields a time-complexity of $\mathscr{O}(KMV\log V)$. In stage (ii), the input layer processes each column of s_t of length KM by 16 convolutional filters of size 10, followed by element-wise activation function over $16(KM-10+1)$ elements at the layer output. Thus, a total of $4\times 16(10+1)(KM-10+1)$ operations are performed at the input layer, which brings a time complexity of $\mathscr{O}(KM)$. The hidden layer with 32 filters of length 10 performs 320 multiplications over the concatenated output of the input layer, which has size $S_{\text{con}} = 4\times 16(KM-10+1)$. This amounts to $32(10+1)(S_{\text{con}}-10+1)$ operations at the hidden layer, thus having a time complexity of $\mathscr{O}(KM)$. At the output layer, fully connected layer with KM units performs $KM32(S_{\text{con}}-10+1)$ multiplications, which brings $\mathscr{O}(K^2M^2)$. Having greater complexity than previous ones, this layer determines the complexity of stage (ii). Altogether, when both stages are combined, VRLS yields a time

Table 6.7 Algorithmic complexity of VRLS (adapted from [39] ©2022 IEEE).

	Time complexity	**Space complexity**
Stage (i): Calculation of s_t (Steps of Algorithm 4)	Step 1: $\mathcal{O}(V)$ Step 2: $\mathcal{O}(V)$ Step 5: $\mathcal{O}(KMV)$ Step 7: $\mathcal{O}(KMV \log V)$ Overall: $\mathcal{O}(KMV \log V)$	$3V+2+4KM$ variables ($t_i, r_i, d_i\ \forall i = 1,...,V$, v_{avg}, t_{now}, and s_t) Overall: $\mathcal{O}(V+KM)$
Stage (ii): CNN processing	Input layer: $\mathcal{O}(KM)$ Hidden layer: $\mathcal{O}(KM)$ Output layer: $\mathcal{O}(K^2M^2)$ Overall: $\mathcal{O}(K^2M^2)$	Input layer: $4 \times 16 \times 10$ variables Hidden layer: 32×10 variables Output layer: $32 \times (4 \times 16(KM - 10 + 1) - 10 + 1) \times KM$ variables Overall: $\mathcal{O}(K^2M^2)$
Total	$\mathcal{O}(K^2M^2 + KMV \log V)$	$\mathcal{O}(K^2M^2 + KM + V)$

V: number of vehicles within DOCA; K, M: number of subchannels and slots of the resource pool configured for V2V communications.

complexity of $\mathcal{O}(K^2M^2 + KMV \log V)$ during its real-time operation, thereby allowing a practical implementation.

In terms of memory requirements, the algorithm in stage (i) stores $3V+2+4KM$ variables (3 per vehicle, v_{avg}, t_{now}, and s_t having $4KM$ entries), which results in a space complexity of $\mathcal{O}(V+KM)$. At stage (ii), CNN stores a total of $4 \times 16 \times 10$, 32×10, and $32 \times (4 \times 16 \times (KM - 10 + 1) - 10 + 1) \times KM$ parameters at its input, hidden, and output layer, respectively, hence yielding a space complexity of $\mathcal{O}(K^2M^2)$. Overall, space complexity of VRLS is $\mathcal{O}(K^2M^2 + KM + V)$, which is practical from the implementation point of view.

For training VRLS, it is possible to collect the reward signal from the network also in a real-world implementation. For example, vehicles could keep track of sent/received V2V message IDs with time and location stamps, which they report to the BSs after going back to the coverage. In turn, the BS calculates the PRR using this information to derive the reward. Such report sent by each vehicle would consist of the IDs and time/location stamps of the messages it has transmitted and received during its past travel within the DOCA. We illustrate the incurred overhead with an example setting as follows. Assuming an average vehicle speed of 50 km/h with an arrival rate of 0.4 vehicles/s/lane, there will be 173 vehicles in a DOCA of 1000 m with 3 lanes/direction at a given time, on average. It would take 72 s on average for a vehicle to travel through the DOCA, where it transmits 720 V2V messages, and receives at most 123 840 messages from other vehicles, assuming a message transmission rate of 10 MHz (i.e., $T_{\text{m}} = 100$ ms [236]) per vehicle, and all transmissions being successfully received by all vehicles. Further assuming that vehicle IDs are represented with 10 bits of information, and it takes 16 bits to represent the timestamp [18] and 64 bits to represent the location stamp of each message [288], each vehicle would then collect and report 1.40 MB of information to the BS. This would correspond to around 1.69 MB/s of uplink traffic per BS on average. The delay in gathering the information does not pose a limitation for training, since the agent acquires experience (the sequence of state-action-reward tuples) in batches before each training step.

On the other hand, training VRLS in simulative environments (and, if needed, re-training during a real-world usage) would circumvent numerous challenges associated with real-world training from scratch. By simulation, it is easier to create and collect sufficient data; hence the training becomes more flexible and less time-consuming. Besides, the costs of additional signaling and processing overhead at the network entities and at the vehicles required to collect data would be avoided.

6.4 Conclusions

In this chapter, we proposed VRLS, a unified RL-based approach to centrally scheduling V2V communications in a DOCA. We showed that VRLS outperforms the state-of-the-art V2V scheduling algorithms by: i) learning about the collisions in the case of non-orthogonal resource assignment to nearby vehicles; and ii) learning that half-duplex (HD) constraint needs to be accounted for. Moreover, VRLS can be trained on a wider range of environments and resource configurations than what would be practically doable in the real world. By training in simulated vehicular environments, VRLS can learn a scheduling policy that is robust and adaptable to environmental changes.

In terms of V2V communication reliability, VRLS outperforms the state-of-the-art mode 4 scheduler by reducing the packet loss by half in case of overloaded network conditions, and performing very close to the maximum possible level under low load. To achieve the reliability targets required by the V2V applications, VRLS requires a much smaller number of resources, while providing higher reliability at larger V2V distances in comparison to mode 4. Furthermore, while achieving similar fairness and V2V communication latency as mode 4, VRLS provides higher awareness among the vehicles.

VLRS has achieved such performance thanks to our design that considered a unified state, reward, and action definition for the RL model so that the structure of all three components remains the same, irrespective of what kind of setting they are applied to. Most importantly, for the state representation, we have aimed at capturing the fundamental features of the vehicular environment that are in particular relevant to resource allocation, such as resource utilization and interference conditions, instead of trying to represent every detail in the environment such as velocity of each individual vehicle. We have further organized this information in a condensed manner such that the size of the state information does not grow with the number of vehicles or the area size. In addition, we have arranged the state variables in a way to respect the separate directions of the road traffic and the half-duplex constraints among the resources. The structure of the DNN we have considered allowed efficient processing of the state information to make decisions on the resource allocation. The policy of the scheduler, i.e., the DNN, was trained with a reward directly reflecting our goal of maximizing the reliability of V2V transmissions, again irrespective of the number of vehicles or resources. These design principles allowed broad applicability of VRLS to different practical scenarios we considered, varying in terms of vehicle density, resource pool configuration, and radio propagation conditions. Further, such design eliminated the need for targeted (re-)training in complex, realistic environments.

Chapter 7

Hybrid Centralized Reinforcement Learning and Distributed Sensing Scheduler for V2V Communications

7.1 Motivation and Contribution

As introduced in Section 2.3, conventionally there are two approaches to scheduling V2V communications: distributed and centralized. While the distributed approach does not require any network infrastructure, and hence can operate irrespective of any network coverage, it is not efficient from the resource utilization point of view. A centralized scheduler can do resource allocation more efficiently, by maintaining a global view of the network based on the collected information from vehicles related to their traffic, mobility, etc. On the other hand, the centralized approach is also limited by the information available from the vehicular environment. Any unseen or unexpected conditions on the V2V links may degrade the communication performance. This becomes especially an important challenge as we consider the problem of coordinating the resources for V2V communications taking place beyond the network coverage. Furthermore, the approach we have taken in the earlier chapters, to assign a single resource per each vehicle only once (before it enters the out-of-coverage area), might become unfeasible under certain conditions encountered outside the coverage.

While VRLS that we introduced in Chapter 6 is able to perform well under different environments varying in terms of vehicular mobility or wireless channel conditions; as an RL-based solution, its performance is subject to degradation when the distribution of the environment variables goes much beyond the one that it encountered during training.

Parts of Chapter 7 including the results have been published in [36] ©2019 IEEE.

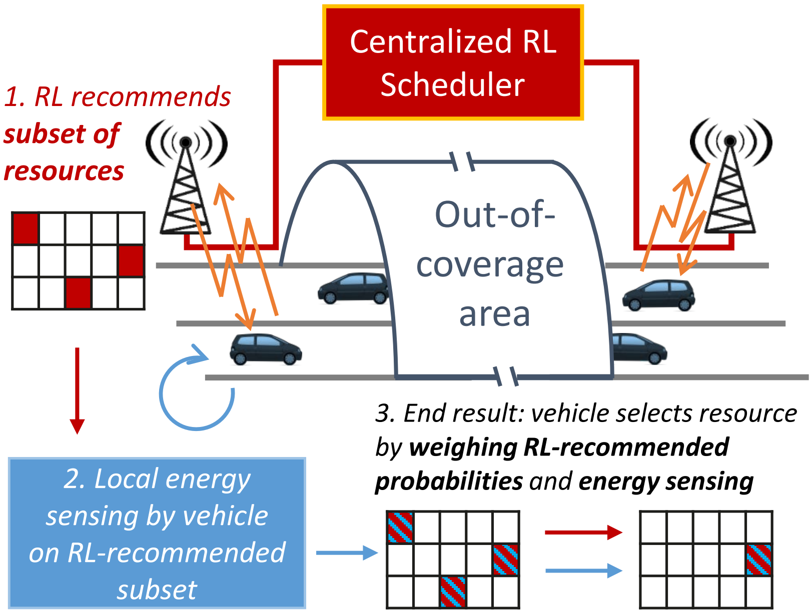

Centralized RL Scheduler
1. RL recommends subset of resources
Out-of-coverage area
2. Local energy sensing by vehicle on RL-recommended subset
3. End result: vehicle selects resource by weighing RL-recommended probabilities and energy sensing

Fig. 7.1 Hybrid sensing- and reinforcement-learning-(RL)-based approach for scheduling V2V communications in delimited out-of-coverage area (DOCA) [36] ©2019 IEEE.

Specifically, in this chapter, we evaluate its performance in the case of an exceptional event, namely a traffic accident on the road creating over congestion and stop-and-go traffic. The policy developed by VRLS for highly congested scenarios is to schedule all transmissions above the channel capacity into a single resource that it "sacrifices" for the sake of saving all other resources from collisions, similar to a congestion control mechanism. While this policy would work up to a certain extent in a free flow of road traffic, it fails in the scenario we consider in this chapter since a large number of vehicles using the "sacrificed" resource queue up in the same vicinity due to the accident, and their transmissions collide.

In this chapter, motivated by the distinct challenges that both centralized (including our RL-based) and distributed schedulers face, and also by the question of whether joining the forces of centralized and distributed approaches together would bring any benefit as compared to either of them, we propose a *hybrid* approach, which combines the centralized RL-based solution with the distributed sensing mechanism. In this approach, we employ a centralized RL scheduler, which first recommends a *subset* of resources to the vehicles going outside coverage, with *associated probabilities* of selecting each resource. Using the allocated subset of resources, vehicles locally apply an energy-sensing mechanism to determine the final resource they will transmit their V2V messages by weighing and combining the RL-recommended probabilities with the sensing results. With this approach, RL takes the global network view into account and provides a high-level strategy for resource allocation, whereas the sensing mechanism selects among the RL-selected subset of resources based on local and hard-to-predict (dynamic, ephemeral) conditions. Our ultimate goal in designing the hybrid solution is to enable the benefits of RL-based centralized solution, *while also ensuring that the algorithm can adjust to a large range of unforeseen scenarios* for which the RL agent was not trained. Examples of such scenarios are extreme traffic jams, construction works, wrong-way highway driving, etc.

In the literature, existing work on hybrid approaches that combine centralized and distributed methods for scheduling V2V communications is limited, as we discussed in Section 2.4. So far, the works mainly considered scheduling device-to-device (D2D) communication between users co-existing with cellular ones to optimize system-level performance metrics [289], [290], [291]. Targeting vehicular users, the work in [149] proposes the *co-existence* of two radio interfaces per vehicle, enabling switching between the centralized cellular-based D2D and distributed 802.11p mechanisms for latency optimization. A hybrid solution more relevant to our approach is proposed by [150], where vehicular users select resources that are divided into geographical zones, by estimating co-channel interference and comparing with threshold values determined by the base station, to maximize link reliability and sum rate. To the best of our knowledge, no hybrid approach that targets resource allocation for outside-coverage vehicular users has been yet proposed in the literature.

We evaluate the performance of our hybrid solution in terms of the reliability of periodic broadcast V2V transmissions in a predicted out-of-coverage area, namely in DOCA, and compare it with the centralized VRLS and the distributed mode 4 algorithms. Results indicate that the hybrid approach outperforms both in highly dynamic scenarios, and is at least as good in other scenarios. As such, the hybrid approach can complement VRLS, where it can be employed upon detecting unusual conditions in the network, such as an accident, for which the VRLS agent is not trained.

The rest of this chapter is organized as follows. Section 7.2 describes our proposed hybrid method, Section 7.3 presents the evaluation results, and Section 7.4 concludes the chapter.

7.2 Hybrid Reinforcement Learning and Sensing Scheduler

The hybrid approach we propose is composed of two main components: a centralized RL agent providing resource “recommendations” to the vehicles, and a resource selection procedure by the vehicles that combines the recommendation and the sensing information they collect. As illustrated in Fig. 7.1, the RL agent is responsible for allocating each vehicle entering DOCA a probability distribution over the resources in the resource pool, where probabilities indicate a preference of selecting each resource. The vehicle utilizes this information combined with its sensing information, each time it (re-)selects a resource for its transmissions inside DOCA. The exact resource selection procedure per vehicle consists of the following steps.

1. Gather the probability distribution over resources from the centralized RL scheduler (via BS) timely before going OOC, and select the top 20% recommended resources (i.e., the ones with the highest probability values) according to the distribution. Keep the selected set of resources throughout the DOCA, along with the associated probability values.

2. Gather sensing results on the selected set of resources, by running a sensing window as in the mode 4 algorithm to collect RSSI measurements from the last 1000 time slots, and calculate the average RSSI value for each resource. For the resources that could not be sensed (due to HD constraint), use the average of the calculated values of other resources.

3. To determine the final resource to transmit, multiply the RL-recommended probability values with the average RSSI values for each resource, and select randomly according to the weighted distribution.

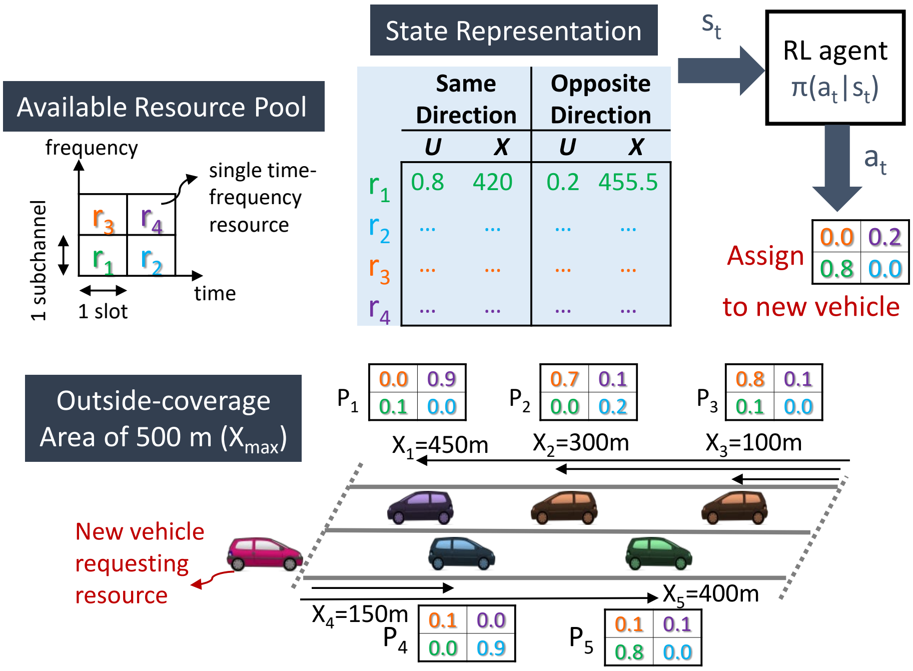
4. Use the selected resource for the next certain number of transmissions determined by the random counter C_{resel} as in mode 4, then reselect a new resource by repeating the steps 2 and 3.

The core idea of our algorithm is to have persistent recommendations on the resources from the centralized RL-based scheduler, where it effectively provides a subset of resources with selection weights, while letting the vehicle select the final resource considering the sensing information it collects dynamically from its local environment. This way, the RL-scheduler having the global view of the environment determines a high-level strategy for resource allocation, whereas sensing helps the vehicle to select actual resources based on transient and hard-to-predict conditions surrounding it.

7.2.1 Design of the RL Agent

We reuse our design of VRLS in Chapter 6 for the centralized RL-based scheduler, however, in a probabilistic fashion to form and interpret the agent's input and output, i.e., the state and the action, respectively, as described in the following.

In order to allocate resources to the vehicles, the RL scheduler makes use of the information about outside coverage. The RL agent observes the state s_t of the environment,



	Same Direction		Opposite Direction	
	U	*X*	*U*	*X*
r1	0.8	420	0.2	455.5
r2	...	...	...	...
r3	...	...	...	...
r4	...	...	...	...

Fig. 7.2 Radio resource pool (top left) and RL state representation with the agent (top right) for an exemplary scenario (bottom) [36] ©2019 IEEE.

containing information about each resource regarding its utilization U, and the most probable location X, per direction, at each time t a vehicle i enters DOCA. The agent then outputs probability distribution P_i for that vehicle i, indicating probability of using each resource in the resource pool, which sum up to unity. An example state representation is illustrated in Fig. 7.2. Inside a DOCA of length $L_{\text{DOCA}} = 500$, i.e., $X_{\max} = 500$ m, vehicles with indices 1 to 5 have allocated probability distributions P_1 to P_5, at distances X_1 to X_5 (estimated at the scheduler by multiplying the time passed since each assignment with the average speed of the vehicular traffic) measured from the respective entrances of DOCA, and a new vehicle entering that needs to be yet allocated is given. There are only 4 resources available in the resource pool, denoted with $r_1 - r_4$. U is calculated as the sum of probability values per resource that is previously assigned to the vehicles on the corresponding direction. E.g., for r_1, U for on the same direction as the entering vehicle is calculated by adding P_4 and P_5, i.e., $0.0 + 0.8 = 0.8$, or for the opposite direction U is found by adding P_1, P_2, and P_3, i.e., $0.1 + 0.0 + 0.1 = 0.2$. On the other hand, X is calculated by considering the conditional probabilities whether each vehicle is using the resource or not, along with their estimated distances. E.g., X of r_1 on the same direction as the entering vehicle can be calculated by $P_4X_4 + (1 - P_4)P_5X_5 + (1 - P4)(1 - P5)X_{\max}$, i.e., $0.0 \times 150 + (1 - 0.0) \times 0.8 \times 400 + (1 - 0.0) \times (1 - 0.8) \times 500 = 420$.

We have designed the state space in a way to provide a compact representation of the environment to the RL agent. U indicates how much each resource is occupied in each direction, probabilistically, and X gives proximity information of the allocations, that is essential for reusing the resources in each direction. Moreover, load per direction can be derived by accounting for the total utilization from the first and the third columns. Additionally, the representation carries the direction information of the incoming vehicle by ordering the two pairs of columns by the respective same-direction first, and the opposite-direction next. Note that, the representation requires only the *available* information at the BSs. Namely, the state variables could be calculated on-the-fly by the cooperating BSs that keep the track of their past allocations to the vehicles entering and leaving the DOCA at both ends, and provide information to the central scheduler.

Following our VRLS design in Chapter 6, we further normalize the state variables with respect to their maximum possible values $U_{\max}$ and $X_{\max}$, where $U_{\max}$ is derived by dividing $X_{\max}$ by the vehicle length. Our normalized state representation makes the agent capable of dealing with different vehicular environments, irrespective of the number of vehicles, resources, or size of the DOCA. In addition, normalized variables make the learning more efficient by limiting the size of the state space. Furthermore, in order to avoid the unnecessary dependency of the agent's outcomes on the *ordering* of the resources in the state, we randomly shuffle the rows of the state matrix when providing to the agent, while preserving the HD relations, as proposed in VRLS (cf. Section 6.2).

Despite its compactness, our state space can contain many possible combinations of resources and their utilization information, which leads us to apply approximate RL methods [164]. Accordingly, we *parametrize* the policy, as well as the value function that is used to train it, using two DNNs called *actor* and *critic* networks, respectively, as in VRLS. Taking the state representation as its input, the actor outputs the action probabilities, i.e., probability of selecting each of the resources in the pool, as illustrated in Fig. 7.2. Following the VRLS design, this is established by the fully connected final layer of the actor-DNN with number of units equal to the number of resources in the resource pool. The layer utilizes softmax activation function to produce the probability distribution over the units, i.e., resources. However, differing from the VRLS agent that *pre-selects* a single specific resource to be assigned to a vehicle based on this distribution, the agent in our hybrid approach provides the *complete* probability distribution to the vehicle, leaving the final selection to it, using the mechanism described in Section 7.2.

As in VRLS, we use Algorithm 3 we have provided in Chapter 5 to train the actor-critic DNNs, by executing multiple actor-learners in parallel, which is shown to be more efficient in time and asymptotic performance [173] (cf. Section 2.5.3). Each actor-learner interacts with a different instance of the environment. After every *epoch* of state-action-reward sequence, it updates the parameters of the DNNs used for the policy and the state values for all agents. For the training, we use the same reward definition as that of VRLS (described in Section 6.2.5).

7.3 Evaluation

In this section, we evaluate and compare the performance of our hybrid solution with two algorithms: centralized VRLS we presented in Chapter 6 and the distributed mode 4 scheduler. We consider two specific scenarios that the VRLS and the hybrid RL agents may face in reality: Scenario 1) an environment that has "regular" road traffic conditions and similar characteristics to training environment of the agents; and Scenario 2) an environment which has extreme, unexpected road traffic conditions, which are also never observed by the agents during their training. We also consider the cases in between by extending Scenario 1 with two variants: 1-a) V2V transmit powers increased to the maximum; and 1-b) speed deviation of the vehicles is doubled, and evaluate how algorithms perform under these conditions deviating from the training environment. The details of the evaluation environments and the training environment of the RL agents are presented in the following subsections, and summarized in Table 7.1.

