Utilizing Reinforcement Learning for Adaptive Sensor Data Sharing over C-V2X Communications

Bryse Flowers (*Student Member, IEEE*), Yu-Jen Ku (*Student Member, IEEE*),
Sabur Baidya (*Member, IEEE*), and Sujit Dey (*Fellow, IEEE*)

***Abstract*—Vehicular networking has seen continued evolution over decades with the recently emerging paradigm of Cellular Vehicle-to-Everything (C-V2X) communications beginning to pick up momentum for adoption on today's roadways. Initial iterations of C-V2X grew from the LTE Device-to-Device framework and targeted application use cases that required the exchange of small packets of information: where a vehicle is, what it is doing, etc. Many of the next generation of use cases require the transfer of sensory data from vehicles to the edge, e.g., tele-driving, cooperative perception, computation offloading, etc. This work evaluates whether today's commercially available vehicular networking solutions can support the higher data rates required to carry this sensory data from vehicles to a Roadside Unit using a C-V2X testbed based on C-V2X Mode 4, which operates autonomously in shared spectrum. It is experimentally shown that C-V2X is capable of carrying the most common form of vehicle sensor data, images, with a frame latency of approximately 50 ms; however, these transmissions are often unreliable due to C-V2X Mode 4's lack of adaptation capability. To mitigate this, a Reinforcement Learning (RL) problem is proposed that can adapt the transmission parameters of image frames using readily available out of band information in order to achieve a 23.3% relative improvement in effective throughput. Extensions to this RL problem are developed that allow explicit control over a desired risk tolerance, such as the probability of transmitting an image but not receiving it. Using this extension, the developed RL solution learns an adaptive transmission policy that successfully delivers 87.1% of the image frames which are transmitted (a 33.6% absolute improvement) while still maintaining the throughput advantage. Ultimately this work finds that today's commercial vehicular networking solutions are capable of supporting applications that require sensor data sharing by using RL to overcome the limitations of C-V2X Mode 4.**

***Index Terms*—C-V2X, Reinforcement Learning, Testbed, Wireless Communications**

I. Introduction

Vehicle-to-Everything (V2X) technologies have continually evolved over decades from Dedicated Short Range Communications (DSRC) to current Cellular V2X (C-V2X), which has been shown to provide increased reliability over DSRC [2]. The advent of V2X communications has enabled vehicles to share their location and trajectories with other vehicles on the road in the form of Cooperative Awareness Messages (CAMs) [3] or Basic Safety Messages (BSMs) [4] - providing a path to improved roadway safety and efficiency. While these messages are relatively small, many next-generation use cases target sensor data sharing, significantly increasing necessary data rates. C-V2X initially grew from the LTE Device-to-Device framework and utilizes the PC5 Sidelink interface. Operation of these radios in Mode 4 [5] allows for distributed resource selection by each vehicle using a sensing-based approach without needing any aid from the network. The drawback of this approach is that the technology is primarily targeted for broadcast messages with no built-in link adaptation methodology to enable reliable unicast transmissions - perfect for CAMs and BSMs but not ideal for sensor sharing.

This work asks *can today's commercially available C-V2X radios support sensor data sharing?* The insights gained in this work by exploring this question will undoubtedly carry over to future generations of C-V2X [6], where inclusions of re-transmission and channel state feedback protocols will improve C-V2X unicast but do not fully capture the breadth of adaptations explored in this work for real-time sharing of sensor data. To study this, a C-V2X testbed is deployed on the University of California, San Diego (UCSD) campus, which consists of a standalone Road-Side Unit (RSU). It is shown that *C-V2X Mode 4 can support transmission of JPEG compressed image frames from vehicle to the roadside edge with a typical frame latency around 50 ms* (depending on the level of compression used). However, transmission reliability remains challenging due to the larger message sizes inherent to sensor data over smaller single-packet transmissions, as even small-scale packet loss can be dramatically amplified. This work shows that this reliability challenge can be overcome through the utilization of Reinforcement Learning (RL) to autonomously adapt the transmission of images from vehicles using available out-of-band information (due to the lack of feedback in C-V2X Mode 4). Moreover, State-of-the-Art (SOTA) RL methodology is extended to allow for explicitly setting a reliability constraint to force high Frame Delivery Rates (FDRs) when transmission is attempted. This methodology enables C-V2X Mode 4 to achieve a FDR of 87.1% while achieving an 11.8% higher goodput than a static configuration of always transmitting high-quality images - *C-V2X Mode 4 can be used for reliable transmission of image frames while maintaining high image quality*. The proposed RL methodology is developed and demonstrated with com-

This work is supported in part by the Center for Wireless Communications at University of California San Diego, the Smart Transportation Innovation Program, and Qualcomm Technologies Inc.

Portions of this work appeared in [1] where the testbed (Section II) and preliminary analysis of the collected dataset (Section III) were presented.

The authors are with the Center for Wireless Communications, University of California San Diego 92093 USA (e-mail: {bflowers, yuku, dey}@ucsd.edu) and University of Louisville, Kentucky 40292 USA (email: sabur.baidya@louisville.edu)

mercial C-V2X Hardware-in-the-Loop (HIL), showing that the methodology is immediately applicable to today's roadways.

A. Related Work

Millimeter Wave (mmWave) V2X. It can be tempting to use wider spectrum, such as mmWave to support needed rates for real-time transfer of high volume data. Indeed, multiple works have proposed and studied mmWave V2X [7], [8] and shown that one of the most difficult tasks, beam management in mobility, could be overcome through the incorporation of data driven techniques [9]–[12]. While these studies are surely important for the long-term, mmWave is unlikely to see commercial success for V2X in the near-term. This work focuses on augmenting the version of C-V2X that is commercially available today using the relatively narrow bandwidth of the 5.9GHz Intelligent Transportation Systems (ITS) band [13] where prior studies have only been conducted in simulation [14]–[16]. Although the capabilities of C-V2X will undeniably be more limited than those of mmWave, this work is more directly applicable to roadways today and therefore can help the community iteratively evolve to the next generation of use cases before mmWave V2X could become commonplace.

Application Specific Studies. Many prior works have conducted algorithmic studies for specific applications that would utilize vehicular sensor data. For instance, how to fuse detections from multiple cameras [17]–[19] or lidars [20]–[22]. Or how to partition computing tasks between vehicles, edge, and cloud nodes [23], [24]. While these works provide evidence of the utility of sharing vehicular sensor data, they often ignore or abstract the V2X communications necessary to support these applications; thus, as this work focuses primarily on the optimization of the wireless transmission of sensor data, it can be considered orthogonal to these algorithmic studies. A likely fruitful research thrust would be an exploration that jointly considers the application and the wireless transmissions, e.g., [25] considers selective transmission of detected roadway objects to conserve bandwidth using knowledge of the contents of sensor data. Some other works [26], [27] proposed vehicular task offloading to edge computing node with communication scheduling and resource allocation, to minimize the vehicular task delay. Ultimately, these areas are, at present, understudied in realistic settings, and this work will be complementary to these ongoing efforts.

Adaptive Bit Rate (ABR) Algorithms. Perhaps the closest comparison to this work would be the numerous studies of adaptive video streaming which seek to optimize the trade space between video quality and the probability of rebuffering. One key distinguishing factor of this work is that the C-V2X Sidelinks utilized are only single-hop connections and therefore have different characteristics than the multi-hop connections more widely studied where network congestion and queuing delays are primary concerns as opposed to this work whose primary concern is the vehicular wireless link, which changes rapidly and therefore conventional ABR algorithms cannot adapt quickly enough. Further, ABR algorithms typically use estimates of 1) throughput [28], [29], 2) buffer status [30], or 3) both [31], [32], which are not available in this work due to the blind broadcast nature of C-V2X Sidelinks and real time nature of the sensor data sharing task. The closest parallel in adaptive video transmission to this work is the study of using physical layer metrics to inform video transmission [33]. This work differs from that study in three ways. First, the necessity of real-time image transmission; thus, there can be no buffering of data and re-transmissions are likely to make the transmission have excessive latency. Second, the air interfaces used in this work are different, utilizing PC5 on Sidelink instead of the typical Uu interface of cellular networks with a proper Base Station. Third, instead of utilizing metrics from the physical layer, which are unfortunately unavailable in C-V2X Mode 4 due to the lack of feedback in the protocol, this work utilizes *out of band* information to adapt transmissions.