7.3.1 Training Environment and Learning Performance

For training purposes, we consider a simplistic environment while respecting the main characteristics of the vehicular network, with the goal of enabling a faster and efficient training, as described in Section 3.1.6. A DOCA of 500-m-length with 30 constant-speed vehicles, with their speeds drawn from $\sim N(75,25)$ km/h, is assumed. To maintain a constant-load traffic, vehicles are wrapped-around upon leaving the DOCA, yet, with an exponentially distributed time offset with 2.5 s mean [236]. A screenshot of the simulated mobility in the environment is provided in Fig. 7.4(a). Vehicles transmit periodic broadcast V2V messages with $T_{\mathrm{m}} = 100$ ms. We assume a resource pool of 2 subchannels and 10 slots, in order to simulate loaded conditions, considering the number of vehicles in the environment. Wireless channel model is abstracted to constant transmission ranges of $R_{\mathrm{Tx}} = 120$ m, where an unsuccessful reception is assumed if the ranges of the transmitters using the same resource intersect, or due to the HD constraint. Further details are provided in Table 7.1.

Training of the hybrid and the VRLS agents are conducted on the same environment, with training parameters presented in Table 7.1. For calculating the reward during the training of both agents, PRR is measured at $0-100$ m Tx-Rx distance, as described in Section 3.2.

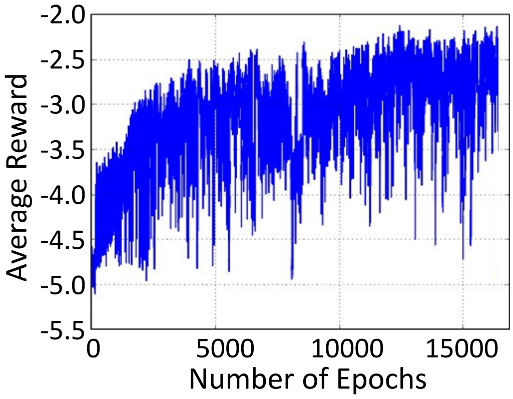


Fig. 7.3 Learning curve of the RL agent for the hybrid solution on the training environment.

Different than VRLS, when training the hybrid agent, vehicles (re-)select their resource every 10 transmissions randomly according to the probability distribution they receive from the agent, without incorporating any sensing mechanism during the training. Whereas in training VRLS, a vehicle utilizes the same resource provided by the VRLS agent for its all transmissions through the DOCA. We report the learning curve of the hybrid agent in Fig. 7.3. The learning performance showed convergence after around 15000 epochs, which is an order of magnitude larger than that of the VRLS agent (cf. Fig. 6.12). This is mostly due to recurring re-selection of resources by the vehicles randomly, rather than utilizing the one-off selected action (resource) provided by the agent, where the added randomness into the environment makes learning difficult.

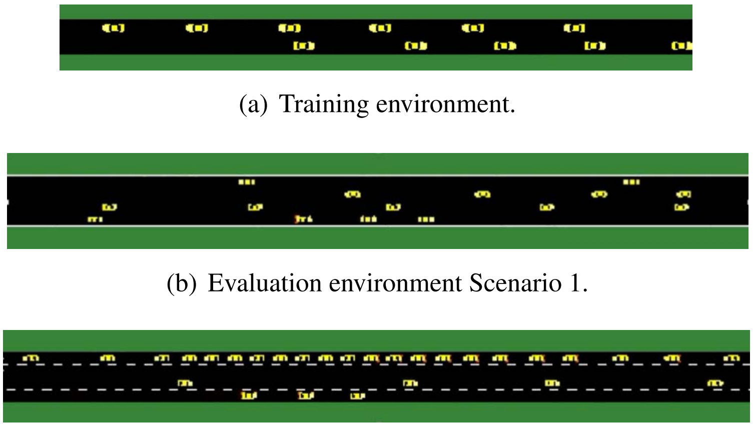
7.3.2 Evaluation Environments and Results

For the evaluation environments, we consider realistic channel model and mobility as described in Section 3.1. We assume a DOCA of $L_{\text{DOCA}} = 1000$-m length with two lanes per direction. From both ends, vehicles arrive with a probability of 0.4/s [236], with time-varying speeds initially drawn from a normal distribution, which also accounts for a time-varying traffic load in DOCA. The simulation parameters are provided in Table 7.1.

We first evaluate the performance of the algorithms in Scenario 1, where vehicle speeds are normally distributed around 120 km/h with 10% deviation, resulting in a total number of vehicles between around 15 and 45 at a given time in DOCA. A screenshot of the simulated mobility is provided in Fig. 7.4(b). In Fig. 7.5(a) and 7.5(b), we report the results for PRR with the mean (solid curve) and the standard deviation (shaded area) over the distance between the transmitters and receivers, in meters, and the PIR with mean and percentiles over distances up to 50 m, respectively. In Fig. 7.5(b), we also report the proportion P that the

Table 7.1 Simulation parameters (adapted from [36] ©2019 IEEE).

	Training Environment	**Evaluation Environments**	
		Scenario 1	**Scenario 2**
DOCA size	Straight highway section; lane width = 4 m		
	500 m, 1 lane/direction	1000 m, 2 lanes/direction	
Vehicle speeds	$\sim N(75,25)$ km/h	$\sim N(120,12)$ km/h $\sim N(120,72)$ km/h	$\sim N(120,72)$ km/h
Vehicle mobility	30 vehicles with wrap-around $\sim Exp(0.4)$[236]; constant speeds	Poisson arrival per direction with $\sim Exp(0.4)$ [236]; realistic SUMO mobility [247]; stop-and-go traffic in Scenario 2	
Message traffic	$S_{\mathrm{m}} = 190$ B, $T_{\mathrm{m}} = 100$ ms		
Resource pool	10 slots by 2 subchannels		
V2V Network and Channel Parameters of the Evaluation Environments [236]			
Network	10 MHz (50 RBs) bandwidth at $f_c = 5.9$ GHz with 32 RBs active, 1 subchannel = 16 RBs, 1 slot = 1 ms, MCS index = 9		
Antennae	1 Tx and 2 Rx omni-dir. with 3 dBi gain and 9 dB Rx noise figure Transmission power = $\{-5,23\}$ dBm; Thermal noise level = 174 dBm/Hz		
Path loss model	LOS in WINNER+B1 with antenna height = 1.5 m; path loss at 3 m is used for distance < 3 m		
Shadowing fading	Log-normal distributed with 3 dB std. dev.; decorrelation distance of 25 m		
Mode 4 Configuration Parameters			
$T_1 = 4$ ms, $T_2 = 14$ ms, $C_{\mathrm{resel}} \sim Unif[5,15]$, $P_{\mathrm{keep}} = 0$, $Thr_{\mathrm{sense}} = -120$ dBm			
RL Training Parameters			
$L_{\mathrm{epoch}} = 60$; actor and critic learning rates of $\alpha = 10^{-3}/(1+0.01 \times \#ep^{1.1})$			



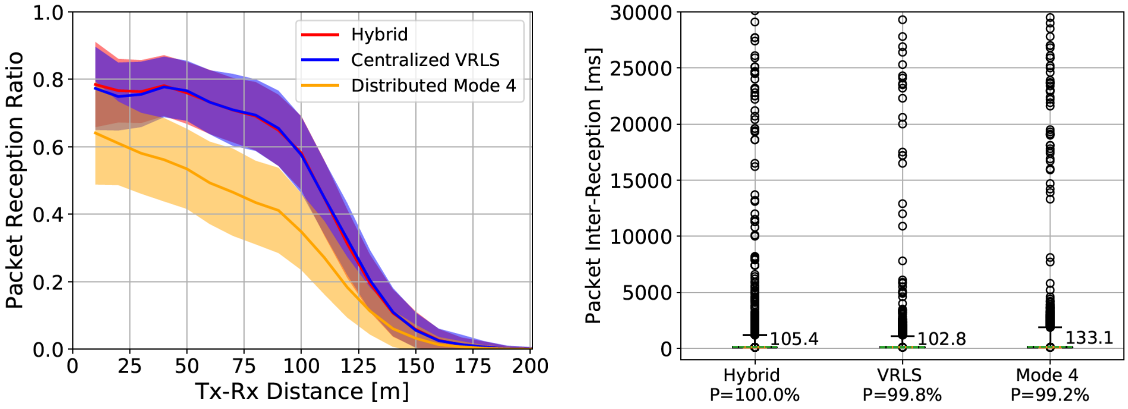
(a) Training environment.

(b) Evaluation environment Scenario 1.

(c) Evaluation environment Scenario 2.

Fig. 7.4 Screenshots of the training and evaluation environments, taken from the mobility simulator SUMO [247].

PIR results are based on. P is the ratio of number of PIR measurements used for each plot, normalized to the maximum of all algorithms. Note that PIR can only be measured when



(a) Mean (solid curve) and standard deviation (shaded region) of PRR vs Tx-Rx distance.

(b) Mean (denoted), 0.1^{st} and 99.9^{th} percentiles (whiskers), and outliers (rings) of PIR at Tx-Rx distances of $0-50$ m, with proportion P of number of PIR measurements used for each plot, relative to the algorithm with the maximum.

Fig. 7.5 Comparison of hybrid solution with the state-of-the-art in Scenario 1 [36] ©2019 IEEE.

more than a single reception from a specific transmitting vehicle takes place. Accordingly, the measurements do not include blackouts, i.e., single or no reception at all from a vehicle. Thus, P indicates the extent to which an algorithm experiences such blackouts (i.e., lower P indicates more frequent blackouts).

By observing Fig. 7.5(a), we can see that the performance of the hybrid and the centralized scheduler are similar and better than that of distributed mode 4 scheduler. This is in line with our previous results for VRLS in Chapter 6, where it was able to outperform mode 4 in the considered scenarios. Our results show that, under such conditions, the hybrid solution is also able to achieve a similar performance. Here, note that the resource recommendations of the hybrid agent are further combined with the sensing results. The policy developed by both VRLS and the hybrid solution is to divide the resources per direction dynamically depending on their load, while reusing them and avoiding collisions and HD errors as much as possible. We could observe that the gain of the two solutions compared to mode 4 increases with the distance between the transmitter and the receiver, maintaining a PRR degradation softer than mode 4 up to distances of around 100 m, after which the path loss effect of the channel dominates. Such performance is achieved by the similar policies learned by the two schedulers, which respect the global dynamics, hence avoiding the hidden-node problem, from which mode 4 algorithm suffers. Fig. 7.5(b) shows that the hybrid solution and VRLS achieve similar PIR performance, whereas mode 4 performs the worst with around 30%

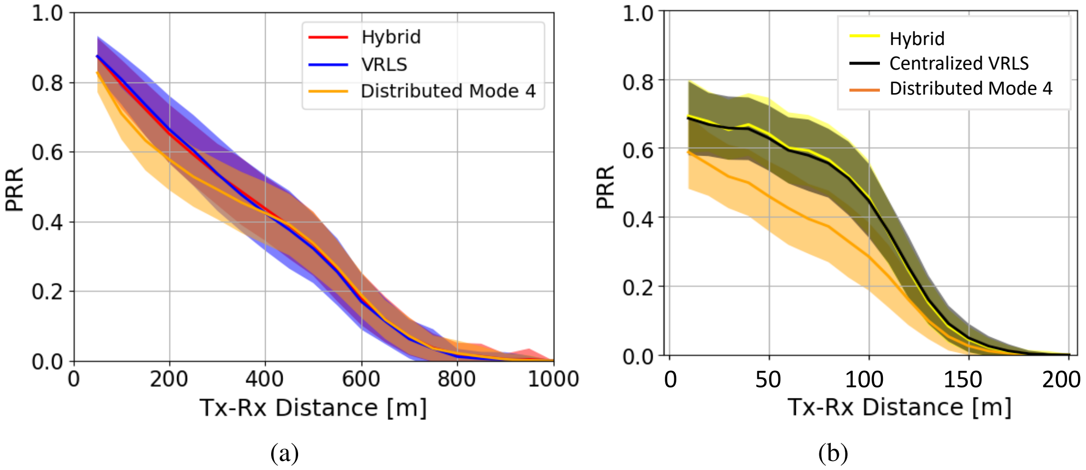
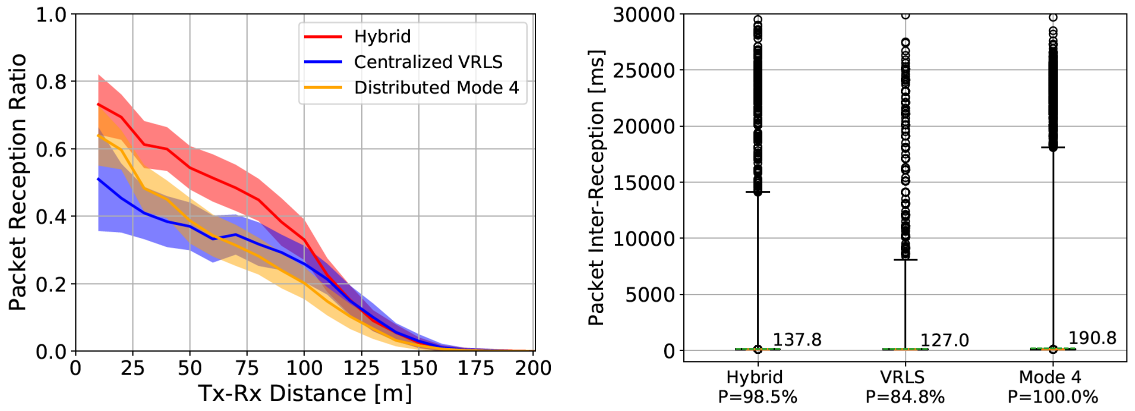


Fig. 7.6 Comparison of hybrid solution with the state-of-the-art in two variants of Scenario 1: a) transmit powers increased to 23 dBm; and b) speed deviations increased to 60%. Mean (solid curve) and standard deviation (shaded region) of PRR vs Tx-Rx distance is shown.

larger PIR, on average. Note that for all algorithms, all quartiles of PIR are at around 100 ms, which is equal to T_m, and with similar P values observed. The similarity indicates equitable comparison of the PIR measurements among the algorithms.

Next, we evaluate the performance of the algorithms in two variants of Scenario 1: a) with transmit powers increased to the maximum value of 23 dBm; and b) with speed deviations increased from 10% to 60%. We provide the PRR of the algorithms in Fig. 7.6. In variant (a), V2V transmissions achieve higher transmission rates at larger distances due to increased transmit power. However, the agents were not trained for such different channel behaviour. In the end, hybrid solution and VRLS perform very similar, while mode 4 showing a marginal difference to them, thanks to increased transmit powers enabling better sensing among vehicles. Increased transmit powers also benefit PIR, where all algorithms equally achieve a mean PIR of 100 ms. In the case of the scenario variant (b), higher speed deviation of vehicles result in lower performance of all algorithms, as compared to original Scenario 1. The PRR drops below unacceptable rates due to highly varying interference conditions in a loaded environment. Nevertheless, the hybrid solution exhibits a similar performance as VRLS, by outperforming mode 4.

In Scenario 2, we increase the speed deviations in the environment to 60%, as in Scenario 1-b, and yet further introduce an unexpected event in the DOCA environment, namely a stop-and-go traffic, e.g., due to a road accident. Vehicles in one direction are stopped at a specific location for 20 s that results in highly congested traffic and the so-called “accordion effect” (as can be seen in Fig. 7.4(c)), with the total number of vehicles in DOCA at a given



(a) Mean (solid curve) and standard deviation (shaded region) of PRR vs Tx-Rx distance.

(b) Mean (denoted), 0.1st and 99.9th percentiles (whiskers), and outliers (rings) of PIR at Tx-Rx distances of $0-50$ m, with proportion P of PIR measurements used for each plot, relative to the algorithm with the maximum.

Fig. 7.7 Comparison of hybrid solution with the state-of-the-art in the congested period of Scenario 2 [36] ©2019 IEEE.

time reaching up to around 80 (i.e., around 2.5 times the average number). The vehicles are being stopped for 20 s starting from the 400th s until the end of the simulation at 1000th s. We report the PRR of Scenario 2 in Fig. 7.7(a), where the measurements belong to the last 400 s of the simulation, hence representing the congested vehicular traffic. In Fig. 7.7(b), we provide the PIR with the value of P for each of the measurements from the same (congested) period of the simulation.

In Fig. 7.7, we first see that the overall *absolute* performance of all three algorithms is worse than Scenario 1, which is mainly caused by the highly increased traffic load and changing vehicular dynamics due to congestion. Similarly, the V2V resources become highly congested, leading to PRR below acceptable levels. Under such extreme conditions, as per the *relative* performance of the algorithms, Fig. 7.7(a) shows that the hybrid approach can improve the PRR with respect to both VRLS and mode 4, while VRLS performing occasionally worse than mode 4, especially at short ranges.

During the overloaded conditions in Scenario 2, the policy developed by VRLS is to “dump” all vehicles above the “capacity” of the resources into a single specific resource, where they all collide. This policy prevents the collisions on the rest of the resources by sacrificing one of them, hence creating a sort of an admission control mechanism. However, such a policy in case of queued-up vehicles that are in close proximity would result in persistent collisions of the transmissions using the same resource. On the other hand, in the case of a hybrid algorithm, vehicles are provided with a subset of recommended resources where they do not necessarily stick to a specific one that would create persistent collisions, but alternate between them, thus enabling diversity across the congested resources over time and space. The higher-level policy of assigning the resources according to load per direction still applies, eventually leading to a performance better than the other algorithms. Fig. 7.7(b) shows that PIR is larger than Scenario 1 for all algorithms, which is in line with their PRR performance. VRLS achieves the best performance, followed by the hybrid and mode 4 algorithms having around 10% and 50% longer PIRs on average, respectively. However, VRLS yields around 15% lower P compared to other algorithms, which indicates that VRLS suffers from blackouts more frequently, which is a phenomenon that is not reflected by the PIR measurements. Although the hybrid solution also yields slightly more persistent collisions compared to mode 4, it achieves considerably lower mean PIR, which is largely due to the much higher PRR it provides.

7.4 Conclusions

Motivated by the individual challenges that distributed and centralized schedulers face in reality, and considering whether a combined approach can provide any performance gains as compared to either of the approaches, we proposed a novel hybrid approach for scheduling V2V communications that combines our centralized RL-based approach with the distributed sensing-based mechanism. The proposed hybrid method incorporates long-term recommendations from the RL agent on utilizing the resources, followed by the vehicles selecting their final resources taking dynamical and local sensing information into account. Consequently, the method combines the best of both worlds: i) the global view (based on load, direction, etc.) and trained policy of the RL agent, and ii) the local view and the ability of sensing on adapting to hard-to-predict events.

The benefits of our hybrid solution come into play in the case of highly dynamic and overloaded scenarios with unexpected conditions differing significantly from those experienced during the training of the RL agent, such as a road congestion due to an accident. Under such conditions, the hybrid solution achieves higher communication reliability compared to the centralized RL or the distributed sensing scheduler alone, otherwise delivering similar performance. Given such performance, the hybrid solution can substitute VRLS whenever such unforeseen conditions are detected in the environment.

Nevertheless, training of the hybrid RL agent comes with the cost of longer training times, i.e., an order of magnitude that of the VRLS agent. Further, as a solution that involves a distributed component, the performance of the algorithm is also subject to vulnerabilities of the distributed resource allocation mechanism. To illustrate, selecting a higher percentage of resources (i.e., greater than 20% that we considered) from the recommended set for distributed sensing yielded worse performance, which we have not included in our evaluation results. Similarly, we have also observed that the frequency of resource re-selection also had an impact on the performance. Hence, the proposed technique can further benefit from the enhancements to the distributed resource allocation mechanisms proposed in the literature (cf. Section 2.4), as well as by different strategies to combine the centrally allocated resources with the sensing results, e.g., via weighting techniques, which however require further exploration and careful tuning of the algorithmic parameters depending on the deployed setting.

Chapter 8

iVRLS: In-coverage Vehicular Reinforcement Learning Scheduler

8.1 Motivation and Contribution

In the previous chapters, we focused on expected, delimited out-of-coverage areas, and proposed centralized scheduling of V2V communications utilizing techniques based on RL. In this chapter, we extend our centralized RL-based approach to the conventional in-coverage scheduling problem. As introduced in Section 2.3.3, for centralized scheduling of V2V communications under coverage, current solutions in the literature propose heuristics based on the location information of vehicles to enable spatial resource reuse (cf. [32], [137], [141], [142]). However, these algorithms typically assume ideal coverage conditions, without any loss of the control signaling between the base station (BS) and vehicles, which is used to request and assign the resources. Furthermore, the algorithms rely on high-frequency, dynamic scheduling updates, which actually increase their dependency on the control signaling reliability, besides resulting in high control signaling overhead. Therefore, more efficient algorithms that can operate at least equally well under realistic, intermittent coverage conditions would be beneficial.

In this chapter, we consider two ways to meet the design of such an improved solution. As a first enhancement, we propose extending the longevity of the resource assignments under coverage interruptions. Specifically, as mentioned above, legacy methods for centralized resource allocation are not designed to cope with failures in the links between the vehicles and the scheduling entity (e.g., a BS). As described in Section 2.4.1, upon detecting any link failure, vehicles *release* their allocated resource, thus delaying or dropping their V2V

Parts of the results and Section 8.2 in Chapter 8 have been published in [38] ©2021 IEEE.

transmissions. If they support the distributed resource allocation mode, they would resort to a random selection of the resources from the configured resource pool under such exceptional cases, and switch to the distributed resource allocation mode once the sensing results become available. In this chapter, we rather consider an enhancement to the centralized scheduling itself, and propose that, instead of deferring from transmissions, vehicles continue using their allocated resources even if they detect any link failures to the scheduling entity, for a (pre-)defined duration, e.g., until connecting back to the scheduler. This would mitigate the degradation of V2V communication quality due to otherwise dropped or delayed transmissions, or due to vehicles switching to autonomous resource allocation methods, which suffer from inefficient resource utilization.