B. Research Contributions & Paper Outline

This is the first work to experimentally explore the repurposing of today's commercially available C-V2X radios for the higher data rate transmissions of vehicular sensor data sharing. Using image data as a use case study, this work characterizes the latency and reliability of these transmissions in a realistic setting. Further, a RL problem is proposed to overcome the lack of adaptation capabilities in C-V2X Mode 4 which is shown to greatly improve the effective throughput while satisfying a configurable reliability constraint for FDR in the optimization problem. More specifically, the remainder of this work is organized as follows. This work first describes the system setup in Section II before presenting the principle research contributions within Sections III, IV, and V:

1) **Section III.** Characterization of the performance of currently commercialized C-V2X radios for sharing image data from vehicle to the roadside edge in a realistic deployment scenario, through utilization of the UCSD C-V2X testbed, in order to show that C-V2X Sidelinks operating in Mode 4 can, in fact, support the transmission of image frames with sub-100ms frame latency and high reliability within some subsets of the roadway study area.
2) **Section IV.** Investigation of the utility of RL for augmenting the capabilities of C-V2X by adapting WAVE Short Message Protocol (WSMP) Packet Size and JPEG Quality Level according to the vehicle location and recently observed background traffic as an estimate of spectrum congestion; the optimization of the controlling Agent is implemented as HIL training which showcases the ability for life-long learning to adapt to new deployments and how RL methodologies can robustly apply to noisy real world wireless environments.
3) **Section V.** Extension of SOTA RL methodologies to allow network operators explicit control over Quality of Service (QoS) constraints, such as FDR; this methodology is shown to automatically tune a multi-objective reward function to balance risk (i.e. frame loss) with reward (i.e. application *goodput*) and achieves a 23.3% improvement in goodput when the FDR constraint is relaxed while still maintaining a 11.8% goodput advantage when targeting a 90% FDR. This shows that transmissions over C-V2X Mode 4 can not only be optimized

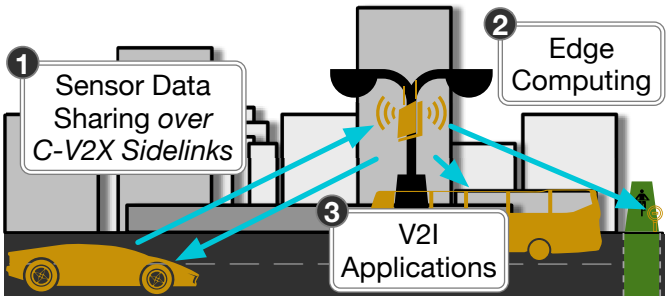


Fig. 1: Model of sharing vehicular sensor data to an RSU; this work is only concerned with Step 1, or the sharing of sensor data over C-V2X Sidelinks.

for goodput, but are capable of achieving highly reliable image transmissions while doing so.

The paper then concludes with ablation studies in Section VI and a summary of findings in Section VII.

II. Background

This section begins with a discussion of the system setup for this work. It then describes the prototype testbed of that system on the UCSD campus that enables the studies performed and, finally, a short description of how image transmission can be realized over C-V2X Sidelinks which were not designed for large scale data transfer.

A. System Setup

Initial applications of V2X primarily focused on vehicles sharing *where, or who, they are* (e.g., an ambulance approaches the intersection) and *what they are doing* (e.g., emergency braking). This work is interested in exploring the next logical step, enabling vehicles to share *what they see*, which can be useful for myriad applications. More specifically, this work explores how to transfer vehicle sensor data to the RSU in a low-latency and reliable fashion, as is shown in Step 1 of Fig. 1, but is agnostic to how that data is processed at the edge or used/disseminated back to roadway users. Image data is taken as a specific use case study for performance evaluation, yet, the methodology could be easily extended to other modalities in future work. Additionally, while the adoption of mmWave or more traditional cellular links in the form of Vehicle-to-Network (V2N) could increase the link budget, this work chooses to explore C-V2X Mode 4. C-V2X Mode 4 operates in a shared spectrum dedicated to ITS and only uses single-hop connections. Therefore, it has the potential for interoperability not tied to a specific network operator (as would be the case in V2N) and is also currently commercialized (as opposed to mmWave V2X which is so-far limited to academic studies). While C-V2X Mode 4 has these advantages, it relies upon autonomous resource allocation that necessitates configuration of transmission parameters by the On-Board Units (OBUs) within vehicles - how to configure these parameters, given the information that is likely to be available to each vehicle, is the problem studied in this work.

B. Deploying a C-V2X Testbed on UCSD's Campus

This work leverages a real deployment of C-V2X radios on the UCSD campus that can provide a close representation of what will be seen in commercial networks. Fig. 2 provides a visual overview of this deployment environment. The UCSD C-V2X testbed consists of two major components: 1) a green computing and communications enabled RSU node deployed on a campus street lamp and 2) OBUs deployed into a fleet of research vehicles. The RSU consists of a Commsignia RSU kit [34] enclosed in a weatherproof box mounted at pedestrian height on the pole. The antennas, for both C-V2X and GPS, are mounted at the maximum allowable height by the Federal Communications Commission: 8m. Each of the antennas is oriented parallel to the lamp post. Both the RSU and OBU contain the same underlying cellular modem, which is a Qualcomm C-V2X 9150 [35] radio running the 3GPP Release-14 C-V2X Standard[1]. The RSU consists of a co-located NVIDIA Jetson TX2 [36] device for edge computing and it is driven by solar energy. Additionally, the RSU is accompanied by an LTE enabled router that provides remote command and control functionality.

The OBU consists of a Commsignia OBU kit [37] installed into a research vehicle. As mentioned previously, the OBU utilizes the same Qualcomm C-V2X 9150 radio as the RSU, however, the antennas are neatly combined into a single module that is easily magnetically mounted to the roof of the vehicle. The OBU was primarily interacted with via a laptop over Ethernet, but the Commsignia OBU kit can also function as a WiFi hotspot that allows for a tablet to connect and display real-time radio status during operation.

The RSU is deployed on Voigt Drive, a road within the UCSD campus, as is shown in Fig. 2. The study area chosen for this work consists of the 100m of road to the immediate east and west of the RSU. Although Section III will show that usable range may potentially extend beyond this (at least for small data sizes), the inter-site spacing for User Equipment type RSUs is assumed to be 100m or the center of intersections for freeway and urban use cases respectively as defined by 3GPP TR 36.885 [38]; thus, this study area more than covers the expected range that is likely to be encountered in commercial deployments.

C. Realizing Image Transmission over C-V2X Sidelinks

The process for image transmission using C-V2X Sidelinks is shown in Fig. 3. Each image must first be serialized to a byte array; without loss of generality[2], this work chooses to utilize the well-known JPEG encoding that enables scaling compression rates by changing a *quality level* that is defined between 0 and 95. The image payload is then too large to fit into a single WSMP packet, which is limited to 1390 by the Commmsignia Application Programmer Interface (API),

[1]While there are 5G specifications recently made available for C-V2X that offer enhanced functionality we were not aware of any commercial offering for these 5G C-V2X devices when deploying the UCSD C-V2X testbed.

[2]The methodology presented in this work is independent of the utilization of JPEG, as the RL algorithms presented could learn to utilize other encoding techniques (or even other data modalities). This exploration is left to future work.

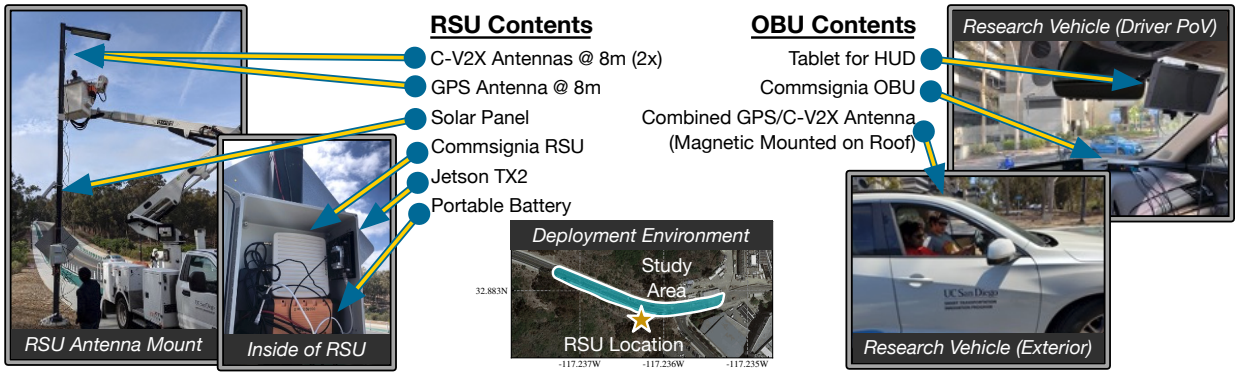


Fig. 2: Overview of the deployment of a C-V2X network on the UCSD campus.

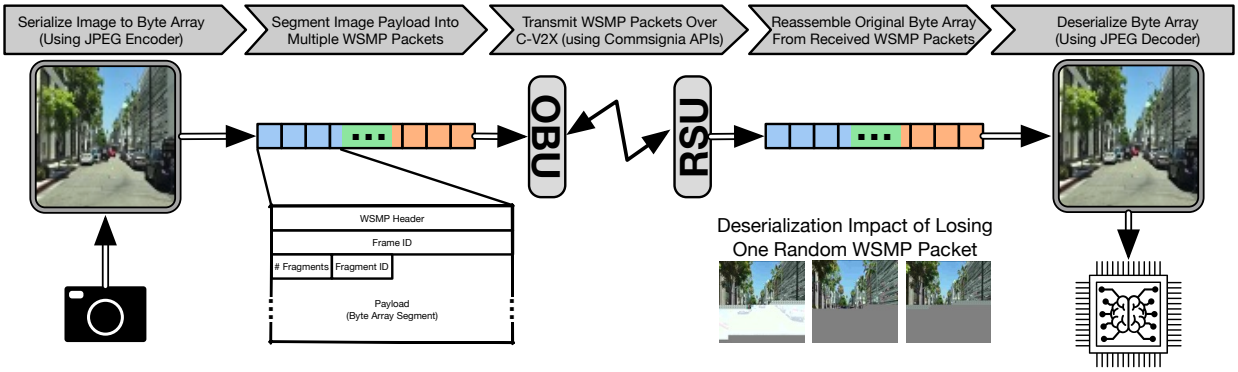


Fig. 3: Flowgraph of image transmission using C-V2X Sidelinks.

therefore, the byte array is segmented into multiple WSMP packets with a specified payload size. Each of these WSMP packets are then transmitted from the OBU to the RSU with zero inter-packet arrival time. At the RSU, each of the WSMP packets containing a fragment of the image payload are reconstructed using meta-data included in each WSMP packet. As shown in Fig. 3, losing even a single WSMP packet can severely degrade the image quality. While it is feasible that Deep Learning models could learn to be robust to packet loss, this work assumes that an image frame is lost if even a single WSMP packet is lost in order to decouple the study of optimizing the wireless communications link and the study of the algorithms employed at the RSU to process the data.

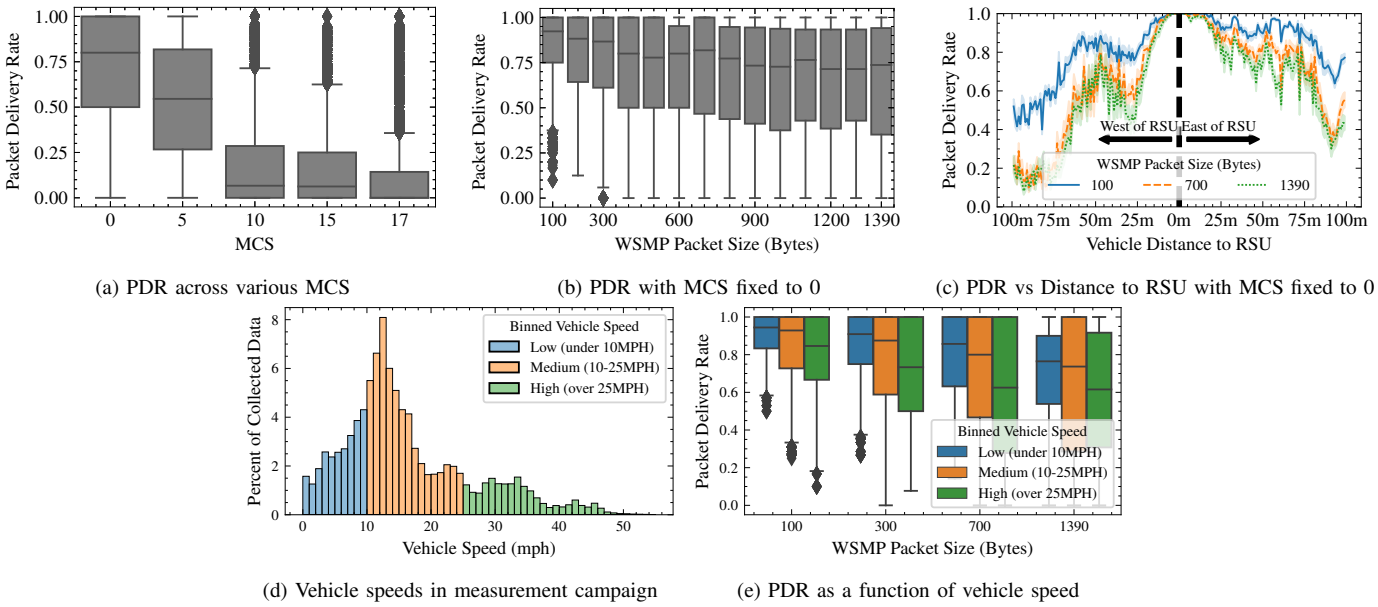
III. Characterizing C-V2X Capabilities for Sharing Vehicular Camera Data

In order to characterize the capabilities of C-V2X for sharing vehicular sensor data in a realistic setting, this work first presents the results of a measurement campaign to capture the WSMP packet level QoS characteristics observed on the UCSD C-V2X testbed. The WSMP packet level QoS can then be transformed into estimates of image frame level QoS metrics through simple equations. Finally, while Packet Delivery Rate (PDR) is primarily a function of the wireless channel, the characterization of latency is dominated by the prevalence of background traffic and therefore this section concludes with a study of frame latency in the presence of varying levels of background traffic.

A. Real World Measurement Campaign

One of the most critical metrics for vehicular communications is a measure of its reliability, which can be characterized by PDR, or the ratio of packets that are successfully received. As with all wireless communications, the PDR will be primarily driven by two dynamic factors: 1) the effective link Signal-to-Noise Ratio (SNR) and 2) the chosen MCS. The former depends on many factors, such as the propagation environment, but, loosely speaking, is a function of distance between the transmitter and receiver under the assumption of a fixed transmit power (that is assumed in this work to be 20dBm). The latter is typically a carefully selected parameter based on a feedback loop from the receiver that provides estimations of the channel state information - a feedback loop that is not available in C-V2X Mode 4 due to its autonomous operation. The MCS in the C-V2X radios used in this work would typically be selected as a function of packet size (i.e. use the minimum MCS that allows for the entire WSMP packet to fit into the transmit opportunity, with the mapping based on Table 7.1.7.2.1-1 in 3GPP TS 36.213 [39]); however, this section explicitly restricts the modem configuration to a single MCS for the purposes of studying the impact of MCS on PDR.

To determine the PDR, the research vehicle (described in Section II) is driven along the roadway of the study area using varying radio configurations and driving speeds. A range of MCS are selected from $\{0, 5, 10, 15, 17\}$ where 0 and 17 are the minimum and maximum values respectively available from Commsignia's APIs. The packet sizes range from 100B to 1390B in increments of 100B, with 1390B being the maximum



(a) PDR across various MCS

(b) PDR with MCS fixed to 0

(c) PDR vs Distance to RSU with MCS fixed to 0

(d) Vehicle speeds in measurement campaign

(e) PDR as a function of vehicle speed

Fig. 4: WSMP packet level QoS characterization from measurement campaign on the UCSD C-V2X testbed. As expected, PDR generally decreases as a function of Modulation & Coding Scheme (MCS), WSMP packet size, propagation distance, and vehicle speed; however, the relationship is non-linear and, in the case of distance, asymmetric, due to site specific propagation characteristics.