On top of this enhancement, we then propose *iVRLS (in-coverage vehicular reinforcement learning scheduler)*, a centralized RL-based solution similar to VRLS we presented in Chapter 6. While VRLS is designed to *pre-schedule* resources for well-known out-of-coverage parts of the route (e.g., a tunnel) via one-time scheduling assignments before vehicles leave the coverage, iVRLS takes the advantage of resource assignments that are possible at all times under coverage, and makes use of the instantaneous and exact knowledge of vehicular mobility and data traffic. In particular, the RL model assigns each resource based on the estimation of the current and future instances of interference conditions among the vehicles until the next assignment, by taking their current resource allocation and velocity into account. Whereas the next assignment might be delayed depending on any (expected or unexpected) coverage loss that might occur.

For our performance analysis, we select the state-of-the-art centralized scheduling algorithm MRD in [32] to serve as a baseline, due to its realistic and practical assumptions, as well as its benchmarked performance, as discussed in Section 2.3.3. We further extend the MRD operation with the enhancement proposed above, for a fair comparison with iVRLS. In our evaluations, we first consider ideal coverage conditions, and show that, when the frequency of the resource allocation is reduced, the performance of the centralized scheduling algorithms degrades, thus creating a similar effect as coverage interruptions. We then extend our evaluations to consider more practical scenarios with realistic coverage assumptions with varying mobility and traffic load. In comparison to MRD, iVRLS achieves better performance under lossy coverage conditions and a relatively low frequency of scheduling updates, thus yielding less signaling overhead to achieve the same reliability targets. We also evaluate the case where the proposed enhancement to centralized scheduling is not applied, i.e., vehicles release their resources upon coverage interruptions. Such operation results in a considerable amount of degradation and variance of V2V message reliability for both algorithms even with the highest possible frequency of scheduling updates and a low traffic load.

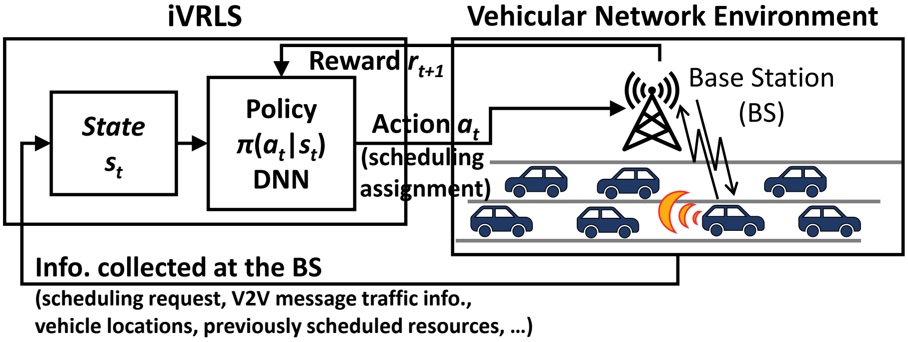

iVRLS
Vehicular Network Environment
Reward r_{t+1}
State s_t
Policy $\pi(a_t|s_t)$ DNN
Action a_t (scheduling assignment)
Base Station (BS)
Info. collected at the BS
(scheduling request, V2V message traffic info., vehicle locations, previously scheduled resources, …)

Fig. 8.1 In-coverage Vehicular Reinforcement Learning Scheduler (iVRLS) (adapted from [38] ©2021 IEEE).

Finally, we evaluate the performance of the algorithms in specifically designed scenarios that vary in terms of the deployment of BSs and tunnels (i.e., known out-of-coverage areas), accounting for different coverage conditions in the network. While MRD, which is designed with a perfect coverage assumption in mind, can better benefit from relatively good coverage conditions; in case of coverage losses iVRLS yields marginally larger V2V distances at which a given target PRR is achievable. Overall, we demonstrate that iVRLS offers a unified, versatile solution for deployment irrespective of coverage, enabling simplified implementation and robustness to coverage variations in the network.

In the rest of this chapter, Section 8.2 presents iVRLS, Section 8.3 provides our evaluation results, and Section 8.4 draws the conclusions.

8.2 iVRLS: In-coverage Vehicular Reinforcement Learning Scheduler

iVRLS targets the centralized scheduling problem using the framework of RL, as depicted in Fig. 8.1. Our design of iVRLS is based on that of VRLS (cf. Section 6.2), with the main differences being the definition of the state information of the environment, and the possibility of taking actions anytime and anywhere within the coverage of the BSs.

8.2.1 State Representation

State representation s_t, at the instance t a vehicle is requesting a resource, contains the information collected from the vehicular network environment in a compact and useful way. We design s_t to indicate expected "interference" level I_r on each resource r in the configured V2V resource pool, in case it was assigned to the requesting vehicle. We represent I_r by the inverse of the distance between the vehicles using the same resource and the requesting vehicle, over current and future instances of their message transmissions. Specifically, $s_t = [I_{r=1}, I_{r=2}, ..., I_{r=R}]$ is a vector with number of elements equal to the number of resources R in the resource pool, with each element $I_r = I_r^{i=1} + I_r^{i=2} + I_r^{i=3} + ...$ representing the sum of expected "interference" coming from the set of vehicles $i = \{1, 2, 3, ...\}$ using the same resource r. In case no vehicle using a given resource, $I_r = 0$. Otherwise, I_r^i for a single vehicle i using the resource r is calculated as:

$$\begin{aligned} I_r^i =& \frac{1}{N_{\mathrm{SR}}}\left(\frac{1}{|\Delta x|} + \frac{1}{|\Delta x + \Delta v T_{\mathrm{m}}|} + \frac{1}{|\Delta x + \Delta v 2 T_{\mathrm{m}}|} + ... + \frac{1}{|\Delta x + \Delta v (N_{\mathrm{SR}} - 1) T_{\mathrm{m}}|}\right), \quad (8.1) \\ =& \frac{1}{N_{\mathrm{SR}}} \sum_{n=1}^{N_{\mathrm{SR}}} \frac{1}{|\Delta x + \Delta v (n-1) T_{\mathrm{m}}|}, \quad (8.2) \end{aligned}$$

where $\Delta x = x_i - x_{\mathrm{req}}$ and $\Delta v = v_i - v_{\mathrm{req}}$ are the relative distance and the speed between vehicle i and the requesting vehicle, respectively. Therefore, the first term in the parentheses represents the "current interference", while every other term indicating the "expected interference" in the future, based on the changing positions of vehicles over time with message generation periodicity T_{m}, until the next scheduling event. Further, in case of a known out-of-coverage area, such as due to a tunnel, the time to next scheduling event is calculated by dividing the length of the tunnel to the average speed of vehicles in the environment. An average over current and future instances is then taken to represent "overall" interference. In order to avoid singularity, we further modify the denominator in the sum as:

$$I_r^i = \frac{1}{N_{\mathrm{SR}}} \sum_{n=1}^{N_{\mathrm{SR}}} \frac{1}{\max(\Delta x_{\mathrm{min}}, |\Delta x + \Delta v (n-1) T_{\mathrm{m}}|)}, \quad (8.3)$$

where $\Delta x_{\mathrm{min}} > 0$ is a fixed, minimum inter-vehicle distance. Overall, higher value of I_r in the state representation indicates higher expected "interference" on a resource, in case assigned to the requesting vehicle.

8.2.2 Action Definition

Same as VRLS, action a_t of iVRLS denotes assigning a single resource r to the vehicle requesting at instance t, from the resource pool configured for V2V communications. iVRLS selects the resource to be assigned based on its policy $\pi(a_t|s_t) \rightarrow [0,1]_R$, which is a mapping from state of the vehicular environment to a probability distribution over the set of possible actions, i.e., R resources in the resource pool. The resource is then selected at random according to the distribution. In case more than one vehicle request a resource at the same time, the actions are taken in a random order.

8.2.3 Reward, Policy Deep Neural Network, and the Training Algorithm

In order to train iVRLS, we use the same reward definition as that of VRLS, reflecting the reliability of V2V transmissions in the environment. Namely, $r_{t+1} = -10 \times (1 - \mathrm{PRR})$, where PRR is measured for the range of interest, e.g., a certain communication range required by a V2V use case. In case no transmission take place between the actions, e.g., when two vehicles request a resource at the same time, the reward of the previous action is provided.

Given the large number of possible state and action combinations as in the case of VRLS, we represent the policy of iVRLS with a deep neural network (DNN). DNN architecture is the same as that of VRLS, yet only using a single 1D convolution layer at the input layer, as state representation of iVRLS consists of a single vector. iVRLS is also trained using the state-of-the-art A3C RL algorithm we extended (cf. Algorithm 3), as VRLS, using the same parameters (cf. Section 6.3.1).

We have experimented with several other design options for iVRLS, in terms of different state representations and DNN architectures, besides the design provided here. In Appendix B, we present other design options we considered and a comparison of their learning performance. Our results show that the iVRLS design provided in this chapter offers the best performance among the considered options.

8.3 Evaluation

In our evaluations, we utilize the system model described in Section 3.1 with the parameter values summarized in Table 8.1, and using the following additional assumptions for the in-coverage scheduling. Each vehicle sends SR with periodicity $T_{\mathrm{SR}} = N_{\mathrm{SR}} T_{\mathrm{m}}$ (in multiples of message generation frequency, T_{m}), starting from its first message generation after connecting to the BS. The vehicle then keeps using the same scheduled resource for its V2V transmissions within a period, i.e., until the next SR/SA. Mobility information of vehicles is collected by BS on demand; specifically, at every scheduling instance, which can be acquired via different positioning methods such as based on radio-signaling or global positioning system [292].

Vehicles generate V2V messages of size $B_{\mathrm{m}} = 190$ Bytes with periodicity T_{m}, which we vary in the evaluations. Configured V2V resource pool is assumed to consist of 2 subchannels and 10 time slots, each able to carry a single V2V message combined with

Table 8.1 Simulation parameters (adapted from [38] ©2021 IEEE).

V2V message traffic	$B_\mathrm{m} = 190$ Bytes; T_m: variable.
V2V resources	Carrier frequency = 5.9 GHz; Bandwidth = 10 MHz (50 RBs) with 32 RBs active; MCS index = 9; 1 subchannel = 16 RBs, 1 slot = 1 ms; Resource pool of 2 subchannels by 10 slots, periodically repeating with 80 ms.
V2V channel model	3GPP Channel Model [236] with Path loss: LOS model in WINNER+B1; path loss at 3 m is used for distances < 3 m; Shadowing fading: log-normal distr. with 3 dB std. dev. and 25 m decorr. distance. V2V Tx power = $\{-5, 23\}$ dBm (scenario dependent); Thermal noise level = –174 dBm/Hz; Antennae: 1 Tx and 2 Rx omni-directional with 1.5 m height, 3 dBi gain, and 9 dB noise figure.
UL/DL channel model	COST-Hata Channel Model [240] at 2.1 GHz with 10 dB shadowing fading, 10 MHz (50 RBs) bandwidth; BS DL Tx power = 30 dBm, vehicle UL Tx power = 23 dBm BS antenna: isotropic with 30 m height and 5 dB noise figure.

the control information and protocol overhead. V2V and UL/DL channels are simulated using realistic path loss and fading with parameters indicated in Table 8.1. In the state representation of iVRLS, we set $\Delta x_\mathrm{min} = 3$ m, in accordance with the V2V path loss model used in our evaluations.

8.3.1 Training of iVRLS

We train iVRLS before its deployment, in a setting with simpler mobility and communication model as compared to evaluation scenarios, which enables faster training, as described in Section 3.1.6. In the training environment, 50 vehicles are initially placed uniformly at random on a 1000-m-long two-way highway without any tunnel, and travel with constant speeds randomly selected from normal distribution $\sim\mathcal{N}(75, 25)$ km/h. Upon their exit, they return back to the highway from the opposite direction after a time offset $\sim\mathrm{Exp}(0.4)$ to create an average time gap of 2.5 s between vehicles. V2V transmission range is assumed to be of 120 m. Vehicle-to-BS links in the training are assumed to be error-free. Each vehicle sends a scheduling request every $T_\mathrm{SR} = 1000$ ms with $T_\mathrm{m} = 100$ ms (i.e., $N_\mathrm{SR} = 10$). The reward r_{t+1} is calculated using PRR measured at $0 - 100$ m Tx-Rx distance, as described in Section 3.2. We provide the training curve of iVRLS in Fig. 8.2. It shows the average reward collected by iVRLS over the training epochs (each epoch is a sequence of 60 state-action-

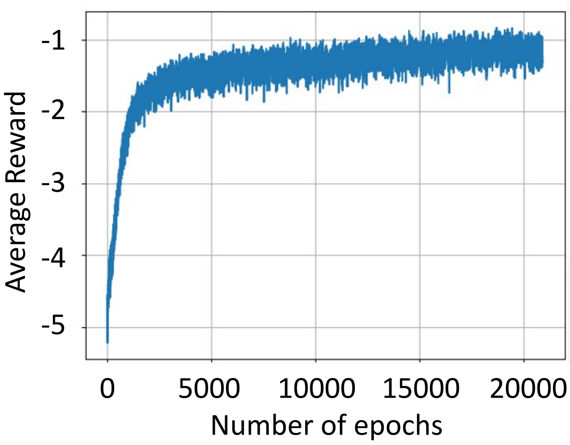


Fig. 8.2 Learning curve of iVRLS [38] ©2021 IEEE.

reward tuples). iVRLS converged to a stable performance level at around 20000 epochs in the training environment.

Besides the training methodology presented here, we further evaluated the impact of learning rate, as well as the frequency of scheduling requests on the training performance, which we present the results in Appendix B. Our evaluations led us to consider the respective values for the training parameters presented here.

8.3.2 Performance Under Ideal Coverage

We first evaluate the performance of in-coverage scheduling under ideal coverage conditions, namely assuming no errors on the UL/DL signaling between the BS and the vehicles. We evaluate the reliability performance of iVRLS, in comparison to MRD and VRLS, in the environment E2-L with a highway length of 500 m as described in Section 6.3.4 (assuming perfect coverage instead of a DOCA). To make a fair comparison, we have extended VRLS for the in-coverage operation. Specifically, instead of the one-time assignments we considered in the case of a DOCA, we consider that VRLS does scheduling whenever a vehicle sends a scheduling request (as done by iVRLS and MRD). Similarly, we have extended the state representation of VRLS to consider information from both in front of and behind the vehicles when calculating Δx and C, as opposed to considering only in front of vehicles (that was calculated only when they enter DOCA).

We report the PRR of the algorithms as a function of distance in Fig. 8.3, for different periods of scheduling requests T_{SR} sent by the vehicles, assuming a message generation period of $T_{\mathrm{m}} = 100$ ms. In order to provide insight into the practical implications of these results, Table 8.2 shows the reliable communication range under different scheduling rates.

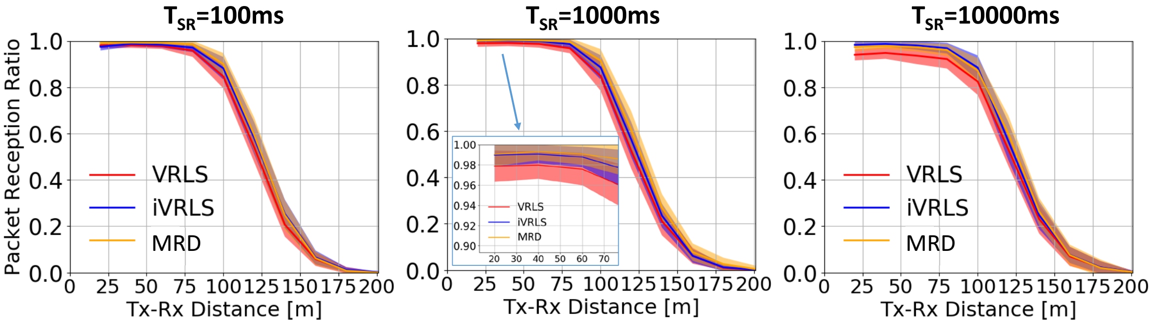


Fig. 8.3 PRR as a function of Tx-Rx distance in environment E2-L (cf. Section 6.3.4) with a 500 m highway length under ideal network coverage, for different scheduling request periods T_{SR} with V2V message generation period of $T_{\mathrm{m}} = 100$ ms.

Table 8.2 Reliable communication range as a function of desired PRR, for different scheduling periodicities.

Scheduling Period	**80% PRR**			**90% PRR**			**95% PRR**		
$T_{\mathbf{SR}}$	iVRLS	MRD	VRLS	iVRLS	MRD	VRLS	iVRLS	MRD	VRLS
100 ms	106 m	108 m	104 m	96 m	100 m	90 m	84 m	88 m	81 m
1000 ms	103 m	107 m	105 m	95 m	100 m	89 m	83 m	86 m	75 m
10000 ms	106 m	106 m	102 m	96 m	93 m	85 m	85 m	83 m	39 m

The range is calculated as the maximum distance at which the mean PRR is above or equal to 80%, 90%, and 95%, considering requirements of different V2V applications (cf. Section 2.2).

Our first observation is that, by comparing Fig. 8.3 and Fig. 6.5(e) in Chapter 6, given the same amount of resources and traffic, the performance of centralized scheduling under coverage is much higher in terms of reliability, as compared to a DOCA having the same vehicular mobility and network conditions. Typically, around 10% larger PRR is achievable within 100 m range with $T_{\mathrm{SR}} = 100$ ms, as compared to that of the DOCA. Similarly, we observe that, by comparing Table 6.6 and Table 8.2, as the coverage conditions get ideal, fewer resources become necessary to support the target reliability of V2V applications.

Secondly, given the same environment and amount of resources available, we observe that the reliable communication range decreases with the lower rate of scheduling updates, in the case of all algorithms. Considering the dynamic vehicular environment, more frequent scheduling updates are required to maintain higher reliability of V2V messages, however, which comes with the expense of increased UL/DL signaling.

Third, MRD and iVRLS achieve very close performance, better than that of VRLS, as they are specifically designed for in-coverage scheduling. VRLS, which is extended for in-coverage operation, performs the worst among the algorithms, for all different rates of scheduling updates.

8.3.3 Performance Under Realistic Coverage with Varying Mobility and Traffic Load

Next, we evaluate the performance of iVRLS in comparison to MRD considering non-ideal network connectivity. Namely, we consider errors on the UL/DL signaling between the BS and the vehicles according to the model described in Section 3.1.2. We are interested in how varying conditions of V2V traffic load and vehicular mobility impact the performance of the schedulers under non-ideal coverage conditions. In terms of mobility, we consider two scenarios: i) the environment E2-L with 1000 m length of a highway, as described in Section 6.3.4; and ii) the stop-and-go traffic as described in Scenario 2 in Section 7.3, in the same environment. For both scenarios, we evaluate the PRR achieved by the algorithms as a function of distance, under different periods of V2V message generation T_{m} and scheduling requests T_{SR}. We report the results in Fig. 8.4 and Fig. 8.5, respectively for the two mobility scenarios. From the V2V applications' point of view, the benefit of a better scheduler can be seen as an increase in the effective communication range, given a certain message delivery probability requirement. In Table 8.3, we provide the reliable communication range as a function of the desired delivery ratio, under different traffic loads varied by the V2V message generation periodicity T_{m} and/or stop-and-go traffic (S&G), with $T_{\mathrm{SR}} = 10000$.

We again first observe from the results in Figures 8.4 and 8.5 that, centralized in-coverage scheduling can make much more efficient use of V2V resources, even under non-ideal coverage assumptions, as compared to out-of-coverage scheduling, when we compare our results to those in Chapter 6 and Chapter 7. Specifically, under the same amount of resources and traffic demand, reliable communication distance at which 80% PRR can be achieved is increased by around 3.5 times.

Second, given the same rate of scheduling updates, the reliability performance of both algorithms degrades with the increased traffic load, as expected, either due to an increased V2V message generation rate or a larger number of vehicles due to stop-and-go traffic. The degradation is much more pronounceable as compared to ideal coverage assumptions, according to our results in Section 8.3.2. Specifically, when comparing the plots in Fig. 8.4, for a scheduling update rate with $T_{\mathrm{SR}} = 10000$ ms, a five times increase in V2V message generation frequency decreases the reliable communication distance at which 80% PRR is achievable by around 16%. Similarly, by comparing the plots in Fig. 8.4 and Fig. 8.5, for $T_{\mathrm{SR}} = 10000$ ms and $T_{\mathrm{SR}} = 10000$ ms, stop-and-go traffic yields around 25% shorter

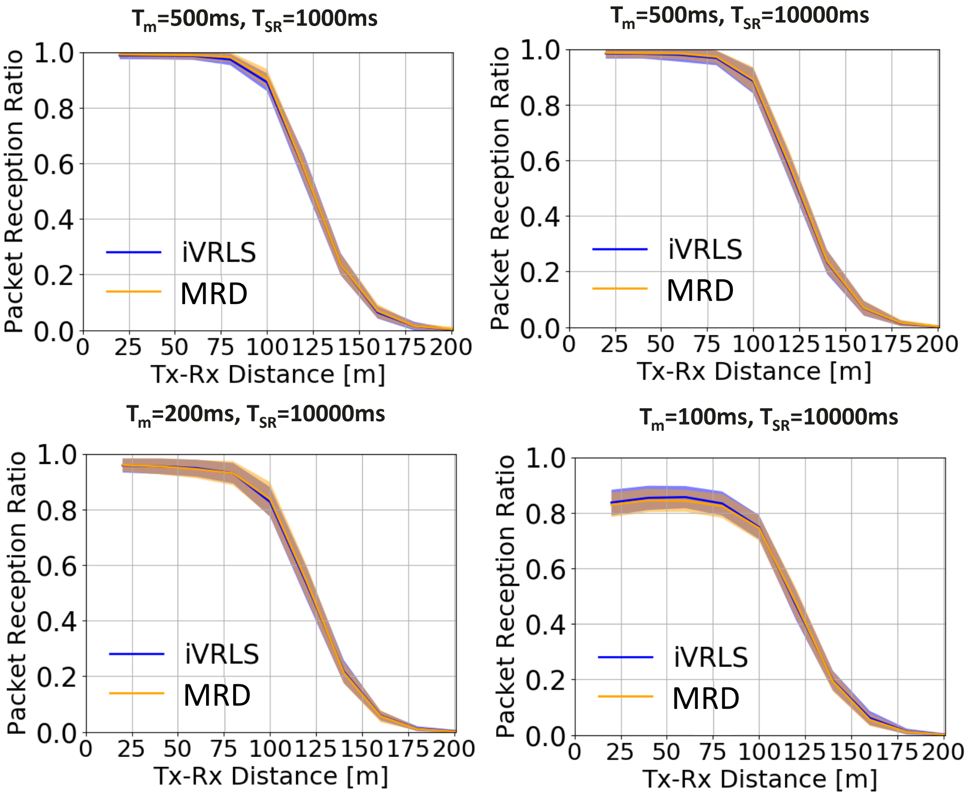


Fig. 8.4 PRR in E2-L (1000 m) for different message generation periods T_m and scheduling request periods T_SR.

Table 8.3 Reliable communication range as a function of desired PRR, for different traffic loads.

V2V traffic load	**80% PRR**		**90% PRR**		**95% PRR**	
(T_m)	iVRLS	MRD	iVRLS	MRD	iVRLS	MRD
500 ms	106 m	106 m	95 m	97 m	84 m	85 m
200 ms	102 m	103 m	85 m	86 m	60 m	50 m
100 ms	89 m	87 m	-	-	-	-
200 ms (S&G)	80 m	77 m	-	-	-	-
100 ms (S&G)	52 m	48 m	-	-	-	-

reliable communication distance. iVRLS and MRD achieve similar performance in the case of relatively low traffic load. However, under high load, iVRLS can make more efficient use of the resources by delivering marginally better reliability while requiring a lower rate of scheduling updates.

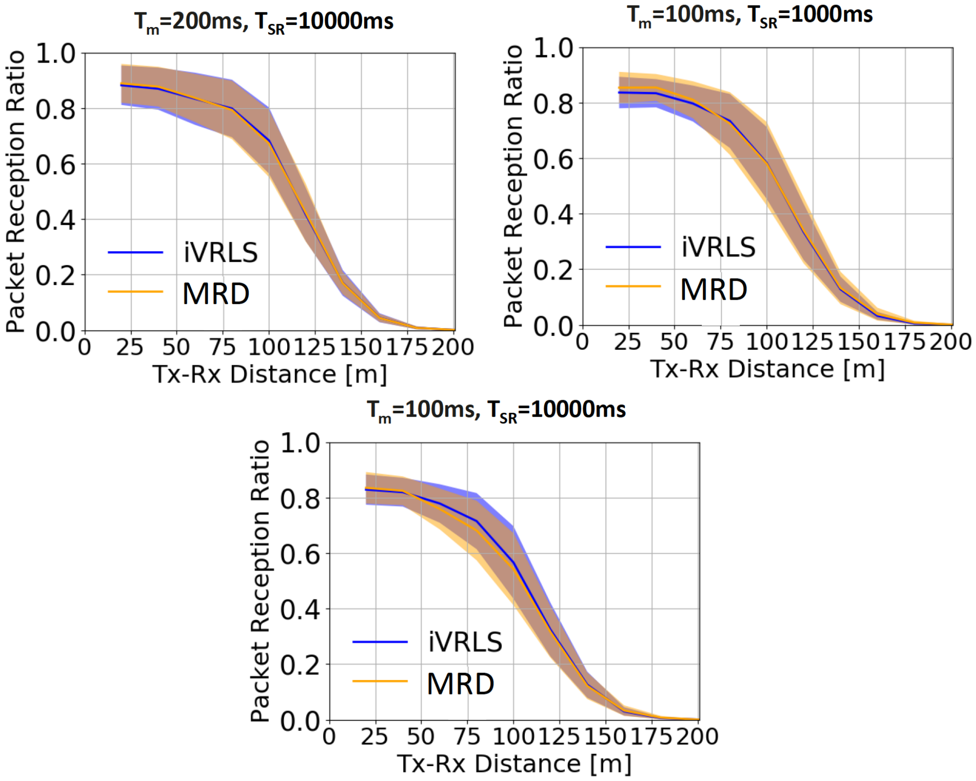


Fig. 8.5 PRR in E2-L (1000m) with stop-and-go traffic for different message generation periods T_{m} and scheduling request periods T_{SR}.

8.3.4 Performance without the Proposed Enhancement for Centralized Scheduling to Support Intermittent Coverage

As discussed in Section 8.1, we propose a simple enhancement to centralized V2V scheduling to account for intermittent network coverage loss, which is in general applicable to any existing solution: upon experiencing the coverage loss, vehicles keep using their latest resource assignments for V2V communications until they connect back to the scheduler and request a new resource. In our evaluations in the previous subsections, we enabled this enhancement to the MRD algorithm, in order to allow a fair comparison with iVRLS.

In this subsection, we demonstrate the impact of our proposed enhancement by evaluating the PRR performance of iVRLS and MRD *without* our proposed enhancement, considering the environment E2-L with 1000 m length of highway. Namely, vehicles *release* their resource allocation in case of the loss of the control signaling between the BSs and the vehicles, used for sending the scheduling requests and scheduling assignments. We provide the results in Fig. 8.6. As observed from Fig. 8.6, average PRR of both iVRLS and MRD decreases well below to desired rate of 80% within $0-100$ m range, and exhibits a large variance,

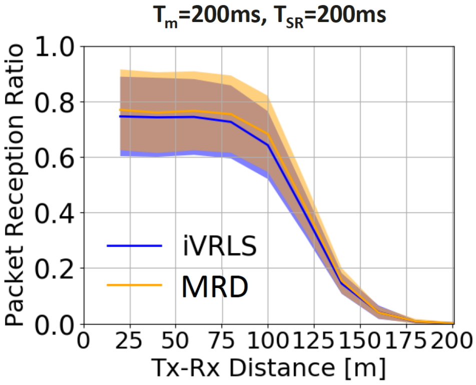
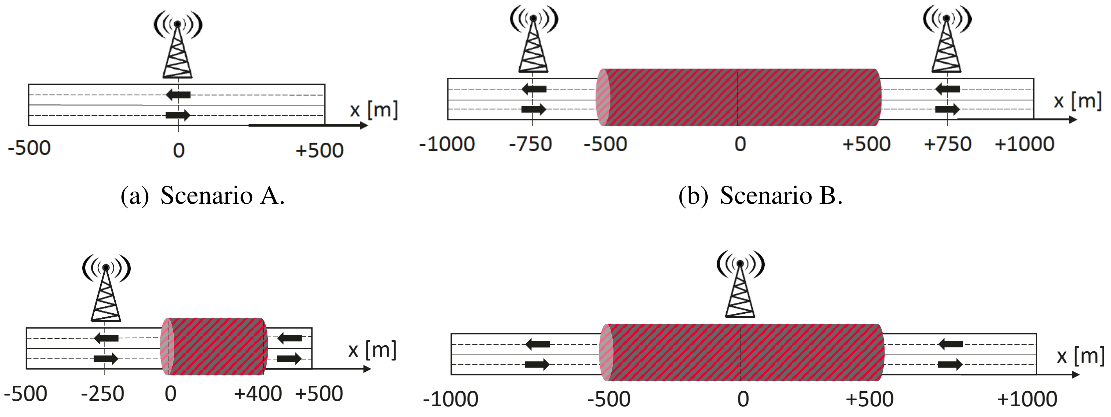


Fig. 8.6 PRR in E2-L (1000m) where vehicles do not keep their existing resources in case the control signaling between the BS and the vehicles is lost.

even under relatively low traffic load with the highest scheduling frequency as compared to evaluations reported in Fig. 8.4. Without our proposed enhancement to centralized scheduling under intermittent coverage, vehicles defer their resource allocations and can not transmit their generated messages, which degrades the reliability of V2V communications.

8.3.5 Performance under Varying Coverage Conditions

In this subsection, we evaluate the performance of iVRLS in specifically designed realistic scenarios that are representative of different network coverage characteristics. We consider four scenarios, denoted A, B, C, and D, as illustrated in Fig. 8.7. All scenarios consist of a two-way straight section of a highway, with 2 lanes per direction. In Scenario A, a single BS serves a 1000 m-long area of road, with scheduling updates every 10 s (i.e., $T_{\mathrm{SR}} = 10000$ ms). We set V2V transmit powers to a low value of -5 dBm, to enable multiple resource reuses within the considered area. Vehicles travel with speeds distributed normally $\sim\mathcal{N}(75,45)$ km/h. In Scenario B, smaller areas are served by BSs, with scheduling updates every 1 s. Vehicles are assumed to travel with speeds distributed normally $\sim\mathcal{N}(120,36)$ km/h. Thus, sparser and less varying road traffic, combined with more frequent scheduling offers better coverage conditions as compared to Scenario A. Yet, Scenario B has a tunnel zone in its central part, where no coverage available at all (vehicles can not send/receive any SR/SA). On the other hand, V2V powers are set to a more realistic value, which is the allowed maximum of 23 dBm. Scenario C is a variant of Scenario A, having a tunnel of 400-m length extending from its mid-way to its east end. Scenario D is a variant of Scenario B, yet having only a single BS deployed, which is located close to the center of the road (outside the tunnel), and vehicles are assumed to transmit with powers of -5 dBm as in Scenario A and Scenario C. Common to all scenarios, vehicles transmit broadcast V2V messages with periodicity $T_{\mathrm{m}} = 200$ ms.



(a) Scenario A.

(b) Scenario B.

(c) Scenario C.

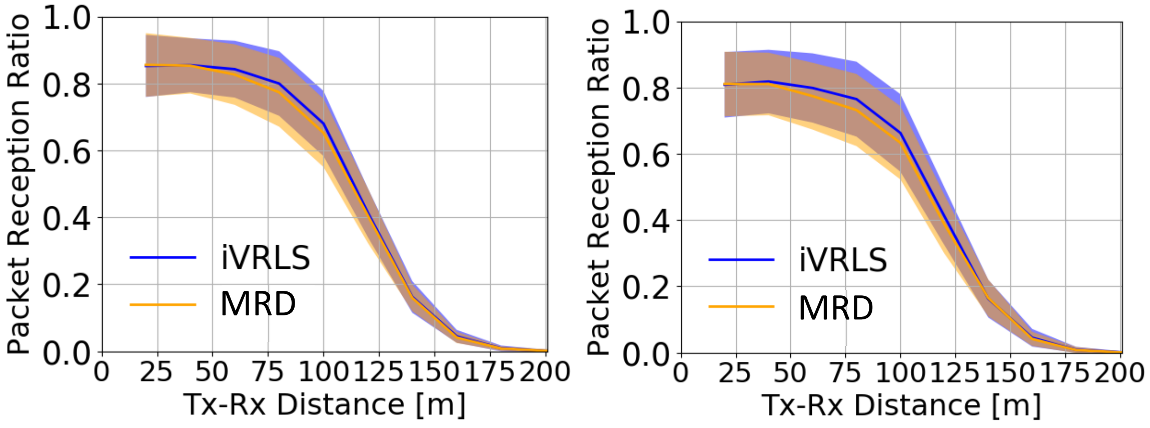
(d) Scenario D.

Fig. 8.7 Evaluation scenarios having different coverage conditions ((a) and (b) from [38] ©2021 IEEE).

BSs are deployed at the places as indicated in Fig. 8.7, with a 45 m longitudinal offset from the center of the road.

We first present the performance of algorithms in Scenario A, separately at the central section ($[-250,+250]$ m) and at the edge sections ($[-500,-250]$ and $[+250,+500]$ m) of the area, in Fig. 8.8. In the results, we report the PRR with mean and standard deviation, measured as a function of transmitter-to-receiver distance. We observe that iVRLS performs marginally better than MRD at all sections of the road, with a gain more pronounced at larger transmitter-receiver distances. In case of edge sections of the road, vehicles suffer from larger path loss to the BS, resulting in increased loss of scheduling requests and assignments, which degrades the performance of both algorithms. In that case, iVRLS shows a larger gain over MRD as compared to the central section of the road. Specifically, given a PRR target of 80%, iVRLS can provide around 14% larger V2V communication range in areas close to the BS, and 35% larger range at the edge of the coverage, in the considered scenario.

Next, in Fig. 8.9, we provide the results for Scenario B, from the two ends of the road under (partial) coverage ($[-1000,-250]$ and $[+250,+1000]$ m), and from the very central part of the tunnel ($[-250,+250]$ m). Overall, compared to Scenario A, Scenario B has less traffic load, accompanied with higher scheduling frequency, which results in overall higher reliability of V2V transmissions. Also, higher transmit power increases the rate of successful transmissions at larger transmitter-receiver distances, up to 800 m. Such ideal conditions benefit the performance of MRD under the coverage, for which it is designed. It provides marginally larger reliability of V2V transmissions in coverage. At close transmitter-receiver



(a) Central section.

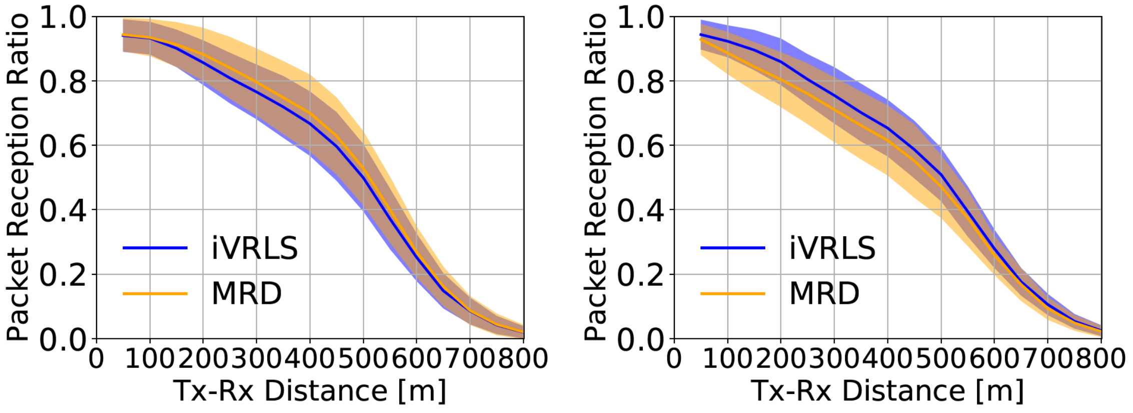
(b) Edge sections.

Fig. 8.8 PRR in Scenario A, measured as a function of Tx-Rx distance, at the central and the edge sections of the road [38] ©2021 IEEE.

ranges, both iVRLS and MRD yield around 95% PRR in average, with values achievable up to 100%. On the other hand, transmissions within the tunnel section of the road suffer from degraded reliability. Under such conditions, iVRLS is able to yield higher performance than MRD at almost all transmitter-receiver distances. Specifically, iVRLS can increase the reliable communication range within the tunnel by 66% and 24% for target PRR requirements of 90% and 80%, respectively.

In the tunnel section of Scenario B, we further evaluate the performance of other alternative schedulers for out-of-coverage V2V communications. Without our proposed enhancement where vehicles keep using their scheduled resources beyond the coverage, one has to combine such schedulers with the centralized in-coverage schedulers. In Fig. 8.10, we provide the performance of the distributed scheduler mode 4 and our centralized solution VRLS designed for out-of-coverage communications, besides the iVRLS. Fig. 8.10 shows that, with our proposed method, extended centralized schedulers perform better than the distributed scheduler mode 4 in this scenario. On the other hand, VRLS, designed solely for out-of-coverage operation, outperforms both of the extended schedulers, as well as mode 4. Accordingly, one could consider combining in-coverage solutions with VRLS for even better reliability outside the network coverage, yet at the expense of increased cost and complexity.

In Fig. 8.11, we report the results for Scenario C, from the west section ($[-500,0]$ m) and the tunnel section ($[0,+400]$ m) of the road. We observe from Fig. 8.11 that the performance of both algorithms degrade in the case of the tunnel, similar to the case of the edge sections in Scenario A, where vehicles suffer from intermittent coverage. Similar to our results in the



(a) In-coverage sections.

(b) Tunnel section.

Fig. 8.9 PRR in Scenario B, measured as a function of Tx-Rx distance, at the in-coverage and the tunnel sections of the road [38] ©2021 IEEE.

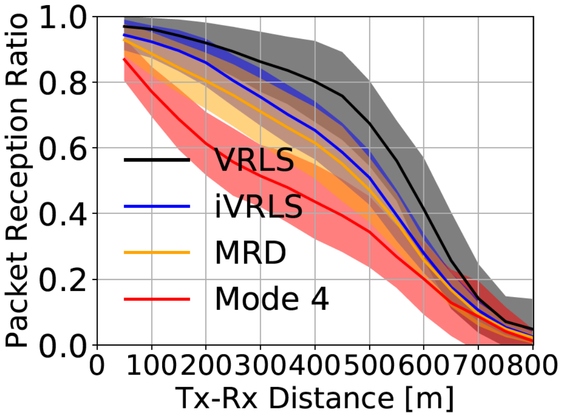
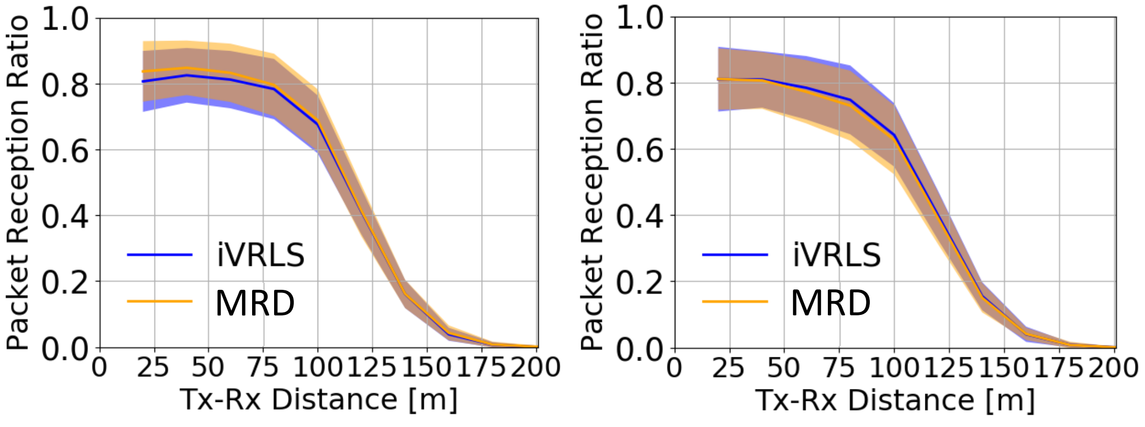


Fig. 8.10 PRR of algorithms in the tunnel section of Scenario B, measured as a function of Tx-Rx distance [38] ©2021 IEEE.

previous scenarios, iVRLS yields marginally better PRR than MRD in the case of the tunnel area, while the BS serving a small area of 500 m length yields good coverage conditions that benefit the performance of MRD.

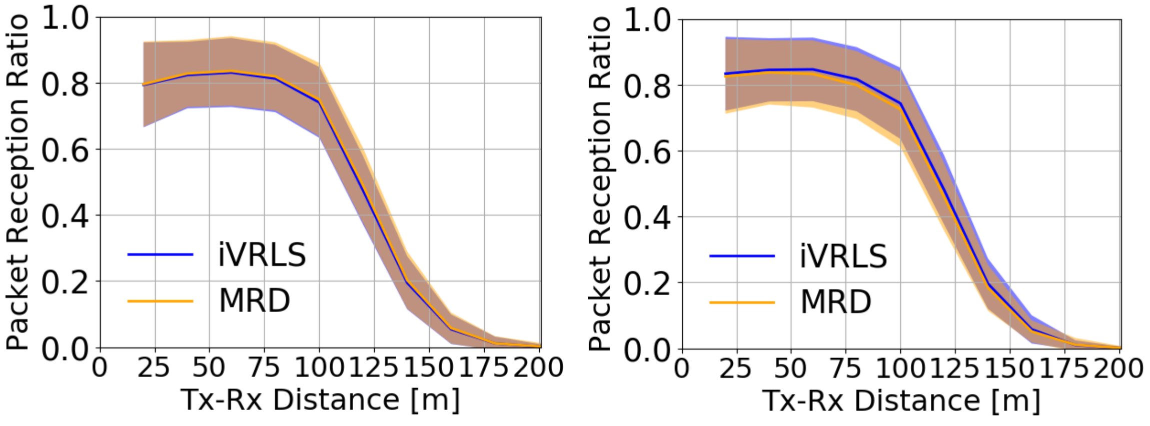
Finally, we provide the results for Scenario D in Fig. 8.12, from the in-coverage sections ($[-1000,-500]$ m and $[+500,+1000]$ m) and the tunnel section ($[-500,+500]$ m) of the road. In Scenario D, we observe from Fig. 8.12 that the performance of the algorithms within the coverage and the tunnel sections of the road are similar, which is different than our observations in other scenarios we considered, where performance of the algorithms under coverage was relatively better. This is because of the BS being deployed at the center of



(a) West section.

(b) Tunnel sections.

Fig. 8.11 PRR in Scenario C, measured as a function of Tx-Rx distance, at the west and the tunnel sections of the road.



(a) In-coverage sections.

(b) Tunnel section.

Fig. 8.12 PRR in Scenario D, measured as a function of Tx-Rx distance, at the in-coverage and the tunnel sections of the road.

the road with a larger distance to the served vehicles as compared to other scenarios, thus, vehicles suffering from higher path loss under coverage. Vehicles at the edge sections of the road get better coverage as they approach to the central section, which also improves the reliability of the transmissions received by the vehicles within the tunnel. Similar to our other results, iVRLS can provide a marginally better communication range within the tunnel, as compared to MRD.

In summary, our evaluations in this chapter lead us to the following observations:

- Centralized scheduling of V2V communications gets more efficient under coverage, thanks to the availability of instantaneous and global knowledge of the vehicular environment at the scheduler. As the coverage conditions get more ideal in comparison to the out-of-coverage areas, fewer resources become necessary to support the required reliability of V2V applications.
- VRLS, when extended for scheduling under coverage, performs marginally worse than iVRLS and MRD, which are specially designed for in-coverage scheduling.
- Without our proposed enhancement to centralized schedulers under intermittent coverage, the reliability performance of V2V communications degrade considerably even with the highest possible frequency of the scheduling updates under a low traffic load.
- Results in Scenarios C and D show that the large path loss between the vehicles and the network due to the non-ideal deployment of base stations degrade the reliability of the V2V messages in a similar way to that of an out-of-coverage area such as a tunnel on the road.
- While relatively good coverage conditions benefit MRD, which is designed assuming perfect coverage conditions; in case of coverage losses, iVRLS yields marginally larger V2V distances at which a given target PRR is achievable.
- Under coverage, more frequent message transmissions or larger density of vehicles, such as due to stop-and-go traffic, degrade the reliability of the V2V transmissions for all scheduling algorithms, due to the increased load on the resources, as expected. The reliability can be improved by increasing the frequency of the scheduling updates, which, however, incurs additional control signaling overhead.
- In the case of relatively low traffic load, iVRLS and MRD perform very similarly, while iVRLS can make more efficient use of the resources by providing a larger reliable communication range under high traffic load or relatively low rate of scheduling updates.