WSMP packet size possible using Commsignia's APIs. For each configuration, the car is driven each direction on the roadway twice at a target speed ranging from 10 to 40 miles per hour, while still adhering to traffic laws and conditions along the route. The WSMP packets are transmitted from the OBU and recorded synchronously with GPS coordinates of the vehicle during transmission. The WSMP packets received at the RSU are recorded and post-processed to determine the PDR for each configuration and vehicle location/speed by averaging over 2s windows of time.

The WSMP PDR characterization results are presented in Fig. 4. Fig. 4a shows the distribution of PDR across all locations and WSMP packet sizes for each of the tested MCS indexes. Unsurprisingly, PDR decreases as MCS is increased; yet, its surprising that MCS above 5 are often completely unusable at longer distances due to the exceptionally low PDR. Therefore, for the purposes of showing the effects of WSMP packet size and distance on PDR, only MCS 0 is used. Fig. 4b shows the distribution of PDR for MCS 0 across all locations for varying WSMP packet sizes. The PDR is quite high for the 100-300B WSMP packet sizes (which would likely be used for BSM and CAM use cases), but decreases for larger WSMP packet sizes (which would be more useful for transmitting larger data sizes such as the image frames studied in this work). Further, it can be seen in Fig. 4c that, while it is true that the link generally degrades as a function of distance, the relationship is non-monotonic and asymmetric as the wireless propagation depends on the site-specific geometry of the deployment studied in this work. For instance, the roadway to the western side of the RSU is occluded by foliage which leads to dramatically worse wireless link performance than similar distances to the east of the RSU.

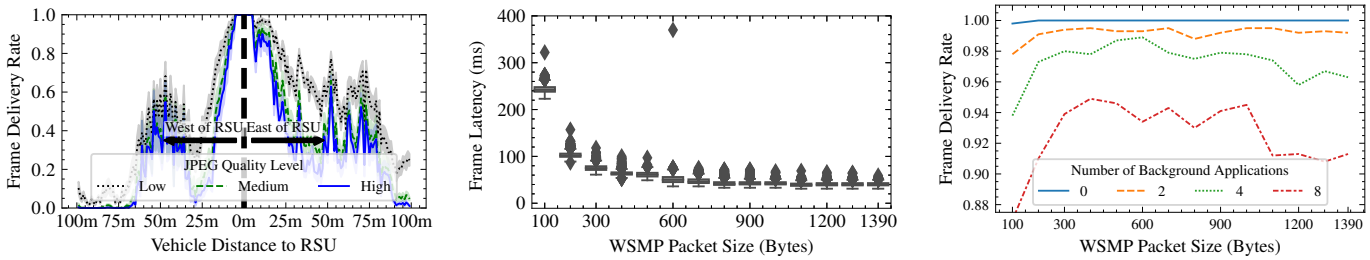
As mentioned previously, the experimental protocol used during the measurement campaign was to vary vehicle speeds. However, practical safety constraints prevented a uniform distribution from being obtained; for instance, the study area contains pedestrian crosswalks, a stop sign, and, in some trials, traffic can reduce the obtainable safe speeds as the roadway speed limit is 25mph. Fig. 4d shows a histogram of the vehicle speeds obtained during the measurement campaign. As can be seen, while it contains a wide range of speeds, it is heavily biased towards medium speed levels due to the roadway conditions. Nonetheless, this data can be used to show, in Fig. 4e for a few subsets of WSMP packet sizes with MCS fixed to 0, that higher vehicle speeds negatively impacts PDR, as would be expected due to increased Doppler shift. Due to the naturally occurring roadway factors that cause vehicle speeds to be heavily biased, the remainder of this work will consider performance across the entire range of speeds obtained without explicitly breaking out the performance into sub-domains.

B. Characterizing Frame Level QoS for RGB Sensor Sharing

Characterizing QoS at the WSMP packet level has been studied in prior works [1], [14]–[16]; however, this work is interested in characterizing an even more challenging task: image frame level QoS. From the WSMP packet level QoS, we can infer much of the frame level QoS (which is plotted in Fig. 5). As previously discussed, the successful reception of a frame is defined to be the successful reception of all of the individual packets required to successfully transmit that frame; therefore, with the assumption that the probability of reception of each packet is Independently and Identically Distributed for a given location, we can define FDR as

$$\text{FDR}(m, p, \mathbf{l}) = \text{PDR}(m, p, \mathbf{l})^n \tag{1}$$

where PDR, and by extension FDR, are functions of MCS (m), WSMP packet size (p), and vehicle location ($\mathbf{l}$) as



(a) FDR at MCS 0, 1390B WSMP Packets (b) Frame latency in presence of background traffic (c) FDR for varying background traffic levels

Fig. 5: Frame level QoS characterization of image transmission. In 5a, the impact of varying JPEG quality levels on FDR are shown. In 5b and 5c, the impact of emulated background traffic is shown where, surprisingly, the background traffic has little impact on frame latency but does result in lost frames.

discussed previously, and n is the number of packets needed to successfully contain the entire image frame.

As can been seen in Fig. 5a, the FDR follows the same shape as the PDR but the need for successful reception of n packets greatly decreases the FDR as PDR degrades. Obviously, needing to successfully communicate fewer packets will improve FDR; thus, this work evaluates the impact of compression, or JPEG quality level, on FDR. Logically, FDR can be improved by sacrificing image quality, in some cases sending a highly compressed image can sometimes more than double the FDR over sending an image with minimal compression. In other cases, utilization of heavy compression only provides a minimal benefit such as at the edge of the roadway study area where channel conditions are too poor to support any type of transmission.

C. Effect of Background Traffic on Frame Latency & Loss

While the previous sections have isolated and studied reliability as a key metric for QoS and how it can be impacted through adaptation of varying transmission parameters, they specifically ignored frame latency which is an undeniably crucial metric given the real-time nature of many vehicular applications. End-to-end packet delay in wireless communications is affected by a slew of different variables. The dominant source of delay in C-V2X is the queuing delay experienced while waiting on available radio resources - the length of this wait is primarily impacted by network congestion. This section describes an experiment designed to measure this frame latency in varying levels of network congestion and the results are presented in Fig. 5b and Fig. 5c.

To model network congestion, this work placed three C-V2X radios in a laboratory setting where wireless channel quality wouldn't be a limiting factor. One radio was used to transmit image frames (as described in Fig. 3) at a rate of 2 Frames per Second (FPS). While this rate is slower than the FPS used in the experiments conducted in the remainder of this work, it ensures that each frame is fully transmitted before the next one is sent and thus each frame transmission is independent. The transmitter sends 1000 frames for each WSMP packet size and these configurations are randomly shuffled to ensure uncontrolled time-varying factors do not impact the experimental results. One radio was used for the reception of the frames and is connected to the same computer as the transmitter to measure frame latency accurately. Each frame consists of 6.5KB of data representing a 300x300 image frame with a medium quality level along with meta-data encoding the transmission parameters and timestamp.

The third radio emulates background traffic applications (e.g. CAM or BSM messages) that will cause network congestion in commercial networks. Each background application is modeled using a Poisson process to determine both the inter-arrival times of the WSMP packets along with the payload size of each packet. The mean periodicity of the background application's WSMP packets, λ_T, is 100ms. The mean size of the background application's WSMP packets is taken to be 300B and is modeled in increments of 100B, (i.e. λ_S is 3). Therefore, each application repeatedly waits $t \sim \text{Pois}(\lambda_T)$ milliseconds and then sends a single WSMP packet of size $s \sim 100 \times \text{Pois}(\lambda_S)$ Bytes. A separate experiment is run for each number of background applications taken from $\{0, 2, 4, 8\}$ to represent a range of network congestion.