8.3.6 Challenges of Deploying iVRLS

The performance gain of iVRLS comes with the cost of training and higher computational complexity, as compared to the heuristic-based approaches, such as MRD. As with any other RL solution, iVRLS requires training, which could be conducted offline, i.e., before its deployment. On the other hand, during its operation, iVRLS needs to compute the state information of the environment and process it with its policy DNN, which incurs a higher processing cost than the operations required by the heuristic algorithms. Despite these challenges, iVRLS would be preferable for deployment to serve areas with rather non-ideal conditions such as intermittent network coverage given its advantageous performance, where area-specific training and operation could be performed.

8.4 Conclusions

Different from the previous chapters, where we considered V2V communications inside expected out-of-coverage areas, in this chapter, we shifted our focus to in-coverage operation, which is, however, vulnerable to unexpected and shorter coverage interruptions in reality. Motivated by our results in the previous chapters showing the benefits of the RL-based approach, we proposed iVRLS, an RL-based centralized scheduler for V2V communications, specifically targeting imperfect network coverage conditions. To support such conditions, state-of-the-art centralized schedulers could be extended by a simple method we propose, where vehicles continue keeping their resources for their V2V transmissions during the periods of coverage loss. Nevertheless, iVRLS performs better than the enhanced version of a state-of-the-art heuristic algorithm under intermittent coverage conditions. In particular, iVRLS can deliver a larger reliable communication range in the case of high traffic load and less frequent scheduling of V2V messages, as well as within the road tunnels without any network coverage. By utilizing the V2V communication resources more efficiently as the network conditions move away from the ideal, iVRLS offers a robust alternative to the existing schedulers across varying conditions of coverage.

Chapter 9

Conclusion and Outlook

The main goal of this thesis was to efficiently and practically address the resource allocation problem for maintaining vehicle-to-vehicle (V2V) communication performance under intermittent cellular network coverage conditions. To this aim, our main approach was based on the idea of exploiting centralized coordination of resources for V2V communications in areas suffering from intermittent coverage. We started by exploring reserving required resources for expected out-of-coverage areas to satisfy the quality of service requirements, and whether resources could be *pre-scheduled* by a centralized network entity efficiently, before vehicles enter such an area. For these tasks, we proposed the centralized scheduler to make use of the available information on its side, such as V2V data traffic demand, vehicle density, as well as predicted mobility of vehicles. We evaluated how varying conditions outside the coverage together with the imperfections in the predictions impact the performance of V2V communications, such as in terms of reliability, latency, and resource utilization. We showed that the idea of centrally reserving and pre-scheduling the resources is feasible. However, the performance of the V2V communications is highly impacted by the erroneous predictions of the scheduler when combined with the varying conditions outside the network coverage.

To make efficient predictions to schedule resources for expected out-of-coverage areas, we proposed a novel approach based on reinforcement learning (RL), motivated by the recent success of RL in many fields including vehicular communications. We proposed a centralized RL agent that learns to assign resources to the vehicles by utilizing the information available, such as the occupancy of resources. We showed that the RL-based scheduler can learn strategies to avoid resource conflicts and make efficient resource reuse enabling it to achieve better performance than the state-of-the-art solutions in terms of V2V packet delivery rate, after reasonable training times. Nevertheless, as confronted commonly in RL-based studies, our solution necessitated careful design considerations in terms of practical applicability to diverse environments, which V2V communications need to support. To address this concern, we proposed a unified RL-based solution for scheduling V2V communications in a diverse set of out-of-coverage areas that vary in terms of vehicle mobility, wireless channel characteristics, and structure of the communication resource pool. We have evaluated the performance of our solution in scenarios having different vehicular density and mobility, V2V data traffic, transmit power, and resource pool configuration. The proposed RL agent can learn a scheduling policy that is robust and adaptable to changes in the environment, thus eliminating the need for targeted (re-)training in complex real-life environments. Specifically, our solution reduces the packet loss rate by half as compared to a state-of-the-art distributed scheduler in highly loaded conditions and performs close to the theoretical maximum level in low-load scenarios. The RL scheduler requires much fewer resources as compared to the distributed scheduler to achieve the reliability targets required by the V2V applications. Further, the RL agent can provide higher awareness among the vehicular users, while delivering similar fairness and latency as the distributed scheduler.

On the other hand, as RL-based solutions require training before their deployment, they might encounter unanticipated network conditions during their operation, which they had never experienced during training. Especially considering the outside of coverage, unexpected or unnoticeable conditions within the vehicular environment may arise, such as due to road accidents that are leading to heavy traffic, where the actions of the trained agent might become unfeasible. Considering such conditions, where the information available at the centralized scheduling entity becomes a limiting factor, we proposed a hybrid solution that combines the centralized RL-based scheduling with the distributed scheduling approach. The employed RL scheduler recommends a subset of resources to the vehicles going outside the coverage. Whereas vehicles dynamically and locally apply an energy-sensing mechanism on the recommended resources. They consequently select the final resource for their transmission by weighing the RL-recommended probabilities with their sensing results. This way, we combine the best of both worlds: the global view (V2V traffic load, mobility, etc.) and the policy of the trained RL agent, with the local view of the vehicles enabling adaptation to dynamic, unpredictable conditions. We showed the ability of such a solution to improve V2V communication performance in a scenario with extreme traffic congestion due to an accident outside the network coverage.

Finally, we shifted our focus onto the in-coverage scheduling of V2V communications, in particular having imperfect connectivity between the vehicles and the BSs that provide access to the centralized scheduling entity, which is contrary to the unrealistic assumptions of the state-of-the-art solutions. As a basic enhancement to the existing centralized scheduling approaches, we proposed that vehicles keep using their allocated resources upon intermittent coverage loss until they can establish back the connection with the BS. Further, we proposed an RL-based centralized scheduler, similar to our solution for the out-of-coverage areas, which in this case can benefit from the availability of instantaneous and exact knowledge of the vehicular environment, and the possibility of resource assignments at all times, all thanks to the network coverage. The results showed that our RL-based scheduler achieved more efficient usage of resources by providing higher V2V communication reliability in comparison to the enhanced version of a state-of-the-art scheduling algorithm under non-ideal coverage conditions or high traffic load, and otherwise delivering similar performance.

We see several possible ways forward to extend this thesis work:

- Since we target the out-of-coverage problem for vehicular communications, we have mainly considered highway scenarios in this work, with parts of the road lacking cellular network coverage, which is the most typical case encountered in reality. The predictive resource allocation concept for vehicles, however, could be extended to *urban scenarios or more complex road topologies*, especially considering V2V communications under coverage. Similarly, the idea of managing resources beyond the network coverage could be extended for *other types of direct device-to-device communications* (e.g., for industrial IoT or public safety applications) involving different use cases, deployment scenarios, communication requirements, and wireless channel characteristics.

- The centralized coordination of the resources for V2V communications taking place outside the coverage could be extended with the idea of *relaying*, where vehicles could exchange information with the resource management entity via multi-hop transmissions. Such functionality could provide more dynamic and granular coordination of the communication resources, yet at the cost of increased complexity, larger signaling overhead, and higher latency.

- The resource allocation task we have considered could be extended to *directional transmissions*, e.g., using multi-antenna systems, involving further mechanisms such as beamforming and power control for the V2V transmissions at the physical layer, thus further exploiting resource reuse over the spatial domain. Such extensions would benefit unicast and multicast communications with directed transmissions, as envisioned for the next generation of V2V use cases. Given the further increased combinatorial complexity, it would be interesting to see whether and how RL can handle such resource allocation problems.

- The way we have applied RL to the resource allocation problem could be also extended in various ways. The impact of using *different learning techniques*, such as using graphstructured inputs, or distributed multi-agent approaches by combining the learning at the centralized network entity and the vehicle sides could be studied.

- Finally, although we have used extensively realistic simulators, evaluated performance on a variety of complex and realistic scenarios, and discussed the real-world implications of our approach, it would be necessary to see its performance in *real life*, especially with regards to training and operating the RL models. It would be interesting to observe and address the challenges arising in reality, which we might have omitted.

Appendix A

Design of the deep neural network architecture and data augmentation technique of VRLS

Design options

D1 Input layer: 4 separate 1D convolutional layers with tanh activation function, each processing one of the 4 columns of s_t. Hidden layer: merging layer and a 1D convolutional layer with tanh activation function. Output layer: fully connected layer with softmax activation function. This is a conventional design for convolutional neural networks, where similar architectures are utilized in many applications, such as [172], [171] and [293]. We have used this design in Chapter 5 for the vehicular environment E2.

D2 Same as D1, except that a maxpooling operation is added to the output of the hidden layer. This is another conventional variant of convolutional neural networks, especially widely used in image recognition and classification tasks, such as in [169] and [285].

D3 Input layer: single 1D convolutional layer with tanh activation function processing all s_t. Hidden layer: maxpooling operation followed by a perceptron with tanh activation function, repeated for number of subchannels in the resource pool (e.g., $K = 2$), each processing the corresponding part of the input layer's output. Output layer: fully connected layer with softmax activation function. We have designed this option based on the method proposed in [294] to overcome the dependence of actions on the ordering of elements in s_t.

D4 Same as D3, except that another hidden layer is added before the output layer with a single maxpooling operation followed by a perceptron as in D3. Based on [294], the cascaded layers still hold the ordering property.

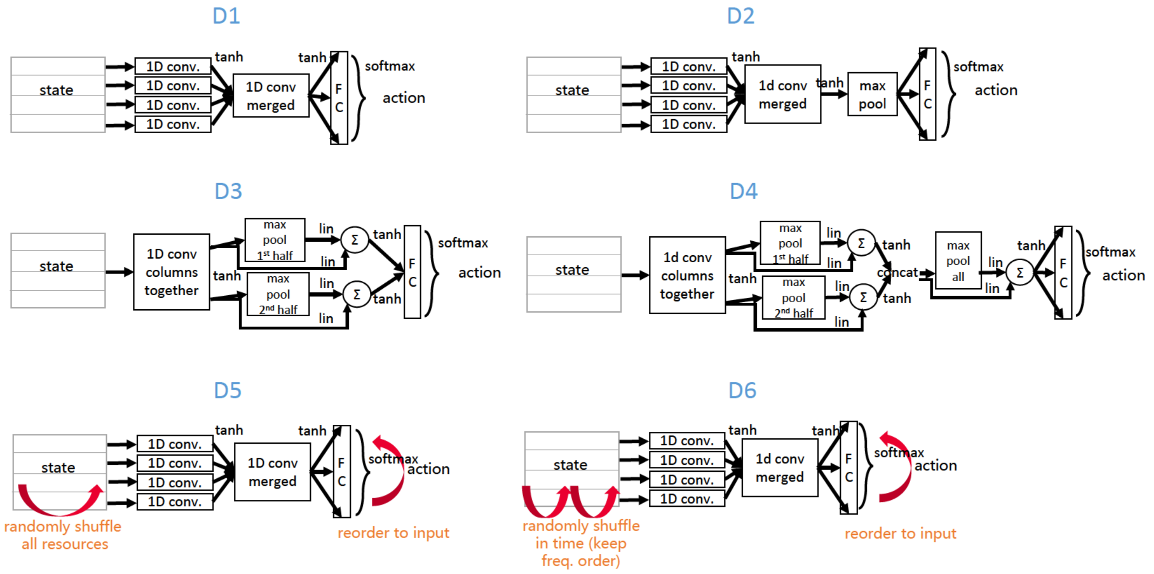


Fig. A.1 Considered design options D1-D6 for DNN architecture and data augmentation.

D5 Same as D1, except that we do data augmentation as described in Section 6.2.4. We have used this design for VRLS in Chapter 6, for the RL agent for hybrid scheduling in Chapter 7, and for iVRLS in Chapter 8.

D6 Same as D1, except that we do data augmentation by randomly shuffling the resources only in time domain, different than shuffling both in time and frequency domain as in D5.

Fig. A.1 illustrates the design options D1-D6 described above.

In Fig. A.2 we report the learning curves of RL agents denoted A1-A6 using the design options D1-D6. The agents are trained in the training E0 environment described in Section 6.3.1. All agents utilize the same state and reward definitions in Section 6.2.

In Fig. A.3 we provide the V2V reliability performance of the RL agents A1-A6 evaluated in the training environment E0, in terms of packet reception ratio (PRR) defined in Section 3.2. For each agent, we report PRR measured at two ranges $R1 = 0-50$ m and $R2 = 50-100$ m around the transmitters, in terms of the mean and percentile values of 10000 measurements collected every 10 s from the environment.

From the results in Fig. A.2 and Fig. A.3, we make the following observations:

- Adding a pooling layer to the DNN leads to a higher variance in the asymptotic performance of the agent (by comparing A1 and A2 in Fig. A.2). Similarly, A2 results in a lower mean and higher deviation of PRR as compared to A1 as observed from Fig. A.3.

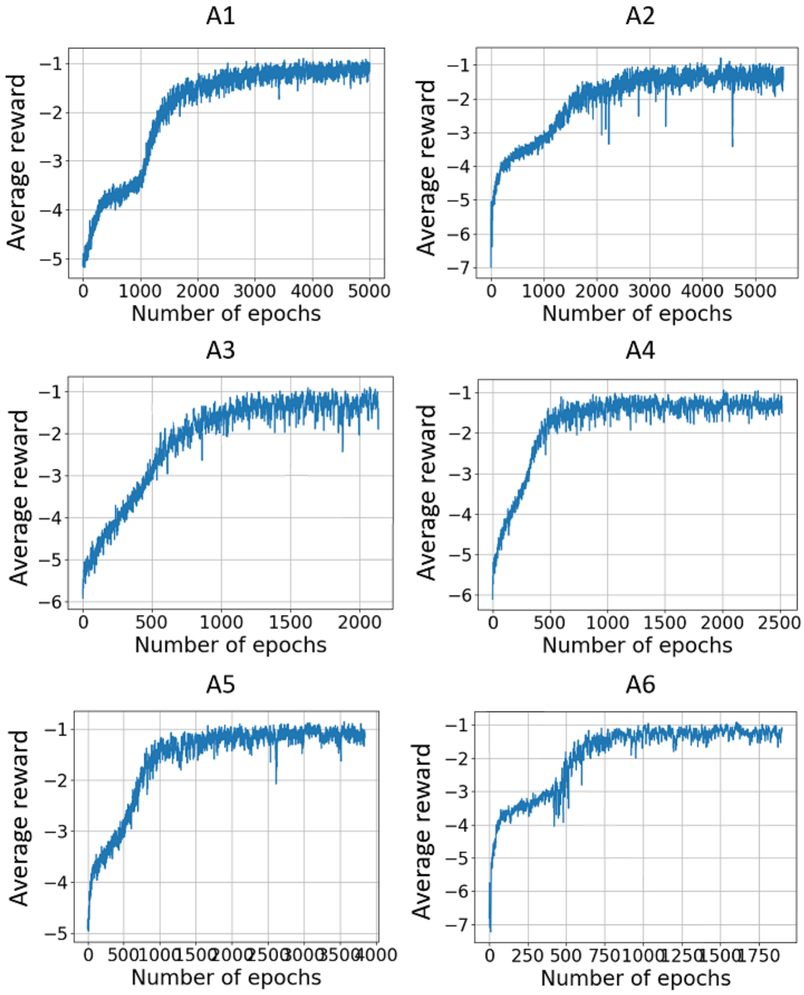


Fig. A.2 Learning curves of the RL agents A1-A6 that use the designs D1-D6, respectively, in the training environment E0 described in Section 6.3.1.

- The architecture design considered in [294] (i.e., in A3 and A4) leads to faster convergence in learning as could be observed from Fig. A.2. Agents with this design (A3 and A4) can achieve marginally lower performance than A1 (Fig. A.3). On the other hand, this architecture is not practical as the number of maxpooling and perceptron pairs in the hidden layer needs to be varied according to the number of subchannels in the resource pool, thus requiring a different variant of the DNN to be trained for every possible resource pool configuration.

- The data augmentation increases the convergence rate of learning, as observed by comparing A5 and A6 with A1 and A2 in Fig. A.2. Further, shuffling of the resources randomly both in time and frequency leads to a better performance in terms of PRR

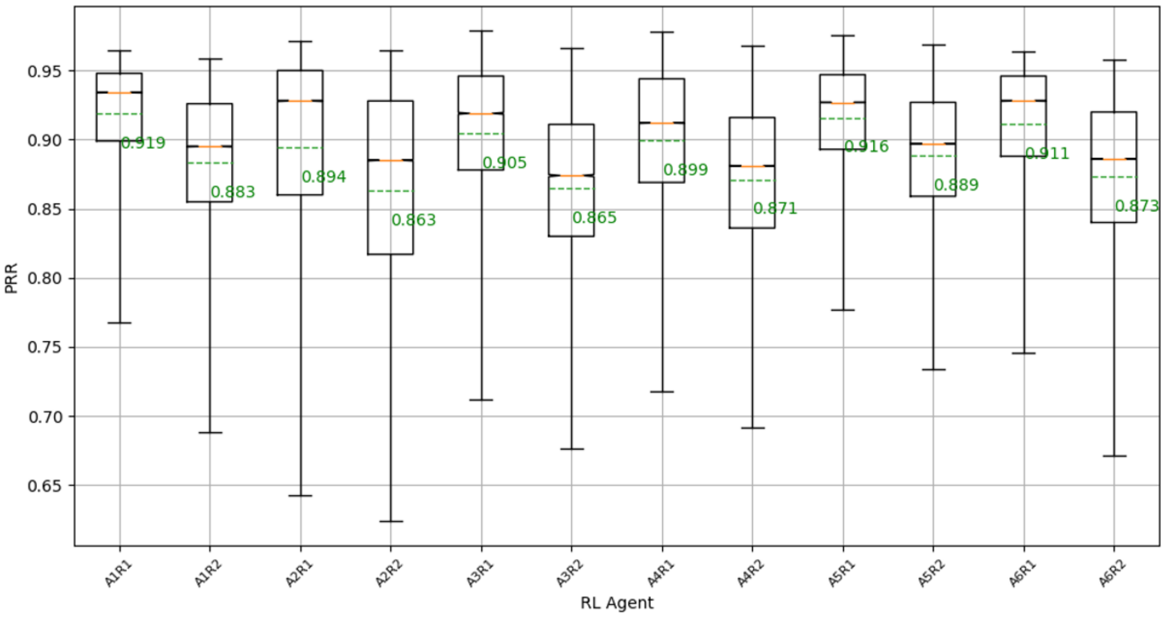


Fig. A.3 PRR performance of the RL agents A1-A6 in the environment E0, measured for two ranges $R1 = 0 - 50$ m and $R2 = 50 - 100$ m around the transmitters, with mean (green, dashed, denoted), median (red), 25th and 75th percentiles (box), and 5th and 95th percentiles (whiskers).

with a relatively larger mean and less variance than shuffling the resources only in time.

Our observations lead to the selection of the design option D5 for our RL agents in the respective chapters, which achieves relatively fast convergence in training and the best reliability performance in the considered environment among the examined options.

Appendix B

Design and training of the iVRLS agent

State representation design options

S1 Same state representation as in VRLS (cf. Section 6.2.1), except that instead of Δx, we represent I as in iVRLS (cf. Section 8.2.1), for each direction. Namely, s_t has 4 columns, two for each road direction, representing the normalized number of vehicles C each resource r assigned to, and the expected interference level I_r on that resource, per direction, respectively.

S2 Same as S1, however, with a total of 2 columns, each representing the C and I *for both directions*, respectively. That is, for each resource, information on both directions are taken into account when calculating its C and I.

S3 State representation with a single column only for I containing information for both directions. We have used this design for iVRLS in Chapter 8.

In Fig. B.1 we report the learning curves of the RL agents denoted A1-A3 utilizing the state representations S1-S3, in the training environment described in Section 8.3.1. All agents utilize the same action and reward definitions, as well as the same DNN architecture described in Section 8.2. As could be observed from Fig. B.1, A1 and A2 show convergence after around 10000 epochs, however with some variance in terms of performance. Whereas, A3 achieves the best performance with marginally larger average reward and much less variance, after around 15000 epochs. The results indicate that separate state representation per road direction, as well as representing the resource occupancy information become unnecessary in the case of in-coverage scheduling, which leads us to select option S3 for iVRLS.

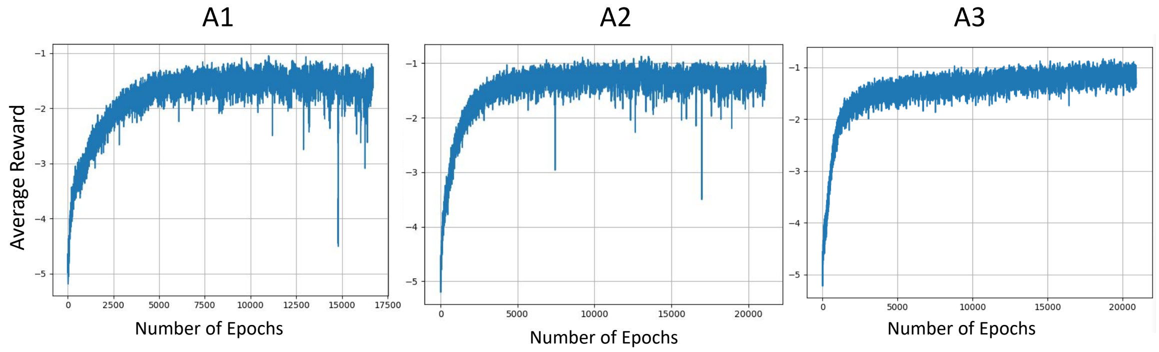


Fig. B.1 Learning curves of the RL agents A1-A3 that use designs D1-D3, respectively, in the training environment described in Section 8.3.1.