Somewhat unexpectedly, the network congestion does not cause significant differences in the latency of delivered frames as shown in Fig. 5b. While frame latency decreases as WSMP packet size increases (as would be expected due to needing to transmit a fewer number of WSMP packets for a fixed frame size) it does not have significant variance, or *jitter*. The impact of network congestion does however show up in FDR as shown in Fig. 5c. When background traffic is not present (i.e. 0 background applications), effectively 100% of the frames are successfully delivered; however, as the network congestion is increased by adding background applications, the FDR decreases. This is likely caused by either collisions or the C-V2X radio enforcing a Packet Delay Budget (PDB) for each WSMP packet; if the PDB cannot be satisfied due to insufficient spectral resources then the packet is dropped[3]. Interestingly, the results shown in Fig. 5c indicate that choosing the largest WSMP packet size may not always be optimal as it increases the chance that frame delivery will be unsuccessful when network congestion is encountered. Co-existence between the vastly different C-V2X use cases will be further discussed in Section VI-D.

[3]PDB is a configurable parameter in C-V2X and can be adapted in real-time by choosing amongst *priority levels*. Neither static nor dynamic PDB adaptation is performed in this work. The priority level of each sent WSMP packet is left as the default which chooses a 100ms value for PDB. Potential adaptations of PDB are left to a future study.

IV. Utilizing RL for Adaptive Image Transmissions Over C-V2X Sidelinks

The prior section demonstrated that C-V2X sidelinks are capable of carrying image frames but that the optimal transmission parameters vary based on the transmitting vehicle's location (and, by extension, the specific cell site geometry that influences wireless propagation characteristics) and the network congestion. The current section begins by formulating the adaptive image transmission task as a RL problem in order to provide a general framework that can learn to optimally transmit images from vehicles to the roadside edge in any given RSU deployment environment. This work will refer to this framework as Sensor Sharing OveR C-v2x sidElinks using Reinforcement lEaRning (`sorcerer`). This section then describes a system architecture developed in this work which unlocks training an RL Agent remotely deployed alongside commercial C-V2X radios using cloud based computing resources for training[4]. The experience collection and performance evaluation of this work are performed by the RL Agent with HIL meaning that the presented methodology can immediately be deployed on today's roadways. The section continues by presenting the methodology for emulating channel (modeled by the characterization undertaken in Section III) and network congestion conditions in a lab setup in order to provide a repeatable environment for experimentation and direct comparison of methodologies. Finally, the section concludes with a performance evaluation of an Agent trained to maximize application *goodput* and shows that it outperforms every static transmission configuration tested.

A. Defining the RL Problem

As shown in Fig. 6, there are three crucial signals to design when defining a RL problem: action, state, and reward. The problem studied in this work lends itself readily to make adaptation decisions on a per frame basis - where the frame rate is chosen arbitrarily in this work to be 10 FPS. The current section studies the impact of two continuous actions that can be taken on this frame (with two additional binary actions considered in Section V). The first action allows for indirectly adapting the frame size, B, by changing the JPEG quality level, q, used for compressing the current image frame before transmission. The second action allows for indirect control over latency and FDR by adapting WSMP packet sizes. As was established in Fig. 5, the optimal value of these changes based on the vehicle location and the amount of network congestion; thus, the state space should reflect estimates of these values. At each frame, the vehicle's current location, distance to the RSU, and the number of WSMP packets that have been observed from background traffic in the last second are utilized as the state space. The latter is likely an underestimate of network congestion as only successfully decoded WSMP packets are reported to the higher levels of the C-V2X stack where this metric is computed, but was a useful proxy for network congestion in this work given the limitations of the physical radio[5].

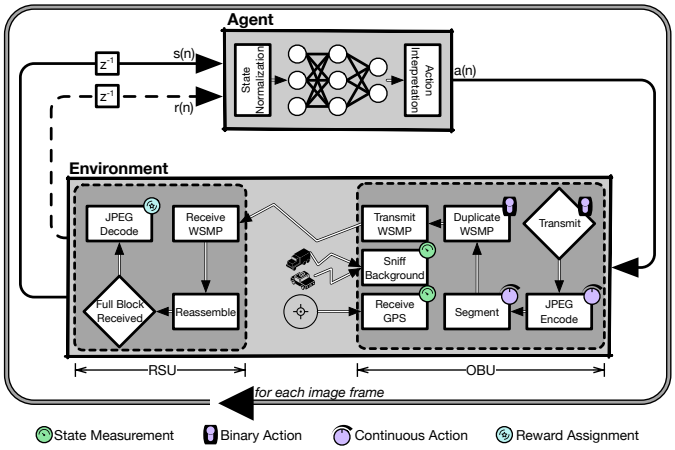


Fig. 6: For each image frame, the Agent utilizes the current state estimates to select the optimal action for the current frame being transmitted. The transmission of multiple WSMP packets containing the frame undergo wireless channel effects and are impacted by network congestion. If the entire frame is able to be received by the RSU, a reward is computed describing the utility of that image frame transmission - this reward is maximized by RL.

The third signal, reward, is particularly important as it defines *what* should be optimized. In the problem studied in this work, there are three axes of optimization: 1) image quality, 2) frame latency, and 3) FDR. The latter two are directly measurable in real time - each frame can be timestamped by the sender and assigned a unique frame identifier to determine whether frames have been lost in transit. The former, image quality, can be challenging for real time estimation. One metric that is widely used for determining the quality of image reconstruction is Peak SNR (PSNR). However, this metric requires knowledge of the original and received frame and is expensive to compute; these practical constraints lead this work to leave PSNR out of the reward function. Generally, using a higher compression ratio will degrade quality. This work therefore uses the frame data size as a proxy metric for image quality. Frame size can be interpreted as the encoding bit-rate which is also used as a proxy for Quality of Experience (QoE) in many ABR works [31]. These three metrics can then be logically combined into a single reward signal as

$$r(i) = \omega \mathbb{1}_D \frac{B}{T} \tag{2}$$

where B represents the frame size, T represents the frame latency, and $\mathbb{1}_D$ is an indicator that is 1 if the frame is successfully delivered to the receiver and 0 otherwise. This function can be intuitively interpreted as the application perceived instantaneous *goodput*, or the amount of usable data received within the time taken to deliver the frame, where ω is a constant used to scale the metric to measure Mbps. Investigating an alternate formulation of the reward function to dynamically prioritize FDR will be undertaken in Section V

[4]The logic for determining transmission parameters is done in real time, on vehicle, and therefore is not impacted by communication latency to/from cloud resources. RL training is a separate process, performed relatively infrequently, which is not sensitive to the increased latency of using cloud resources.

[5]Channel Busy Ratio (CBR) is a physical layer metric used in C-V2X that could more accurately estimate congestion, but the values being reported by the C-V2X radios in use were invalid and thus couldn't be used for the evaluation in this work. The usage of CBR in the state space, however, is not incompatible with the RL framework presented.

and exploring different definitions of QoE from prior works is explored in Section VI-A.

Given the signals defined above, this RL problem can be solved in multiple ways. This work chose to utilize deep RL due to its recent successes in many fields and its ability to arbitrarily model any policy. Specially, Soft Actor Critic (SAC) [40] was adopted as it is the state-of-the-art in continuous action spaces. The Agent is implemented by learning to parameterize a Beta distribution [41], which is bounded between $[0, 1]$ and then scaled and shifted to match the domain of each action. The discount factor, γ, was chosen to be 0.1 as the effect of each frame transmission is limited to a short time horizon (e.g. additional impacts of queuing will only impact the most recent subsequent transmissions). The policy is updated every 256 frames and a replay ratio of 4 is used to speed convergence. The Agent is configured to act randomly for 2000 frames in order to seed the replay buffer with a diverse set of experiences. The learning rate used is 3e-4 and 5e-3 is used as the Polyak averaging factor for soft updates of the target Q networks. The target entropy is taken as the negative size of the action space. Extensions to SAC will be discussed in Section V, but the current section simply uses SAC as described in prior works.

B. System Architecture Implementation

Many prior works have studied the applications of RL in wireless networks; however, these studies are generally conducted in simulated environments. This work applies RL in real-time on top of commercial C-V2X radios in order to demonstrate that RL is a mature methodology ready for practical applications in today's cellular networks. Doing so requires a robust engineering implementation that spans multiple nodes and necessitates networked operation of distributed systems. The system architecture developed for this work is shown in Fig. 7 and is described in further detail below.

The system consists of four nodes: 1) a frame source on the vehicle that periodically transmits image frames from the OBU, 2) a frame sink at the RSU that receives the image frames, 3) a RL task and experience database that is running within cloud computing resources, and 4) an OBU that is emulating background traffic. Discussion of the latter is deferred to Section IV-C. The source node hosts an application that sends image frames at a constant rate. A Connectivity Manager (`conman`) exposes a Python interface to this application for image transmission. The `conman` hosts the distributed network intelligence developed in this work which manipulates the image frame (e.g. JPEG compression), performs segmentation and reassembly of frames, and selects the WSMP parameters for transmission. In order to make decisions, `conman` needs a real time state estimate which is provided over ZeroMQ sockets connected to a Location and PC5 Service. The Location Service interfaces with an external GPS sensor to provide real-time latitude and longitude estimates of the source node's location which are published to `conman`. The PC5 Service uses the Commsignia Software Development Kit to interface with the OBU. It registers listeners to receive C-V2X traffic and publishes the payload of these packets to `conman` which can then determine the number of packets overheard from background applications. Further, the PC5 Service translates the payloads from `conman` to WSMP packets for transmission over C-V2X. After each transmission, `conman` uploads a partial experience to the cloud that consists of the state vector, the chosen action vector, and the transmission is timestamped and identified with a monotonically increasing frame number.

The sink node also hosts a PC5 Service for interfacing with the RSU. The background traffic is discarded and successfully reassembled frames are delivered to the application (as previously mentioned, if any WSMP packets belonging to a frame are dropped then the frame is discarded). The application then has all necessary information for computing the reward described in (2) and uploads it to the cloud based experience database. If the frame is not delivered, then no update needs to be provided as the reward can be considered as 0. While this methodology assumes time synchronization between the source and sink applications, there are numerous algorithms for maintaining this synchronization and therefore this is not a limiting assumption. Further, the ability to upload experience to the cloud assumes each node has network connectivity (e.g. a typical 5G connection over licensed spectrum for the OBU or wired back haul to the RSU). In practice, this is a reasonable assumption as the majority of new vehicles have 5G connectivity and the volume of data from experience reports is minuscule compared to the size of image frames. Additionally, latency requirements for experience reports can be significantly relaxed from the requirements for frame delivery.

There are two related tasks running within the cloud. The first is a database that uses a ZeroMQ socket, listening on a public interface, to collect experience reports and store them into a database. The database task performs the necessary linkages between reports such as: 1) matching state/action pairs uploaded by source node with the associated reward uploaded by the sink node as well as 2) augmenting each entry with the next visited state as it is necessary to compute the Bellman equations. Both linkages are easily achievable through usage of the frame number. This database is useful in its own regard as a monitoring tool that can provide near real time performance information of sensor sharing from vehicle to edge; yet, it's primary usage in this work is to be periodically queried by a RL task to return a random subset of experience for updating the policy network (using SAC as described in Section IV-A). This RL task listens to experience updates as they are delivered to the cloud in order to determine when to perform a training step. Once the policy network parameters are updated from this training step, they can be delivered to the instance of `conman` running on the source node. As the cloud node cannot be aware of all instances of `conman`, the source nodes open a connection back to the publicly accessible cloud node to subscribe to these updates (again using ZeroMQ).

C. Trace Driven Emulation of Field Conditions in the Lab

The system architecture presented in Fig. 7 can be deployed onto mobile vehicles, but this work extends it to also support

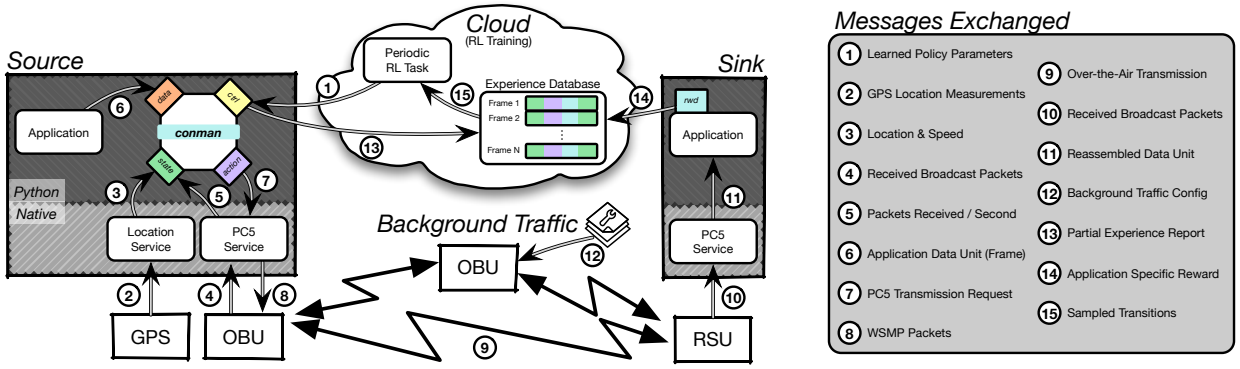


Fig. 7: System architecture of an Agent deployed onto a vehicle that can determine the transmission parameters for image frames using real time estimates of the current state and report the chosen actions, and associated state vector, back to the cloud as a partial experience that is identified by a frame number. If the frame is successfully received, the receiving application reports the application specific reward (e.g. frame latency) to the cloud to complete the experience entry for use in RL. A RL task running within the cloud periodically samples from these experiences to learn a more optimal transmission policy and then updates the policy parameters of the deployed agents.

trace driven emulation of the observed wireless channel characteristics on the C-V2X testbed (discussed in Section III) within a laboratory environment. This emulation capability provides lower cost and heightened reproducibility in order to facilitate the algorithmic experimentation and comparisons done in this work. Instead of a live GPS location, a GPS trace from a drive through the C-V2X testbed's study area is repeatedly read back from a file to emulate mobility. Whenever a WSMP packet is sent, the PC5 Service looks up the corresponding PDR for the given MCS, WSMP packet size, and vehicle location and encodes this PDR into an 8-bit trailer on the WSMP payload. The receiving PC5 Service decodes the PDR and conducts a Bernoulli trial to determine whether or not to artificially drop the WSMP packet to emulate poor wireless channel conditions. This allows the Agent to learn from, and be evaluated on, a link profile using data that is collected from the C-V2X testbed as described in Section III.

The background traffic is emulated from a separate OBU as described in Section III-C. However, each application is implemented as a two-state Markov process (i.e. ON-OFF application) in order to vary the amount of network congestion. The time that each application spends in each state is drawn from a Poisson distribution with a mean of 15s. The initial state of each application is randomly chosen. The total number of potential applications is 8 and the mean number of applications that will be active over the long term is 4 owing to the equal amount of time spent in the ON and OFF states.

Taken all together, the Agent interacts with a system as close as possible to real-world conditions as a laboratory setting will allow. During each transmission opportunity, the Agent is provided with a real image file, compresses and serializes it using JPEG, and transmits this byte array over a commercial C-V2X radio. The emulated background traffic and artificial packet loss introduced as a function of data playback, both described above, provide varying levels of network congestion and enable the Agent to experience random realizations of packet loss as they would occur due to the wireless channel conditions observed in the field.

D. Performance Evaluation

The Agent's performance during training is shown in Fig. 8. While the remainder of this work focuses on the Agent's performance after convergence, which is the more important operational period, Fig. 8 provides insight into what could occur if an Agent was trained on a live network from random initialization. The Agent begins by acting randomly to perform an initial exploration of the action space, it then adapts its behavior from the gathered reward signals to continually improve the operation of the link over time until eventually converging towards a more deterministic, and optimal, policy for image transmission. Fig. 8 shows the Agent's average performance in the environment over a rolling window of eight thousand image transmissions; as can be seen, the Agent's behavior quickly improves from the performance of taking random actions to outperforming all of the static configuration methodologies tested after only 150k image transmissions (and continues to refine its performance even after surpassing the performance of these alternatives).