DNN architecture design options

We consider the following DNN architecture options in the design of the RL agent utilizing the state representation S3:

D1 Similar design as used for VRLS (cf. Section 6.2.3), where only a single 1D convolutional layer is utilized at the input layer, as S3 consists of a single column. The hidden and output layers consist of another convolutional and a fully connected layer, respectively, as in VRLS.

D2 Same as D1, however, the hidden convolutional layer is replaced by a fully connected layer, with the number of units equal to the number of resources.

D3 All three layers (input, hidden, output) are fully connected, each with a number of units equal to the number of resources.

We report the learning curves of the agents utilizing the DNN architectures D2 and D3 in Fig. B.2 (D1 is utilized by the agent A3 in Fig. B.1). As Fig. B.2 demonstrates, replacing the convolutional layers with fully connected ones highly degrades the learning performance. In the case of D2, where the hidden layer is replaced with a fully connected layer, the agent can achieve much lower average reward as compared to A3, even after a training of more than 25000 epochs. In the case of D3, where all layers of the DNNs are fully connected, the agent does not even show a sign of convergence within the first 1000 epochs, while D2 can lead to a much faster convergence as could be seen from the magnifying box in Fig. B.2. As discussed in Section 2.5.1, convolutional layers offer the advantage in terms of learning speed and performance, thanks to much fewer parameters that need to be trained. We, therefore, consider option A3 using D1 in our iVRLS design.

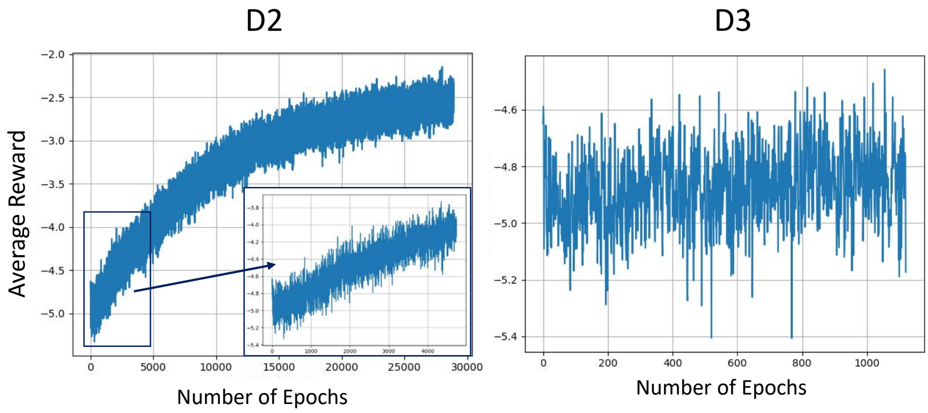


Fig. B.2 Learning curves of the RL agents that use the DNN designs D2 and D3 with the state representation S3, in the training environment described in Section 8.3.1.

Learning rate design options

We evaluate the impact of learning rate α on the learning performance of the RL agent. In the design of VRLS and iVRLS we have considered $\alpha = 10^{-3}/(1+0.01 \times \#ep^{1.1})$, where the learning rate gets smaller with the number of epochs in order to enable better convergence properties as discussed in Section 2.5.1, starting from the initial value of 0.001. Besides this considered option, we have evaluated the learning performance where we set the initial value of α to 0.002, 0.005, 0.01, and 0.05. Except for the first value, we could not observe any learning, i.e., the agents did not show any convergence in terms of collected average reward. We provide the learning curve for the case 0.002 in Fig. B.3. It took around two times longer for the agent to achieve a comparable performance with A3 that utilizes the initial value of 0.001, where it showed a larger variance after convergence. We have therefore considered the initial value of 0.001 for the learning rate in our design.

Impact of the scheduling periodicity on training

We evaluate the impact of the frequency of scheduling updates on the training performance, where each vehicle in the environment sends a scheduling request with periodicity $T_{\mathrm{SR}} = N_{\mathrm{SR}} T_{\mathrm{m}}$ (i.e., in multiples of T_{m}), starting from its first message generation after connecting to the network. We evaluate the learning performance of the agent A3 for the cases $N_{\mathrm{SR}} = \{1, 10, 100\}$ with the message generation periodicity $T_{\mathrm{m}} = 100$ ms, and report the learning curves in Fig. B.4.

In the case of $N_{\mathrm{SR}} = 1$, all vehicles in the environment request a new resource for every single V2V transmission. Such a large number of requests (growing with the number of vehicles in the environment) within a short time makes it very difficult for the RL agent to

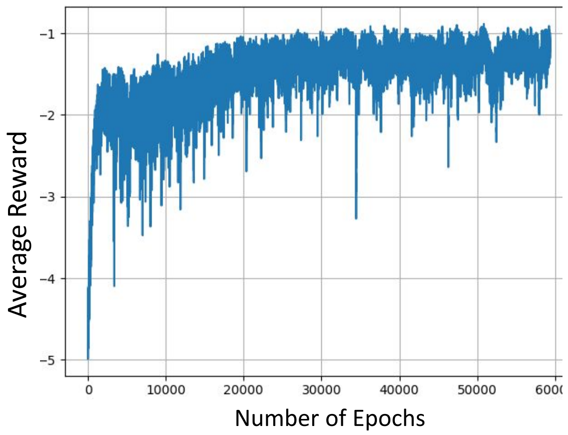


Fig. B.3 Learning curve of RL agent A3, with the initial value of learning rate set to 0.002, in the training environment described in Section 8.3.1.

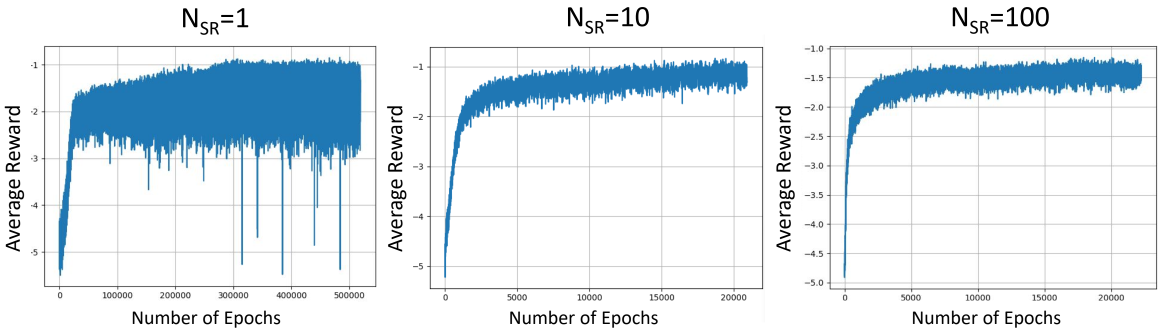


Fig. B.4 Learning curves of RL agent A3 with different periodicities of scheduling each vehicle, expressed in multiples N_{SR} of their V2V message generation periodicity T_{m}, in the training environment described in Section 8.3.1.

assess the quality of its actions. Namely, the reward reflecting the PRR calculated within a very short duration (even less than the packet generation periodicity T_{m}) might not represent the reliability of the transmitted messages in the environment properly. Accordingly, the RL agent can not converge to a stable performance level, with an increasing variance throughout the training. On the other extreme, in the case of $N_{\mathrm{SR}} = 100$, the agent reaches a stable performance level, however, capped at a smaller maximum of average reward due to the lower frequency of scheduling updates. A smaller rate of scheduling requests coming from the vehicles prevents the agent from observing the changing dynamics of the environment, which makes its decisions sub-optimal. As a consequence, we utilize $N_{\mathrm{SR}} = 10$ when training iVRLS, which leads to a stable performance level with a larger average reward than the other cases.

List of Figures

List of Tables

List of Abbreviations

1D	one-dimensional
2D	two-dimensional
2G	second-generation (cellular network technology)
3G	third-generation (cellular network technology)
3GPP	3^{rd} Generation Partnership Project
4G	fourth-generation (cellular network technology)
5G	fifth-generation (cellular network technology)
A2C	advantage actor-critic
A3C	asynchronous advantage actor-critic
AC	admission control
ACK	acknowledgment
ACRL	actor-critic reinforcement learning
AI	artificial intelligence
AoI	age of information
APP	application
ARQ	automatic repeat request
BE	Bellman equations
BS	base station
BSM	basic safety message
BT	Bluetooth
C-ITS	Cooperative Intelligent Transport Systems
C-V2X	cellular V2X
CAM	cooperative awareness message
CDMA	code division multiple access
CG	configured grant

CNN	convolutional neural network
COST	European Cooperation in Science and Technology
CR	cognitive radio
CSI	channel state information
CSMA/CA	carrier-sense multiple access with collision avoidance
CSMA	carrier-sense multiple access
CW	contention window
D2D	device-to-device
DCC	decentralized congestion control
DCI	downlink control information
DDPG	deep deterministic policy gradient
DDQN	double deep Q-learning
DENM	decentralized environmental notification message
DL	downlink
DNN	deep neural network
DOCA	delimited out-of-coverage area
DQN	deep Q-network
DSRC	Dedicated Short-Range Communications
DTV	digital television
EDCA	enhanced distributed channel access
ETSI	European Telecommunications Standards Institute
EVA	emergency vehicle alert
FC	fully-connected
FCC	Federal Communications Commission
FCD	floating car data
FDD	frequency division duplex
FDMA	frequency division multiple access
HAPS	high-altitude platform systems
HARQ	hybrid automatic repeat request
HD	half-duplex

HetNet	heterogeneous network
I2V	infrastructure-to-vehicle
ID	identification
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IoV	Internet of vehicles
ISO	International Organization for Standardization
ITS-G5	Intelligent Transport Systems - 5.9 GHz
ITS	Intelligent Transport Systems
ITU	International Telecommunication Union
IVC	inter-vehicle communication
iVRLS	in-coverage Vehicular Reinforcement Learning Scheduler
KPI	key performance indicator
LAN	local area network
LOS	line-of-sight
LSTM	long short-term memory
LTE	Long Term Evolution
MAC	medium access control
MCD	multi collision domain
MC	Monte Carlo
MCS	modulation and coding scheme
MD	allocation with Maximum reuse Distance
MDP	Markov decision process
MIMO	multiple-input multiple-output
ML	machine learning
MPC	multi-path component
MRD	maximum reuse distance
N2V	network-to-vehicle
NIST	US National Institute of Standards and Technology
NOMA	non-orthogonal multiple access
NR	New (5G) Radio

ns-3	network simulator version 3
NTN	non-terrestrial network
OBU	on-board unit
OFDMA	orthogonal frequency-division multiple access
OFDM	orthogonal frequency-division multiplexing
OMNet++	Objective Modular Network Testbed in C++
OOC	out-of-coverage
P2V	pedestrian-to-vehicle
PDR	packet delivery ratio
PHY	physical
PI	policy iteration
PIR	packet inter-reception
PL	platoon lead
PPO	proximal policy optimization
PRR	packet reception ratio
QoS	quality of service
RADAR	radio detecting and ranging
RAN	radio access network
RB	resource block
RIC	radio access network intelligent controller
RL	reinforcement learning
RNN	recurrent neural network
RRC	radio resource control
RSSI	received signal strength indicator
RSU	roadside unit
RTS/CTS	request to send/clear to send
Rx	receiver
SAC	soft actor-critic
SAE	The US Society of Automotive Engineers
SARSA	state-action-reward-state-action
SA	scheduling assignment

SC-FDMA	single-carrier frequency-division multiple access
SCD	single collision domain
SDMA	space division multiple access
SINR	signal-to-interference-and-noise ratio
SL	sidelink
SNR	signal-to-noise ratio
SPS	semi-persistent scheduling
SR	scheduling request
SUMO	Simulation of Urban MObility
TB	transmission block
TBLER	transport block error rate
TDMA	time division multiple access
TD	temporal-difference
TIM	traveler information message
Tx	transmitter
UAV	unmanned aerial vehicle
UL	uplink
US	United States
USA	United States of America
UWB	ultra wide band
V2I	vehicle-to-infrastructure
V2N2I	vehicle-to-network-to-infrastructure
V2N2P	vehicle-to-network-to-pedestrian
V2N2V	vehicle-to-network-to-vehicle
V2N	vehicle-to-network
V2P	vehicle-to-pedestrian
V2V	vehicle-to-vehicle
VANET	vehicular ad-hoc network
VLC	visible light communications
VRLS	Vehicular Reinforcement Learning Scheduler
VRU	vulnerable road user

VSimRTI	V2X Simulation Runtime Infrastructure
WAVE	wireless access in vehicular environments
WLAN	wireless local area network
ZB	ZigBee

References

[1] National Geographic Society. (2022, May) Transportation Infrastructure. Publisher: National Geographic Society. [Online]. Available: https://education.nationalgeographic.org/resource/transportation-infrastructure

[2] World Health Organization, "Global status report on road safety 2018," World Health Organization, Geneva, Global report, 2018, ISBN: 978-92-4-156568-4. [Online]. Available: https://www.who.int/publications-detail-redirect/9789241565684

[3] B. Williams, *Intelligent Transport Systems Standards*. Boston: Artech House, 2008.

[4] (2021) Automotive Intelligent Transport Systems (ITS). ETSI. [Online]. Available: https://www.etsi.org/technologies/automotive-intelligent-transport

[5] M. Fallgren, M. Dillinger, T. Mahmoodi, and T. Svensson, *Cellular V2X for Connected Automated Driving*. Wiley Online Library, 2021.

[6] M. Zaki. (2016, Jun.) The path to 5G: Paving the road to tomorrow's autonomous vehicles [video]. Qualcomm. [Online]. Available: https://www.qualcomm.com/news/onq/2016/06/path-5g-paving-road-tomorrows-autonomous-vehicles

[7] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu, "Connected roads of the future: Use cases, requirements, and design considerations for vehicle-to-everything communications," *IEEE Vehicular Technology Magazine*, vol. 13, no. 3, pp. 110–123, 2018.

[8] "IEEE guide for wireless access in vehicular environments (WAVE) architecture," *IEEE Std 1609.0-2019 (Revision of IEEE Std 1609.0-2013)*, pp. 1–106, 2019.

[9] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.

[10] ETSI, *Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Trans-port Systems operating in the 5 GHz frequency band*, ETSI Std. ES 202 663 V1.1.0, November 2009.

[11] IEEE, *IEEE standard for information technology - local and metropolitan area networks-specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: Wireless access in vehicular environments*, IEEE Standards Association and others Std. 802.11 p, 2010.

[12] 3GPP, “Overall description of Radio Access Network (RAN) aspects for Vehicle-to-everything (V2X) based on LTE and NR,” 3GPP, Tech. Rep. TR 37.985, November 2019, v1.0.0.

[13] “5GAA V2X terms and definitions,” 5GAA Automotive Association, Technical report, 2017. [Online]. Available: https://5gaa.org/wp-content/uploads/2017/08/5GAA-V2X-Terms-and-Definitions110917.pdf

[14] G. Fodor, E. Dahlman, G. Mildh, S. Parkvall, N. Reider, G. Miklós, and Z. Turányi, “Design aspects of network assisted device-to-device communications,” *IEEE Communications Magazine*, vol. 50, no. 3, pp. 170–177, 2012.

[15] M. Klügel, “Operation and control of device-to-device communication in cellular networks,” Dr.-Ing. dissertation, Technische Universität München, 2018.

[16] M. Botsov, “Radio resource management for automotive device-to-device communication in future cellular networks,” Dr.-Ing. dissertation, Technische Universität Berlin, 2020.

[17] M. Boban and P. M. d’Orey, “Exploring the practical limits of cooperative awareness in vehicular communications,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3904–3916, 2016.

[18] ETSI TC ITS, *Intelligent Transport Systems; Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, Std. ETSI EN Std 302 637-2 V.1.3.1, 2014.

[19] ——, *Intelligent Transport Systems; Vehicular Communications; Basic Set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service*, Std. ETSI EN Std 302 637-3 V.1.2.0, 2013.

[20] ——, “Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions,” Tech. Rep. ETSI TR 102 638 V1.1.1, June 2009.

[21] 3GPP, *Enhancement of 3GPP support for V2X scenarios*, 3GPP Std. TS 22.186, December 2018, v16.1.0.

[22] ——, *Service requirements for V2X services*, 3GPP Std. TS 22.185, June 2018, v15.0.0.

[23] “C-V2X use cases and service level requirements volume I,” 5GAA Automotive Association, Technical report, 2020. [Online]. Available: https://https://5gaa.org/wp-content/uploads/2020/12/5GAA_T-200111_TR_C-V2X_Use_Cases_and_Service_Level_Requirements_Vol_I-V3.pdf

[24] M. Noor-A-Rahim, Z. Liu, H. Lee, G. G. M. N. Ali, D. Pesch, and P. Xiao, “A survey on resource allocation in vehicular networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 701–721, 2022.

[25] D. Calabuig, D. Martin-Sacristan, M. Botsov, J. F. Monserrat, and D. Gozalvez, “Comparison of LTE centralized RRM and IEEE 802.11 decentralized RRM for ITS cooperative awareness,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2018, pp. 1–6.

[26] M. Wang, M. Winbjork, Z. Zhang, R. Blasco, H. Do, S. Sorrentino, M. Belleschi, and Y. Zang, "Comparison of LTE and DSRC-based connectivity for intelligent transportation systems," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, 2017, pp. 1–5.

[27] D. Sempere-García, M. Sepulcre, and J. Gozalvez, "LTE-V2X mode 3 scheduling based on adaptive spatial reuse of radio resources," *Ad Hoc Networks*, vol. 113, p. 102351, 2021.

[28] R. Borralho, A. Mohamed, A. U. Quddus, P. Vieira, and R. Tafazolli, "A survey on coverage enhancement in cellular networks: Challenges and solutions for future deployments," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1302–1341, 2021.

[29] H. Ye, L. Liang, G. Y. 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, June 2018.

[30] L. Liang, H. Ye, G. Yu, and G. Y. Li, "Deep-learning-based wireless resource allocation with application to vehicular networks," *Proceedings of the IEEE*, vol. 108, no. 2, pp. 341–356, 2020.

[31] H. Mao, M. Alizadeh, I. Menache, and S. Kandula, "Resource management with deep reinforcement learning," in *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*. ACM, 2016, pp. 50–56.

[32] G. Cecchini, A. Bazzi, M. Menarini, B. M. Masini, and A. Zanella, "Maximum reuse distance scheduling for cellular-V2X sidelink mode 3," in *2018 IEEE Globecom Workshops (GC Wkshps)*, 2018, pp. 1–6.

[33] T. Sahin and M. Boban, "Radio resource allocation for reliable out-of-coverage V2V communications," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[34] T. Şahin, R. Khalili, M. Boban, and A. Wolisz, "Reinforcement learning scheduler for vehicle-to-vehicle communications outside coverage," in *2018 IEEE Vehicular Networking Conference (VNC)*, 2018, pp. 1–8.

[35] T. Sahin, R. Khalili, M. Boban, and A. Wolisz, "VRLS: A unified reinforcement learning scheduler for vehicle-to-vehicle communications," in *2019 IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, 2019, pp. 1–7.

[36] T. Şahin, M. Boban, R. Khalili, and A. Wolisz, "A hybrid sensing and reinforcement learning scheduler for vehicle-to-vehicle communications," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1136–1143.

[37] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Şahin, and A. Kousaridas, "A tutorial on 5G NR V2X communications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1972–2026, 2021.

[38] T. Şahin, M. Boban, R. Khalili, and A. Wolisz, “iVRLS: In-coverage vehicular reinforcement learning scheduler,” in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1–7.

[39] T. Şahin, R. Khalili, M. Boban, and A. Wolisz, “Scheduling out-of-coverage vehicular communications using reinforcement learning,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 11 103–11 119, 2022.

[40] T. Sahin and M. Boban, “Devices and methods for D2D communication,” Patent WO2 019 174 744, Sep., 2019. [Online]. Available: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2019174744

[41] T. Sahin, M. Boban, and M. Webb, “Network entity, user equipments and methods for using sidelink resources,” Patent WO2 020 244 741, Dec., 2020. [Online]. Available: https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2020244741

[42] “List of C-V2X Devices,” 5GAA Automotive Association, Technical report, 2021. [Online]. Available: https://5gaa.org/wp-content/uploads/2021/11/5GAA_List_of_C_V2X_devices.pdf

[43] ETSI TC ITS, “Intelligent Transport System (ITS); Vulnerable Road Users (VRU) awareness; Part 1: Use Cases definition; Release 2,” Tech. Rep. ETSI TR 103 300-1 V2.1.1, September 2019.

[44] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[45] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-Advanced for Mobile Broadband*. Academic press, 2013.

[46] H. Moustafa and Y. Zhang, *Vehicular Networks: Techniques, Standards and Applications*. Auerbach Publications, 2009.

[47] 3GPP, “Study on LTE support for Vehicle-to-Everything (V2X) services,” 3GPP, Tech. Rep. TR 22.885, September 2015, v1.0.0.

[48] K. Laberteaux and H. Hartenstein, *VANET: Vehicular Applications and Inter-Networking Technologies*. John Wiley & Sons, 2009.

[49] T. Abbas, *Measurement Based Channel Characterization and Modeling for Vehicle-to-vehicle Communications*. Lund University, 2014.

[50] R. A. Uzcategui, A. J. De Sucre, and G. Acosta-Marum, “Wave: A tutorial,” *IEEE Communications Magazine*, vol. 47, no. 5, pp. 126–133, 2009.

[51] A. Festag, “Cooperative intelligent transport systems standards in Europe,” *IEEE Communications Magazine*, vol. 52, no. 12, pp. 166–172, 2014.

[52] M. Harounabadi, D. M. Soleymani, S. Bhadauria, M. Leyh, and E. Roth-Mandutz, “V2X in 3GPP standardization: NR sidelink in Release-16 and beyond,” *IEEE Communications Standards Magazine*, vol. 5, no. 1, pp. 12–21, 2021.

[53] SAE V2X Core Technical Committee, *V2X Communications Message Set Dictionary*, SAE International Std. J2735, July 2020.

[54] S. Bartoletti, B. M. Masini, V. Martinez, I. Sarris, and A. Bazzi, "Impact of the generation interval on the performance of sidelink C-V2X autonomous mode," *IEEE Access*, vol. 9, pp. 35 121–35 135, 2021.

[55] The CAMP Vehicle Safety Communications Consortium, "Vehicle Safety Communications Project Task 3 Final Report Identify Intelligent Vehicle Safety Applications Enabled by DSRC," U.S. Department of Transportation, NHTSA, Tech. Rep. DOT HS 809 859, 2005.

[56] S. Mumtaz, K. M. Saidul Huq, M. I. Ashraf, J. Rodriguez, V. Monteiro, and C. Politis, "Cognitive vehicular communication for 5G," *IEEE Communications Magazine*, vol. 53, no. 7, pp. 109–117, 2015.

[57] A. Pyattaev, K. Johnsson, S. Andreev, and Y. Koucheryavy, "Proximity-based data offloading via network assisted device-to-device communications," in *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, 2013, pp. 1–5.

[58] C.-H. Yu, K. Doppler, C. B. Ribeiro, and O. Tirkkonen, "Resource sharing optimization for device-to-device communication underlaying cellular networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 8, pp. 2752–2763, 2011.

[59] A. F. Molisch, *Wireless Communications*, 2nd ed. Wiley Publishing, 2011.

[60] K. Serizawa, M. Mikami, K. Moto, and H. Yoshino, "Field trial activities on 5G NR V2V direct communication towards application to truck platooning," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–5.

[61] D. Martín-Sacristán, S. Roger, D. Garcia-Roger, J. F. Monserrat, A. Kousaridas, P. Spapis, S. Ayaz, and C. Zhou, "Evaluation of LTE-Advanced connectivity options for the provisioning of V2X services," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1–6.

[62] M. C. Lucas-Estañ, B. Coll-Perales, C.-H. Wang, T. Shimizu, S. Avedisov, T. Higuchi, B. Cheng, A. Yamamuro, J. Gozalvez, M. Sepulcre, and O. Altintas, "On the scalability of the 5G RAN to support advanced V2X services," in *2020 IEEE Vehicular Networking Conference (VNC)*, 2020, pp. 1–4.

[63] M. C. Lucas-Estañ, B. Coll-Perales, T. Shimizu, J. Gozalvez, C.-H. Wang, B. Cheng, M. Sepulcre, S. Avedisov, T. Higuchi, and O. Altintas, "Analysis of 5G RAN configuration to support advanced V2X services," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1–5.

[64] Z. MacHardy, A. Khan, K. Obana, and S. Iwashina, "V2X access technologies: Regulation, research, and remaining challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1858–1877, 2018.

[65] J. Vučič, C. Kottke, S. Nerreter, K.-D. Langer, and J. W. Walewski, "513 Mbit/s visible light communications link based on DMT-modulation of a white LED," *Journal of Lightwave Technology*, vol. 28, no. 24, pp. 3512–3518, 2010.

[66] M. L. Sichitiu and M. Kihl, "Inter-vehicle communication systems: a survey," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 2, pp. 88–105, 2008.

[67] X. Lin, S. Rommer, S. Euler, E. A. Yavuz, and R. S. Karlsson, “5G from space: An overview of 3GPP non-terrestrial networks,” *IEEE Communications Standards Magazine*, 2021.

[68] M. Giordani and M. Zorzi, “Non-terrestrial networks in the 6G era: Challenges and opportunities,” *IEEE Network*, vol. 35, no. 2, pp. 244–251, 2020.

[69] A. Asadi, Q. Wang, and V. Mancuso, “A survey on device-to-device communication in cellular networks,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1801–1819, 2014.

[70] A. Asadi, V. Mancuso, and P. Jacko, “Floating band D2D: Exploring and exploiting the potentials of adaptive D2D-enabled networks,” in *2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2015, pp. 1–9.

[71] A. K. Ligo and J. M. Peha, “Spectrum for V2X: Allocation and sharing,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 768–779, 2019.

[72] FCC. (1999, Oct.) FCC allocates spectrum in 5.9 GHz range for intelligent transportation systems uses. [Online]. Available: https://www.fcc.gov/document/fcc-allocates-spectrum-59-ghz-range-intelligent-transportation

[73] ——. (2020, Nov.) FCC modernizes 5.9 GHz band to improve Wi-Fi and automotive safety. [Online]. Available: https://www.fcc.gov/document/fcc-modernizes-59-ghz-band-improve-wi-fi-and-automotive-safety

[74] J. Choi, V. Marojevic, C. B. Dietrich, J. H. Reed, and S. Ahn, “Survey of spectrum regulation for intelligent transportation systems,” *IEEE Access*, vol. 8, pp. 140 145–140 160, 2020.

[75] “Deployment band configuration for C-V2X at 5.9 GHz in Europe,” 5GAA Automotive Association, Position paper, 2021. [Online]. Available: https://5gaa.org/wp-content/uploads/2021/06/5GAA_S-210019_Position-paper-on-European-deployment-band-configuration-for-C-V2X_final.pdf

[76] “White paper on ITS spectrum utilization in the Asia Pacific Region,” 2018. [Online]. Available: https://5gaa.org/wp-content/uploads/2018/07/5GAA_WhitePaper_ITS-spectrum-utilization-in-the-Asia-Pacific-Region_FINAL_160718docx.pdf

[77] L. Khoukhi, H. Badis, L. Merghem-Boulahia, and M. Esseghir, “Admission control in wireless ad hoc networks: a survey,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–13, 2013.

[78] M. Schneps-Schneppe and V. B. Iversen, “Call admission control in cellular networks,” in *Mobile Networks*, J. H. Ortiz, Ed. Rijeka: IntechOpen, 2012, ch. 7.

[79] M. Ahmed, “Call admission control in wireless networks: A comprehensive survey,” *IEEE Communications Surveys and Tutorials*, vol. 7, no. 1, pp. 49–68, 2005.

[80] A. Mansouri, V. Martinez, and J. Härri, "A first investigation of congestion control for LTE-V2X mode 4," in *2019 15th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, 2019, pp. 56–63.

[81] A. Bazzi, "Congestion control mechanisms in IEEE 802.11 p and sidelink C-V2X," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1125–1130.

[82] T. Kosch, C. J. Adler, S. Eichler, C. Schroth, and M. Strassberger, "The scalability problem of vehicular ad hoc networks and how to solve it," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 22–28, 2006.

[83] N. Abramson, "THE ALOHA SYSTEM: Another alternative for computer communications," in *Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, ser. AFIPS '70 (Fall). New York, NY, USA: Association for Computing Machinery, 1970, p. 281–285.

[84] L. G. Roberts, "ALOHA packet system with and without slots and capture," *SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, p. 28–42, apr 1975.

[85] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1400–1416, 1975.

[86] M. Amadeo, C. Campolo, and A. Molinaro, "Enhancing IEEE 802.11p/WAVE to provide infotainment applications in VANETs," *Ad Hoc Netw.*, vol. 10, no. 2, p. 253–269, Mar 2012.

[87] M. Hadded, P. Muhlethaler, A. Laouiti, R. Zagrouba, and L. A. Saidane, "TDMA-based MAC protocols for vehicular ad hoc networks: a survey, qualitative analysis, and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2461–2492, 2015.

[88] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, ser. MobiCom '99. New York, NY, USA: Association for Computing Machinery, 1999, p. 151–162.

[89] K. Tang and M. Gerla, "MAC layer broadcast support in 802.11 wireless networks," in *MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority (Cat. No.00CH37155)*, vol. 1, 2000, pp. 544–548 vol.1.

[90] Y. Gunter, B. Wiegel, and H. P. Grossmann, "Cluster-based medium access scheme for VANETs," in *2007 IEEE Intelligent Transportation Systems Conference*, 2007, pp. 343–348.

[91] W. Sun, D. Yuan, E. G. Ström, and F. Brännström, "Cluster-based radio resource management for D2D-supported safety-critical V2X communications," *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2756–2769, 2016.

[92] SAE DSRC Technical Committee, *On-Board System Requirements for V2V Safety Communications*, SAE International Std. J2945-1, April 2020.

[93] ETSI TC ITS, *Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part*, Std. ETSI TS 102 687 V1.2.1, April 2018.

[94] IEEE, *Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Enhancements for Next Generation V2X*, IEEE Standards Association and others Std. 802.11bd, 2018.

[95] W. Anwar, N. Franchi, and G. Fettweis, “Physical layer evaluation of V2X communications technologies: 5G NR-V2X, LTE-V2X, IEEE 802.11bd, and IEEE 802.11p,” in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–7.

[96] ETSI TC ITS, *Intelligent Transport Systems (ITS); Congestion Control Mechanisms for the C-V2X PC5 interface; Access layer part*, ETSI Std. ETSI TS 103 574 V1.1.1, November 2011.

[97] SAE C-V2X Technical Committee, *On-Board System Requirements for LTE-V2X V2V Safety Communications*, SAE International Std. J3161, March 2022, 1-202203.

[98] L. Bonati, M. Polese, S. D’Oro, S. Basagni, and T. Melodia, “Open, programmable, and virtualized 5G networks: State-of-the-art and the road ahead,” *Computer Networks*, vol. 182, p. 107516, 2020.

[99] 3GPP, *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*, 3GPP Std. TS 36.213, Dec. 2019, v16.0.0.

[100] ——, *Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification*, 3GPP Std. TS 36.321, Dec. 2019, v15.8.0.

[101] K. Sehla, T. M. T. Nguyen, G. Pujolle, and P. B. Velloso, “Resource allocation modes in C-V2X: From LTE-V2X to 5G-V2X,” *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8291–8314, 2022.

[102] A. Bazzi, A. O. Berthet, C. Campolo, B. M. Masini, A. Molinaro, and A. Zanella, “On the design of sidelink for cellular V2X: A literature review and outlook for future,” *IEEE Access*, 2021.

[103] M. Ma, K. Liu, X. Luo, T. Zhang, and F. Liu, “Review of MAC protocols for vehicular ad hoc networks,” *Sensors*, vol. 20, no. 23, p. 6709, 2020.

[104] M. A. Abd El-Gawad, M. Elsharief, and H. Kim, “A comparative experimental analysis of channel access protocols in vehicular networks,” *IEEE Access*, vol. 7, pp. 149 433–149 443, 2019.

[105] W. Guo, L. Huang, L. Chen, H. Xu, and J. Xie, “An adaptive collision-free MAC protocol based on TDMA for inter-vehicular communication,” in *2012 International Conference on Wireless Communications and Signal Processing (WCSP)*, 2012, pp. 1–6.

[106] R. Stanica, E. Chaput, and A.-L. Beylot, “Local density estimation for contention window adaptation in vehicular networks,” in *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, 2011, pp. 730–734.

[107] G. M. Abdalla, M. A. Abu-Rgheff, and S.-M. Senouci, “Space-orthogonal frequency-time medium access control (SOFT MAC) for VANET,” in *2009 Global Information Infrastructure Symposium*, 2009, pp. 1–8.

[108] M. Hadded, A. Laouiti, P. Muhlethaler, and L. A. Saidane, “An infrastructure-free slot assignment algorithm for reliable broadcast of periodic messages in vehicular ad hoc networks,” in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, 2016, pp. 1–7.

[109] F. Lyu, H. Zhu, H. Zhou, L. Qian, W. Xu, M. Li, and X. Shen, “MoMAC: Mobility-aware and collision-avoidance MAC for safety applications in VANETs,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10 590–10 602, 2018.

[110] L. Gallo and J. Haerri, “Unsupervised long-term evolution device-to-device: a case study for safety-critical V2X communications,” *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 69–77, June 2017.

[111] J. Yang, B. Pelletier, and B. Champagne, “Enhanced autonomous resource selection for LTE-based V2V communication,” in *2016 IEEE Vehicular Networking Conference (VNC)*, 2016, pp. 1–6.

[112] J. Kim, J. Lee, S. Moon, and I. Hwang, “A position-based resource allocation scheme for V2V communication,” *Wireless Personal Communications*, Aug 2017.

[113] 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.

[114] R. Molina-Masegosa, J. Gozalvez, and M. Sepulcre, “Configuration of the C-V2X mode 4 sidelink PC5 interface for vehicular communication,” in *2018 14th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*, 2018, pp. 43–48.

[115] R. Molina-Masegosa and J. Gozalvez, “System level evaluation of LTE-V2V mode 4 communications and its distributed scheduling,” in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, 2017, pp. 1–5.

[116] M. Gonzalez-Martín, M. Sepulcre, R. Molina-Masegosa, and J. Gozalvez, “Analytical models of the performance of C-V2X mode 4 vehicular communications,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1155–1166, 2018.

[117] A. Bazzi, G. Cecchini, A. Zanella, and B. M. Masini, “Study of the Impact of PHY and MAC parameters in 3GPP C-V2V mode 4,” *IEEE Access*, vol. 6, pp. 71 685–71 698, 2018.

[118] B. Toghi, M. Saifuddin, H. N. Mahjoub, M. O. Mughal, Y. P. Fallah, J. Rao, and S. Das, “Multiple access in cellular V2X: Performance analysis in highly congested vehicular networks,” in *2018 IEEE Vehicular Networking Conference (VNC)*, 2018, pp. 1–8.

[119] A. Nabil, K. Kaur, C. Dietrich, and V. Marojevic, “Performance analysis of sensing-based semi-persistent scheduling in C-V2X networks,” in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5.

[120] S.-Y. Jung, H.-R. Cheon, and J.-H. Kim, “Reducing consecutive collisions in sensing based semi persistent scheduling for cellular-V2X,” in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–5.

[121] P. Wendland, G. Schaefer, and R. S. Thomä, “LTE-V2X mode 4: Increasing robustness and DCC compatibility with reservation splitting,” in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, 2019, pp. 1–6.

[122] S. Heo, W. Yoo, H. Jang, and J.-M. Chung, “H-V2X mode 4 adaptive semipersistent scheduling control for cooperative internet of vehicles,” *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 678–10 692, 2021.

[123] R. Molina-Masegosa, M. Sepulcre, and J. Gozalvez, “Geo-based scheduling for C-V2X networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8397–8407, 2019.

[124] G. Cecchini, A. Bazzi, B. M. Masini, and A. Zanella, “MAP-RP: Map-based resource reselection procedure for autonomous LTE-V2V,” in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2017, pp. 1–6.

[125] S. Sabeeh, P. Sroka, and K. Wesołowski, “Estimation and reservation for autonomous resource selection in C-V2X mode 4,” in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2019, pp. 1–6.

[126] Y. Jeon and H. Kim, “An explicit reservation-augmented resource allocation scheme for C-V2X sidelink mode 4,” *IEEE Access*, vol. 8, pp. 147 241–147 255, 2020.

[127] L. F. Abanto-Leon, A. Koppelaar, and S. H. de Groot, “Enhanced C-V2X mode-4 subchannel selection,” in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5.

[128] S. Sharma and B. Singh, “Context aware autonomous resource selection and Q-learning based power control strategy for enhanced cooperative awareness in LTE-V2V communication,” *Wireless Networks*, vol. 26, no. 6, pp. 4045–4060, 2020.

[129] M. S. Almalag, S. Olariu, and M. C. Weigle, “TDMA cluster-based MAC for VANETs (TC-MAC),” in *2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2012, pp. 1–6.

[130] T.-L. Sheu and Y.-H. Lin, “A cluster-based TDMA system for inter-vehicle communications,” *Journal of Information Science and Engineering*, vol. 30, no. 1, pp. 213–231, 2014.

[131] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, “ADHOC MAC: New MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services,” *Wireless Networks*, vol. 10, no. 4, pp. 359–366, 2004.

[132] W. Guo, L. Huang, L. Chen, H. Xu, and C. Miao, "R-MAC: risk-aware dynamic MAC protocol for vehicular cooperative collision avoidance system," *International Journal of Distributed Sensor Networks*, vol. 9, no. 5, p. 686713, 2013.

[133] M. Hadded, P. Muhlethaler, A. Laouiti, and L. A. Saidane, "A centralized TDMA based scheduling algorithm for real-time communications in vehicular ad hoc networks," in *2016 24th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2016, pp. 1–6.

[134] R. S. Tomar, S. Verma, and G. S. Tomar, "Cluster based RSU centric channel access for VANETs," in *Transactions on Computational Science XVII.* Springer, 2013, pp. 150–171.

[135] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: a survey," *IEEE Communications Magazine*, vol. 51, no. 5, pp. 148–157, 2013.

[136] G. Nardini, A. Virdis, C. Campolo, A. Molinaro, and G. Stea, "Cellular-V2X communications for platooning: Design and evaluation," *Sensors*, vol. 18, no. 5, p. 1527, 2018.

[137] G. Cecchini, A. Bazzi, B. M. Masini, and A. Zanella, "Performance comparison between IEEE 802.11p and LTE-V2V in-coverage and out-of-coverage for cooperative awareness," in *2017 IEEE Vehicular Networking Conference (VNC)*, 2017, pp. 109–114.

[138] 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 802.11 p for vehicle-to-vehicle communications in highway platooning scenarios," *Ad Hoc Networks*, vol. 74, pp. 17–29, 2018.

[139] A. Bazzi, C. Campolo, A. Molinaro, A. O. Berthet, B. M. Masini, and A. Zanella, "On wireless blind spots in the C-V2X sidelink," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 9239–9243, 2020.

[140] Y. Yoon and H. Kim, "Resolving persistent packet collisions through broadcast feedback in cellular V2X communication," *Future Internet*, vol. 13, no. 8, p. 211, 2021.

[141] A. Bazzi, A. Zanella, G. Cecchini, and B. M. Masini, "Analytical investigation of two benchmark resource allocation algorithms for LTE-V2V," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5904–5916, 2019.

[142] L. Hu, J. Eichinger, M. Dillinger, M. Botsov, and D. Gozalvez, "Unified device-to-device communications for low-latency and high reliable vehicle-to-x services," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016, pp. 1–7.

[143] S. Zhang, Y. Hou, X. Xu, and X. Tao, "Resource allocation in D2D-based V2V communication for maximizing the number of concurrent transmissions," in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2016, pp. 1–6.

[144] L. F. Abanto-Leon, A. Koppelaar, and S. H. de Groot, “Parallel and successive resource allocation for V2V communications in overlapping clusters,” in *2017 IEEE Vehicular Networking Conference (VNC)*, 2017, pp. 223–230.

[145] ——, “Subchannel allocation for vehicle-to-vehicle broadcast communications in mode-3,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 2018, pp. 1–6.

[146] L. F. Abanto-Leon, A. Koppelaar, and S. Heemstra de Groot, “Network-assisted resource allocation with quality and conflict constraints for V2V communications,” in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[147] M. I. Ashraf, M. Bennis, C. Perfecto, and W. Saad, “Dynamic proximity-aware resource allocation in vehicle-to-vehicle (V2V) communications,” in *2016 IEEE Globecom Workshops (GC Wkshps)*, 2016, pp. 1–6.

[148] L. Liang, G. Y. Li, and W. Xu, “Resource allocation for D2D-enabled vehicular communications,” *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 3186–3197, 2017.

[149] F. Abbas and P. Fan, “A hybrid low-latency D2D resource allocation scheme based on cellular V2X networks,” in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018, pp. 1–6.

[150] C.-Y. Wei, A. C.-S. Huang, C.-Y. Chen, and J.-Y. Chen, “QoS-aware hybrid scheduling for geographical zone-based resource allocation in cellular vehicle-to-vehicle communications,” *IEEE Communications Letters*, vol. 22, no. 3, pp. 610–613, 2017.

[151] (2021) Breitbandmessung Funkloch-App. Publisher: zafaco GmbH im Auftrag der Bundesnetzagentur. [Online]. Available: https://breitbandmessung.de/

[152] C. M. Silva, B. M. Masini, G. Ferrari, and I. Thibault, “A survey on infrastructure-based vehicular networks,” *Mobile Information Systems*, vol. 2017, 2017.

[153] G. Hutton and C. Baker, “Rural mobile coverage in the UK: Not-spots and partial not-spots,” UK Parliament House of Commons Library, Research Briefing CBP 7069, Apr. 2022. [Online]. Available: https://commonslibrary.parliament.uk/research-briefings/sn07069/

[154] F. Rizzato. (2019, Jun.) Germany’s rural 4G users still spend one-fourth of their time on 3G and 2G networks. Opensignal. [Online]. Available: https://www.opensignal.com/blog/2019/06/13/germanys-rural-4g-users-still-spend-one-fourth-of-their-time-on-3g-and-2g-networks

[155] ——. (2021, Nov.) Germany 5G experience report november 2021. Opensignal. [Online]. Available: https://www.opensignal.com/reports/2021/11/germany/mobile-network-experience-5g

[156] Ready for a road trip? T-Mobile 5G covers (way) more interstate highway miles. T-Mobile Newsroom. [Online]. Available: https://www.t-mobile.com/news/network/ready-for-a-road-trip/

[157] F. Rizzato. (2022, Apr.) Analyzing how 5G has improved the mobile experience on Germany's motorways. Opensignal. [Online]. Available: https://www.opensignal.com/2022/04/07/analyzing-how-5g-has-improved-the-mobile-experience-on-germanys-motorways

[158] (2020) 5G Automotive Association virtual showcase highlights momentum behind C-V2X technology deployment for connected vehicles and smart cities in the U.S. [Online]. Available: http://5gaa.org/news/5g-automotive-association-virtual-showcase-highlights-momentum-behind-c-v2x

[159] (2021) 5G Automotive Association discusses the acceleration of 5G deployment on european roads at MCW barcelona 2021. [Online]. Available: http://5gaa.org/news/5g-automotive-association-discusses-the-acceleration-of-5g-deployment-on-european

[160] M. Mikami and H. Yoshino, "Field trial on 5G low latency radio communication system towards application to truck platooning," *IEICE Transactions on Communications*, vol. 102, no. 8, pp. 1447–1457, 2019.

[161] A. Hegde and A. Festag, "Mode switching strategies in cellular-V2X," *IFAC-Papers*, vol. 52, no. 8, pp. 81–86, 2019.

[162] ——, "Artery-C: An OMNeT++ based discrete event simulation framework for cellular V2X," in *Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 47–51.

[163] ——, "Mode switching performance in cellular-V2X," in *2020 IEEE Vehicular Networking Conference (VNC)*, 2020, pp. 1–8.

[164] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, Massachusetts: The MIT Press, 2018.

[165] A. Amini and A. Soleimany. (2020) 6.S191 Introduction to Deep Learning. Massachusetts Institute of Technology: MIT OpenCourseWare. [Online]. Available: https://ocw.mit.edu/courses/6-s191-introduction-to-deep-learning-january-iap-2020/

[166] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[167] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[168] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[169] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[170] D. Silver, “Lectures on reinforcement learning,” 2015. [Online]. Available: https://www.davidsilver.uk/teaching/

[171] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing Atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, 2013.