As a comparison, three static transmission configurations were tested. In each, the WSMP packet size is fixed to 1390B as this provides the best frame latency (Fig. 5b). Three JPEG quality levels are considered, just as in Fig. 5a. Each static strategy is evaluated for 8k frames (matching the rolling window average of the Agent) for comparison - Fig. 8 shows that the proposed RL adaptive transmission strategy achieves a 6.8% better throughput than the best static methodology used for comparison (always transmitting the highest quality images with the largest WSMP packet size).

It is perhaps unsurprising that choosing to transmit high quality images provides the best comparison methodology as seen in Fig. 8. These high quality images represent the largest frame size and therefore drive instantaneous goodput to high values when the frame is successfully delivered. However, only about a third of these high quality frames are successfully delivered meaning that this strategy is wasteful, in terms of power and spectrum resources, and unreliable, in terms of the application which is inherently safety critical. By comparison, the RL Agent learned a strategy that not only outperformed the high quality image strategy in the defined reward function,

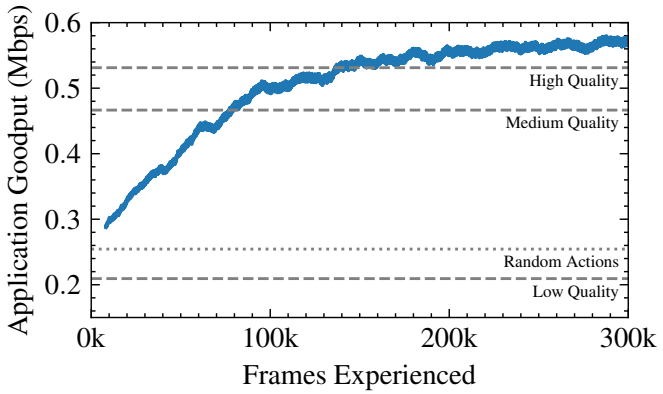


Fig. 8: Performance of `sorcerer` learning to maximize application goodput by adapting the JPEG quality levels and WSMP packet sizes - the Agent achieves 6.8% better goodput than any static configuration of the C-V2X link.

but greatly out-performed this static strategy in terms of FDR. Overall, every strategy has a low FDR with only the transmission of low quality images leading to more frames being delivered than lost; Section V will explore techniques to remedy this.

V. Self-Optimization of Risk vs Reward Tradeoffs

Attempting to provide the best possible goodput can lead to a greedy policy that frequently swings for the fences - sending high quality image frames even if they aren't reliably received as, on average, they provide the best reward. However, this approach is incredibly inefficient in terms of power and spectrum usage. Additionally, from the application's perspective it is better for the network to explicitly declare a service outage is occurring, which could be proactively handled within application logic, instead of quietly dropping the frames in transit. This section seeks to answer the questions: 1) how can reliability be improved? 2) how can a service outage be declared? and 3) how can a target reliability be supplied to the Agent? To answer these questions, this section first describes modifications to SAC to support an additional categorical action space that supplements the continuous action space presented by `sorcerer` in Section IV and allows for declaring a service outage along with increasing FDR through a form of blind Hybrid Automatic Repeat Request (HARQ). Then, methodology that allows for a specified risk constraint to be optimized through Lagrangian relaxation is presented, which this work refers to this as `sorcerer`-Risk Constrained (`sorcerer-rc`). Finally, the section concludes with a discussion of the final performance of all policies and a discussion on the specific learned behaviors.

A. Improving FDR with an Extended Action Space

Improving FDR can certainly be accomplished by a combination of reducing the number of WSMP packets to send (by reducing the JPEG quality level) and/or lowering the WSMP packet size. However, most wireless communications typically utilize a form of Forward Error Correction that allows for correcting errors that occur due to the wireless channel; for example, most cellular networks utilize HARQ that transmits additional data for error correction when it cannot be decoded. As C-V2X Mode 4 utilizes a broadcast channel, and does not have feedback, it utilizes blind HARQ which simply duplicates the WSMP packet, with different coding applied, in order to improve the probability of successful reception. This work (imperfectly) models this by allowing the Agent to choose whether WSMP packets should be transmitted twice and therefore increase reliability at the cost of queuing latency.

Despite best efforts, poor channel conditions can simply mean that it is currently impossible to reliably transmit an image frame. In these cases, the Agent should be allowed to decide not to send at all. By doing so, it can not only conserve power and spectrum resources but, also a service outage can be explicitly declared to the application. The seemingly two binary decisions of choosing whether to send and whether to duplicate, can be more succinctly viewed as a Categorical action space where the Agent can either: 1) choose not to send a frame, 2) send the frame, or 3) send the frame with duplication of WSMP packets. This extended action space enables the Agent to grow the effective service area through duplication and explicitly declare this effective service area by choosing not to transmit. Furthermore, as the Agent can now choose to shrink the service area, any reliability constraint becomes feasible where there is an upper bound on reliability when the Agent is forced to transmit at every opportunity.

As previously mentioned, SAC was chosen for the RL algorithm as it excels when using continuous action spaces. Recent works have explored how to adapt SAC to discrete action spaces [42], [43]; this work leverages these efforts to combine an Agent that concurrently learns a policy for both a continuous (changing WSMP packet size and JPEG quality levels) and a discrete action space (choosing whether to transmit at all or with WSMP packet duplication). The primary difference between the discrete and continuous versions of SAC are in how the Q-function is computed. For a continuous action space, the potential number of actions is infinite and therefore the Q-function, $Q_c(\mathbf{s}, \mathbf{a_c})$, is computed using both state, $\mathbf{s}$, and the sampled continuous action, $\mathbf{a_c}$, as inputs. This action is sampled from the policy, π_c, using the reparameterization trick and is therefore end-to-end differentiable and can be optimized using any gradient based method. However, with discrete action spaces, the value of each state-action pair can be explicitly learned where the Q-function, $Q_d(\mathbf{s})$, only takes the state as input and learns a separate output for each potential discrete action, $\mathbf{a_d}$, where $\mathbf{a_d}$ is a vector that defines the probability of taking each discrete action (or is a one-hot vector describing the sampled action). This work therefore combines these two approaches and models the Q-function for both a continuous and discrete action space as

$$Q(\mathbf{s}, \mathbf{a_c}, \mathbf{a_d}) = Q_c(\mathbf{s}, \mathbf{a_c}) \times \mathbf{a_d}^\top \tag{3}$$

Given the usage of two policy functions for categorical and discrete action spaces and the necessary extension of the Q-function described by (3), the loss functions need to be

updated. The Q-loss, J_Q, is extended as

$$J_Q(\mathbf{s}, \mathbf{a_c}, \mathbf{a_d}, r, \mathbf{s}', \mathbf{a}'_\mathbf{c}, \mathbf{a}'_\mathbf{d}) = \{Q(\mathbf{s}, \mathbf{a_c}, \mathbf{a_d}) - [r + \gamma \min_{i \in \{1,2\}} Q_{T_i}(\mathbf{s}', \mathbf{a}'_\mathbf{c}, \mathbf{a}'_\mathbf{d})]\}^2 \quad (4)$$

where Q_{T_i} represents the target Q networks. The alpha-loss, J_α, which dynamically targets a specific entropy from the policy network can then be computed using the joint entropy of both the continuous and discrete policies.

$$J_\alpha(\mathbf{s}) = \alpha\{H[\pi_d(\mathbf{s})] + H[\pi_c(\mathbf{s})] - \alpha_T\} \quad (5)$$

In (5), $H[\cdot]$ defines the entropy of the policy and α_T represents the target entropy. The usage of α in policy optimization remains the same as standard SAC and will be described further in the next section.