[172] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[173] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Harley, T. P. Lillicrap, D. Silver, and K. Kavukcuoglu, “Asynchronous methods for deep reinforcement learning,” in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ser. ICML’16, 2016.

[174] A. Juliani. (2016) Simple reinforcement learning with Tensorflow Part 8: Asynchronous Actor-Critic Agents (A3C). [Online]. Available: https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-8-asynchronous-actor-critic-agents-a3c-c88f72a5e9f2

[175] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y. Liang, and D. I. Kim, “Applications of deep reinforcement learning in communications and networking: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019.

[176] A. Feriani and E. Hossain, “Single and multi-agent deep reinforcement learning for AI-enabled wireless networks: A tutorial,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1226–1252, 2021.

[177] F. Tang, Y. Kawamoto, N. Kato, and J. Liu, “Future intelligent and secure vehicular network toward 6G: Machine-learning approaches,” *Proceedings of the IEEE*, vol. 108, no. 2, pp. 292–307, 2019.

[178] A. Mekrache, A. Bradai, E. Moulay, and S. Dawaliby, “Deep reinforcement learning techniques for vehicular networks: Recent advances and future trends towards 6G,” *Veh. Commun.*, vol. 33, no. C, jan 2022.

[179] K. Tan, D. Bremner, J. Le Kernec, L. Zhang, and M. Imran, “Machine learning in vehicular networking: an overview,” *Digital Communications and Networks*, 2021.

[180] M. Abbasi, A. Shahraki, M. J. Piran, and A. Taherkordi, “Deep reinforcement learning for QoS provisioning at the MAC layer: A survey,” *Engineering Applications of Artificial Intelligence*, vol. 102, p. 104234, 2021.

[181] H. Ye and G. Y. Li, “Deep reinforcement learning based distributed resource allocation for V2V broadcasting,” in *2018 14th International Wireless Communications Mobile Computing Conference (IWCMC)*, June 2018, pp. 440–445.

[182] ——, "Deep reinforcement learning for resource allocation in V2V communications," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6.

[183] H. Ye, G. Y. Li, and B. F. Juang, "Deep reinforcement learning based resource allocation for V2V communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163–3173, 2019.

[184] S. Bhadauria, Z. Shabbir, E. Roth-Mandutz, and G. Fischer, "QoS based deep reinforcement learning for V2X resource allocation," in *2020 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, 2020, pp. 1–6.

[185] L. Liang, H. Ye, and G. Y. Li, "Multi - agent reinforcement learning for spectrum sharing in vehicular networks," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2019, pp. 1–5.

[186] ——, "Spectrum sharing in vehicular networks based on multi-agent reinforcement learning," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2282–2292, 2019.

[187] R. Hu, X. Wang, Y. Su, and B. Yang, "An efficient deep reinforcement learning based distributed channel multiplexing framework for V2X communication networks," in *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 2021, pp. 154–160.

[188] Y.-H. Xu, C.-C. Yang, M. Hua, and W. Zhou, "Deep deterministic policy gradient (DDPG)-based resource allocation scheme for NOMA vehicular communications," *IEEE Access*, vol. 8, pp. 18 797–18 807, 2020.

[189] H. Yang, X. Xie, and M. Kadoch, "Intelligent resource management based on reinforcement learning for ultra-reliable and low-latency IoV communication networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4157–4169, 2019.

[190] D. Zhao, H. Qin, B. Song, Y. Zhang, X. Du, and M. Guizani, "A reinforcement learning method for joint mode selection and power adaptation in the V2V communication network in 5G," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 2, pp. 452–463, 2020.

[191] X. Zhang, M. Peng, S. Yan, and Y. Sun, "Deep-reinforcement-learning-based mode selection and resource allocation for cellular V2X communications," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6380–6391, 2020.

[192] M. Parvini, M. R. Javan, N. Mokari, B. A. Arand, and E. A. Jorswieck, "AoI aware radio resource management of autonomous platoons via multi agent reinforcement learning," in *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*, 2021, pp. 1–6.

[193] L. Wang, H. Ye, L. Liang, and G. Y. Li, "Learn to compress CSI and allocate resources in vehicular networks," *IEEE Transactions on Communications*, vol. 68, no. 6, pp. 3640–3653, 2020.

[194] H. Park and Y. Lim, "Deep reinforcement learning based resource allocation with radio remote head grouping and vehicle clustering in 5G vehicular networks," *Electronics*, vol. 10, no. 23, p. 3015, 2021.

[195] P. Xiang, H. Shan, M. Wang, Z. Xiang, and Z. Zhu, "Multi-agent RL enables decentralized spectrum access in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10 750–10 762, 2021.

[196] X. Chen, C. Wu, H. Zhang, Y. Zhang, M. Bennis, and H. Vuojala, "Decentralized deep reinforcement learning for delay-power tradeoff in vehicular communications," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

[197] X. Chen *et al.*, "Age of information aware radio resource management in vehicular networks: A proactive deep reinforcement learning perspective," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2268–2281, 2020.

[198] C. Wu, S. Ohzahata, Y. Ji, and T. Kato, "A MAC protocol for delay-sensitive VANET applications with self-learning contention scheme," in *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, 2014, pp. 438–443.

[199] A. Pressas, Z. Sheng, F. Ali, D. Tian, and M. Nekovee, "Contention-based learning MAC protocol for broadcast vehicle-to-vehicle communication," in *2017 IEEE Vehicular Networking Conference (VNC)*, Nov 2017, pp. 263–270.

[200] A. Pressas, Z. Sheng, F. Ali, and D. Tian, "A Q-learning approach with collective contention estimation for bandwidth-efficient and fair access control in IEEE 802.11 p vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 9136–9150, 2019.

[201] D.-j. Lee, Y. Deng, and Y.-J. Choi, "Back-off improvement by using Q-learning in IEEE 802.11 p vehicular network," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, 2020, pp. 1819–1821.

[202] C. Choe, J. Choi, J. Ahn, D. Park, and S. Ahn, "Multiple channel access using deep reinforcement learning for congested vehicular networks," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–6.

[203] R. Pal, N. Gupta, A. Prakash, R. Tripathi, and J. J. Rodrigues, "Deep reinforcement learning based optimal channel selection for cognitive radio vehicular ad-hoc network," *IET Communications*, vol. 14, no. 19, pp. 3464–3471, 2020.

[204] R. F. Atallah, C. M. Assi, and J. Y. Yu, "A reinforcement learning technique for optimizing downlink scheduling in an energy-limited vehicular network," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4592–4601, 2017.

[205] S. Park and Y. Yoo, "Real-time scheduling using reinforcement learning technique for the connected vehicles," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[206] Y. Xia, L. Wu, Z. Wang, X. Zheng, and J. Jin, "Cluster-enabled cooperative scheduling based on reinforcement learning for high-mobility vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12 664–12 678, 2020.

[207] R. Atallah, C. Assi, and M. Khabbaz, “Deep reinforcement learning-based scheduling for roadside communication networks,” in *2017 15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2017, pp. 1–8.

[208] R. F. Atallah, C. M. Assi, and M. J. Khabbaz, “Scheduling the operation of a connected vehicular network using deep reinforcement learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1669–1682, 2019.

[209] Y. Zhou, F. Tang, Y. Kawamoto, and N. Kato, “Reinforcement learning-based radio resource control in 5G vehicular network,” *IEEE Wireless Communications Letters*, vol. 9, no. 5, pp. 611–614, 2019.

[210] F. Tang, Y. Zhou, and N. Kato, “Deep reinforcement learning for dynamic uplink/downlink resource allocation in high mobility 5G HetNet,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 12, pp. 2773–2782, 2020.

[211] M. F. Pervej and S.-C. Lin, “Dynamic power allocation and virtual cell formation for throughput-optimal vehicular edge networks in highway transportation,” in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–7.

[212] ——, “Eco-vehicular edge networks for connected transportation: A distributed multi-agent reinforcement learning approach,” in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, 2020, pp. 1–7.

[213] Y. Liu, Z. Jiang, S. Zhang, and S. Xu, “Deep reinforcement learning-based beam tracking for low-latency services in vehicular networks,” in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.

[214] Y. Su, M. LiWang, Z. Gao, L. Huang, S. Liu, and X. Du, “Coexistence of cellular V2X and Wi-Fi over unlicensed spectrum with reinforcement learning,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[215] Y. Xu, L. Li, B. Soong, and C. Li, “Fuzzy Q-learning based vertical handoff control for vehicular heterogeneous wireless network,” in *2014 IEEE International Conference on Communications (ICC)*, 2014, pp. 5653–5658.

[216] Z. Li, C. Wang, and C. Jiang, “User association for load balancing in vehicular networks: An online reinforcement learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 2217–2228, 2017.

[217] H. Khan, A. Elgabli, S. Samarakoon, M. Bennis, and C. S. Hong, “Reinforcement learning-based vehicle-cell association algorithm for highly mobile millimeter wave communication,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 4, pp. 1073–1085, 2019.

[218] M. A. Salahuddin, A. Al-Fuqaha, and M. Guizani, “Reinforcement learning for resource provisioning in the vehicular cloud,” *IEEE Wireless Communications*, vol. 23, no. 4, pp. 128–135, August 2016.

[219] Y. He, C. Liang, Z. Zhang, F. R. Yu, N. Zhao, H. Yin, and Y. Zhang, "Resource allocation in software-defined and information-centric vehicular networks with mobile edge computing," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–5.

[220] Y. He, F. R. Yu, N. Zhao, H. Yin, and A. Boukerche, "Deep reinforcement learning (DRL)-based resource management in software-defined and virtualized vehicular ad hoc networks," in *Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*, 2017, pp. 47–54.

[221] Y. He, N. Zhao, and H. Yin, "Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 44–55, 2018.

[222] Y. He, F. R. Yu, N. Zhao, V. C. Leung, and H. Yin, "Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 31–37, 2017.

[223] Y. He, Y. Wang, Q. Lin, and J. Li, "Meta-hierarchical reinforcement learning (MHRL)-based dynamic resource allocation for dynamic vehicular networks," *IEEE Transactions on Vehicular Technology*, 2022.

[224] M. Chen, T. Wang, K. Ota, M. Dong, M. Zhao, and A. Liu, "Intelligent resource allocation management for vehicles network: An A3C learning approach," *Computer Communications*, vol. 151, pp. 485–494, 2020.

[225] Q. Zheng, K. Zheng, H. Zhang, and V. C. M. Leung, "Delay-optimal virtualized radio resource scheduling in software-defined vehicular networks via stochastic learning," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 7857–7867, Oct 2016.

[226] Z. Lyu, Y. Wang, M. Liu, and Y. Chen, "Service-driven resource management in vehicular networks based on deep reinforcement learning," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, 2020, pp. 1–6.

[227] L. Lei, Y. Tan, K. Zheng, S. Liu, K. Zhang, and X. Shen, "Deep reinforcement learning for autonomous internet of things: Model, applications and challenges," *IEEE Communications Surveys Tutorials*, vol. 22, no. 3, pp. 1722–1760, 2020.

[228] Y. L. Lee and D. Qin, "A survey on applications of deep reinforcement learning in resource management for 5G heterogeneous networks," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 1856–1862.

[229] Y. Huang, C. Xu, C. Zhang, M. Hua, and Z. Zhang, "An overview of intelligent wireless communications using deep reinforcement learning," *Journal of Communications and Information Networks*, vol. 4, no. 2, pp. 15–29, 2019.

[230] P. V. R. Ferreira, R. Paffenroth, A. M. Wyglinski, T. M. Hackett, S. G. Bilen, R. C. Reinhart, and D. J. Mortensen, “Reinforcement learning for satellite communications: From LEO to deep space operations,” *IEEE Communications Magazine*, vol. 57, no. 5, pp. 70–75, 2019.

[231] H. Ye, L. Liang, G. Y. 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, June 2018.

[232] L. Liang, H. Ye, and G. Y. Li, “Toward intelligent vehicular networks: A machine learning framework,” *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 124–135, 2019.

[233] L. Liang, H. Ye, G. Yu, and G. Y. Li, “Deep-learning-based wireless resource allocation with application to vehicular networks,” *Proceedings of the IEEE*, vol. 108, no. 2, pp. 341–356, 2020.

[234] E. Lee, E.-K. Lee, M. Gerla, and S. Y. Oh, “Vehicular cloud networking: architecture and design principles,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 148–155, 2014.

[235] F. J. Ros, J. A. Martinez, and P. M. Ruiz, “A survey on modeling and simulation of vehicular networks: Communications, mobility, and tools,” *Computer Communications*, vol. 43, pp. 1–15, 2014.

[236] 3GPP, “Study on LTE-based V2X services,” 3GPP, Tech. Rep. TR 36.885, June 2016, v14.0.0.

[237] R. Molina-Masegosa, J. Gozalvez, and M. Sepulcre, “Comparison of IEEE 802.11p and LTE-V2X: An evaluation with periodic and aperiodic messages of constant and variable size,” *IEEE Access*, vol. 8, pp. 121 526–121 548, 2020.

[238] G. P. Wijesiri N.B.A., J. Haapola, and T. Samarasinghe, “A discrete-time markov chain based comparison of the MAC layer performance of C-V2X mode 4 and IEEE 802.11p,” *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2505–2517, 2021.

[239] A. Vogt and J. Bared, “Accident models for two-lane rural segments and intersections,” *Transportation Research Record*, vol. 1635, no. 1, pp. 18–29, 1998.

[240] European Commission and Directorate-General for the Information Society and Media, *COST Action 231 : Digital mobile radio towards future generation systems: Final Report*. Publications Office, 1999.

[241] M. Mezzavilla, M. Miozzo, M. Rossi, N. Baldo, and M. Zorzi, “A lightweight and accurate link abstraction model for the simulation of LTE networks in ns-3,” in *Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM ’12. New York, NY, USA: Association for Computing Machinery, 2012, p. 55–60.

[242] J. Meinilä, P. Kyösti, L. Hentilä, T. Jämsä, E. Suikkanen, E. Kunnari, and M. Narandžić, “D5.3: WINNER+ final channel models,” Tech. Rep., 2010.

[243] J. Wang and R. A. Rouil, "BLER performance evaluation of LTE device-to-device communications," NIST Interagency/Internal Report (NISTIR) 8157, Nov. 2016. [Online]. Available: https://www.nist.gov/publications/bler-performance-evaluation-lte-device-device-communications

[244] Z. Ding, I. Krikidis, B. Rong, J. S. Thompson, C. Wang, and S. Yang, "On combating the half-duplex constraint in modern cooperative networks: protocols and techniques," *IEEE Wireless Communications*, vol. 19, no. 6, pp. 20–27, 2012.

[245] S. Krauß, "Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics," Dr. rer. nat. dissertation, Universität zu Köln, 1998.

[246] J. Erdmann, "Lane-changing model in SUMO," *Proceedings of the SUMO2014 Modeling Mobility with Open Data*, vol. 24, pp. 77–88, 2014.

[247] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. WieBner, "Microscopic traffic simulation using SUMO," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2575–2582.

[248] P. Pak-Poy, "The use and limitation of the poisson distribution in road traffic," in *Australian Road Research Board (ARRB) Conference, 2nd, 1964, Melbourne*, vol. 2, no. 1, 1964.

[249] D. L. Gerlough and A. Schuhl, "Use of poisson distribution in highway traffic. The probability theory applied to distribution of vehicles on two-lane highways," 1955.

[250] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[251] 3GPP, "Study on evaluation methodology of new Vehicle-to-Everything (V2X) use cases for LTE and NR," 3GPP, Tech. Rep. TR 37.885, December 2018, v15.2.0.

[252] M. Boban and P. M. d'Orey, "Measurement-based evaluation of cooperative awareness for V2V and V2I communication," in *2014 IEEE Vehicular Networking Conference (VNC)*, 2014, pp. 1–8.

[253] 3GPP, "Study on enhancement of 3GPP support for 5G V2X services," 3GPP, Tech. Rep. TR 22.886, December 2018, v16.2.0.

[254] R. Van Der Horst and J. Hogema, "Time-to-collision and collision avoidance systems," in *Safety Evaluation of Traffic Systems: Traffic Conflicts and Other Measures, 6th International Cooperation on Theories and Concepts in Traffic Safety (ICTCT) Workshop Proceedings*. Kuratorium für Verkehrssicherheit, 1993, pp. 109–121.

[255] M. E. Renda, G. Resta, P. Santi, F. Martelli, and A. Franchini, "IEEE 802.11p VANets: Experimental evaluation of packet inter-reception time," *Computer Communications*, vol. 75, pp. 26–38, 2016.

[256] M. Sepulcre and J. Gozalvez, "Experimental evaluation of cooperative active safety applications based on V2V communications," in *Proceedings of the Ninth ACM International Workshop on Vehicular Inter-Networking, Systems, and Applications*, ser. VANET '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 13–20.

[257] N. An, T. Gaugel, and H. Hartenstein, "VANET: Is 95% probability of packet reception safe?" in *2011 11th International Conference on ITS Telecommunications*, 2011, pp. 113–119.

[258] A. Varga, "OMNeT++," in *Modeling and Tools for Network Simulation*. Springer, 2010, pp. 35–59.

[259] (2017) ns-3 network simulator. nsnam. [Online]. Available: http://www.nsnam.org

[260] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, 2011.

[261] R. Riebl, H.-J. Günther, C. Facchi, and L. Wolf, "Artery: Extending Veins for VANET applications," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 2015, pp. 450–456.

[262] M. Rondinone, J. Maneros, D. Krajzewicz, R. Bauza, P. Cataldi, F. Hrizi, J. Gozalvez, V. Kumar, M. Röckl, L. Lin *et al.*, "iTETRIS: A modular simulation platform for the large scale evaluation of cooperative ITS applications," *Simulation Modelling Practice and Theory*, vol. 34, pp. 99–125, 2013.

[263] B. Schünemann, "V2X simulation runtime infrastructure VSimRTI: An assessment tool to design smart traffic management systems," *Computer Networks*, vol. 55, no. 14, pp. 3189–3198, 2011.

[264] A. Virdis, G. Stea, and G. Nardini, "SimuLTE-A modular system-level simulator for LTE/LTE-A networks based on OMNeT++," in *2014 4th International Conference On Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*, 2014, pp. 59–70.

[265] G. Piro, N. Baldo, and M. Miozzo, "An LTE module for the ns-3 network simulator," in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*. ICST, 2011, pp. 415–422.

[266] R. Rouil, F. J. Cintrón, A. Ben Mosbah, and S. Gamboa, "Implementation and validation of an LTE D2D model for ns-3," in *Proceedings of the 2017 Workshop on Ns-3*, ser. WNS3 '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 55–62.

[267] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[268] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.

[269] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "OpenAI Gym," *arXiv preprint arXiv:1606.01540*, 2016.

[270] P. Gawłowicz and A. Zubow, "Ns-3 meets OpenAI gym: The playground for machine learning in networking research," in *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2019, pp. 113–120.

[271] The MathWorks Inc., "MATLAB version: 9.3.0.713579 (r2017b)," Natick, Massachusetts, United States, 2022. [Online]. Available: https://www.mathworks.com

[272] National Institute of Standards and Technology, "LTE device to device communication model for ns-3 v0.1 (06/13/2017)," 2017. [Online]. Available: https://github.com/usnistgov/psc-ns3/tree/d2d-ns-3.22

[273] "ns-3.22," nsnam, 2015. [Online]. Available: https://www.nsnam.org/releases/ns-3-22/

[274] (2021) editModesNetwork - SUMO documentation. DLR. [Online]. Available: https://sumo.dlr.de/docs/Netedit/editModesNetwork.html

[275] (2021) Definition of vehicles, vehicle types, and routes - SUMO documentation. DLR. [Online]. Available: https://sumo.dlr.de/docs/Definition_of_Vehicles,_Vehicle_Types,_and_Routes.html

[276] (2021) TraceExporter - SUMO documentation. DLR. [Online]. Available: https://sumo.dlr.de/docs/Tools/TraceExporter.html

[277] (2014) ns3::Ns2MobilityHelper class reference. nsnam. [Online]. Available: https://www.nsnam.org/docs/release/3.19/doxygen/classns3_1_1_ns2_mobility_helper.html

[278] G. Van Rossum and F. L. Drake Jr, *Python reference manual.* Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[279] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

[280] H. Mao, "Pensieve," 2017, Neural adaptive video streaming with Pensieve (SIGCOMM '17), web.mit.edu/pensieve/. [Online]. Available: https://github.com/hongzimao/pensieve

[281] S. Verdu, "Spectral efficiency in the wideband regime," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1319–1343, Jun 2002.

[282] 3GPP, *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception*, 3GPP Std. TS 36.101, Dec. 2019, v16.4.0.

[283] C. Zhao, O. Siguad, F. Stulp, and T. M. Hospedales, “Investigating generalisation in continuous deep reinforcement learning,” *arXiv preprint arXiv:1902.07015*, 2019.

[284] S. Gamrian and Y. Goldberg, “Transfer learning for related reinforcement learning tasks via image-to-image translation,” in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 2019, pp. 2063–2072.

[285] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[286] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, “Mastering the game of Go without human knowledge,” *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[287] P. Wendland, G. Schaefer, and R. Thomä, “An application-oriented evaluation of LTE-V’s mode 4 for V2V communication,” in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 2019, pp. 165–173.

[288] ETSI TC ITS, *Intelligent Transport Systems; Users and applications requirements; Part 2: Applications and facilities layer common data dictionary*, Std. ETSI EN Std 102 894-2 V1.2.1, 2014.

[289] S. Maghsudi and S. Stańczak, “Hybrid centralized–distributed resource allocation for device-to-device communication underlaying cellular networks,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2481–2495, 2015.

[290] Z. Zhou, M. Dong, Z. Chang, and B. Gu, “Combined centralized and distributed resource allocation for green D2D communications,” in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, 2015, pp. 1–6.

[291] X. Li, R. Shankaran, M. Orgun, L. Ma, and Y. Xu, “Joint autonomous resource selection and scheduled resource allocation for D2D-based V2X communication,” in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[292] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, “Survey of cellular mobile radio localization methods: From 1G to 5G,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1124–1148, 2018.

[293] H. Mao, R. Netravali, and M. Alizadeh, “Neural adaptive video streaming with Pensieve,” in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM ’17. New York, NY, USA: Association for Computing Machinery, 2017, p. 197–210.

[294] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, “Deep sets,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.