B. Automatic Objective Tuning to Achieve QoS Constraints

Maximizing network throughput is often best effort; it is expected to fluctuate due to channel and/or network conditions and applications will typically gracefully degrade as throughput declines. However, the reliability of a network is often provided as a constraint. Therefore, this work seeks to allow this constraint to be provided to the Agent optimizing the transmission parameters. A risk signal, ρ, can be defined as

$$\rho = \begin{cases} 1, & \text{if sent and not delivered} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

which is easily computed from the data available in the experience database as described in Section IV-B. The Agent can then be optimized to solve the following problem:

$$\begin{aligned} \max_{\pi_d, \pi_c} \quad & \sum_{i=o}^{\infty} \gamma^i r(i) \\ \text{s.t.} \quad & \mathbb{E}[\rho] \leq \beta_T \end{aligned} \quad (7)$$

which maximizes the reward signal defined by (2) across an infinite discounted time horizon, where γ represents the discount factor that remains 0.1 as initially defined in Section IV-A, while ensuring that the average risk of the policy remains below an operator-supplied threshold, β_T. Optimizing the policy to maximize the discounted sum of rewards has been well studied in various RL literature [44], and is handled in this work by SAC, as previously described. However, constrained policy optimization is more challenging and requires relaxing the optimization problem to

$$\max_{\pi_d, \pi_c} \quad \sum_{i=o}^{\infty} \gamma^i r(i) - \beta\, \mathbb{E}[\rho] \quad (8)$$

where the discounted reward is estimated by the $Q(\cdot)$ function and a new risk function, $\mathrm{P}(\cdot)$, can be defined to estimate the risk of a given state/action pair. Modeling $\mathrm{P}(\cdot)$ using a Neural Network ensures that it is differentiable and can be easily learned by minimizing the mean squared error of its predictions with the empirically observed risk signal.

$$J_\rho = \{\mathrm{P}(\mathbf{s}, \mathbf{a_c}, \mathbf{a_d}), \rho\}^2 \quad (9)$$

The risk function is modeled identically to the Q function described in (3) with the addition of a Sigmoid function on the output to bound the risk between 0 and 1. Additionally, the loss function (9), differs from the loss functions used for training the Q function as it does not require the Bellman equation (since it does not depend on any future values). The optimal coefficient, β, for modeling the constraint can then be learned by minimizing the following cost function:

$$J_\beta = -\beta[\mathrm{P}(\mathbf{s}, \pi_c(\mathbf{s}), \pi_d(\mathbf{s})) - \beta_T] \quad (10)$$

The policy loss is then a dynamically weighted summation of the policy entropy, state/action estimated risk, and Q-value.

$$J_\pi(s) = \alpha\{H[\pi_d(\mathbf{s})] + H[\pi_c(\mathbf{s})]\} - \beta\, \mathrm{P}[\mathbf{s}, \pi_d(\mathbf{s}), \pi_c(\mathbf{s})] + \min_{i \in \{1,2\}} Q[\mathbf{s}, \pi_c(\mathbf{s}), \pi_d(\mathbf{s})] \quad (11)$$

The algorithm for risk constrained SAC is identical to the original SAC algorithm except for the redefinition of loss functions presented above and the addition of the new loss functions, J_β and J_ρ.

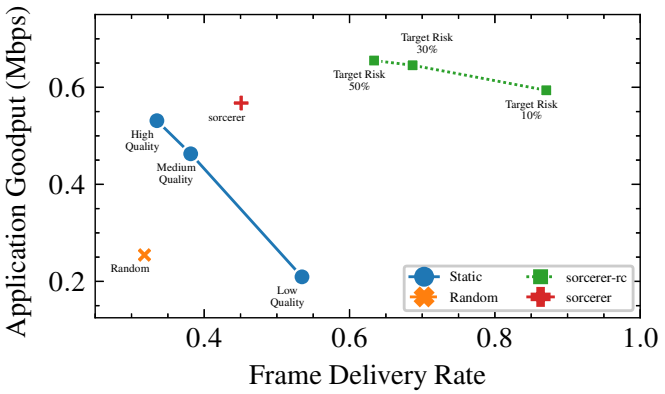


Fig. 9: Evaluation performance of all Agents on 8k frames.

C. Performance Evaluation

A separate Agent is trained for 300k frames (as was done in Section IV) for a range of target risk thresholds: $\{0.5, 0.3, 0.1\}$. After training, the Agent is evaluated for 8k frames (to match the evaluation of the static strategies) with results shown in Fig. 9. Two metrics are computed from those 8k frames: 1) application goodput and 2) FDR. The application goodput is calculated as the average instantaneous goodput over all N frames.

$$\text{goodput} = \tfrac{1}{N} \sum_n \omega \mathbb{1}_D(n) \tfrac{B(n)}{T(n)} \quad (12)$$

The FDR is calculated as the ratio between the frames successfully delivered and those sent.

$$\text{FDR} = \frac{\sum_n \mathbb{1}_D(n)}{\sum_n \mathbb{1}_S(n)} \quad (13)$$

As can be seen in Fig. 9, the extended methodology presented in this section achieves higher goodput than all comparisons, even in the case of the most stringent risk threshold of 10%. This is primarily due to the ability to selectively enhance reliability by duplicating WSMP packets (specific learned policy behavior will be discussed in Section V-D).

The Agent with the most relaxed risk target (50%) achieves a 23.3% improvement in goodput over simply always sending high quality frames while still delivering 9.9% more frames than when always sending low quality frames - *adaptive image transmission provides greatly improved performance regardless of the metric of interest*. Furthermore, the adaptation methodology presented in this section achieves a better tradeoff between FDR and goodput (which can be observed as the slope of each line). The Agent with a target risk of 10% still achieves 4.6% and 11.8% goodput improvements over the simpler methodology from Section IV and the static configuration of sending high quality images respectively. This work simultaneously enables high goodput and FDR with the desired FDR able to be explicitly targeted with a constraint.

The expectation of risk should be directly related to $1-$FDR. Fig. 9 shows that this relationship is closely preserved by the optimization. The Agent with a 10% risk threshold achieves 87.1% FDR and the Agent with a 30% risk threshold achieves a 68.7% FDR. While this is quite close, it still violates the constraint by $\approx 2\%$. We postulate there are two possible causes: 1) bias in risk estimation and/or 2) environment drift between training and evaluation. For the former, many recent RL algorithms utilize multiple Q networks[6] for mitigating estimation bias. This work only used a single network for risk estimation and its possible that using two networks could help to mitigate the estimation bias. This study is left to future work. The other possible reason is environment drift. While this work uses commercial C-V2X radios to show the maturity of the developed approach to adapt to real world challenges, it also means that these real world challenges impact the experimental results. Therefore, the environment experienced during evaluation could have seen additional interference that lowered FDR. Further, the C-V2X radios used in this work periodically cease transmission[7], meaning that a lack of background traffic during portions of training could have led to underestimation of the risk of frame loss. In either case, lifelong learning of RL would eventually adapt the policy to meet the constraint under this environment drift.

D. Exploring the Learned Policy

While this work was shown to have an objective benefit, it can also be useful to perform a subjective analysis of the learned policies. In order to extract policy behavior, a set of sample state vectors are constructed corresponding to every 1m of roadway within the study area. The final piece of state information, the amount of observed network congestion, is simply set to the mean value. This sample state vector can then be passed through the final learned policies to extract an imperfect visualization of their behavior, which is shown in Fig. 10. The discrete decisions that are taken at each location are represented by the shaded/textured background; only the greedy decision, the action with the highest probability, is shown for simplicity. The plots show the mean values of the continuous policy at every location and, because they have different scales, there are two Y-axes corresponding to each potential action. Two policies are visualized. Fig. 10a shows the policy when following the Agent learned with a target risk threshold of 30%, while Fig. 10b shows the behavior of the Agent that was trained with a target risk threshold of 50%.

Comparing the two policies in Fig. 10 shows that when wireless channel conditions are good, a similar policy is learned regardless of the risk tolerance. For example, when the vehicle is nearby the RSU or ≈ 50m to the west of it, both Agents determined that the optimal thing to do is simply to send the highest possible quality image, using the largest packet size and not using duplication to minimize latency. Similarly, both Agents learned that channel quality generally degrades ≈ 50m to the east of the RSU and both handled this by sending the lowest quality images and using duplication and smaller packet sizes to increase reliability. Beyond those regions, the two policies maintain a similar shape, but the scales are different corresponding to the risk tolerance. For instance, ≈ 75m to the east of the RSU, where channel quality begins to improve, each Agent responds to this by increasing the quality of images transmitted, but the policy in Fig. 10a is more conservative in this increase as it is has a tighter risk budget. Furthermore, the Agent trained with a 30% target risk decided that frame transmission at the edges of the study area was simply too unreliable to warrant occurring, showcasing the ability to dynamically learn effective coverage areas.

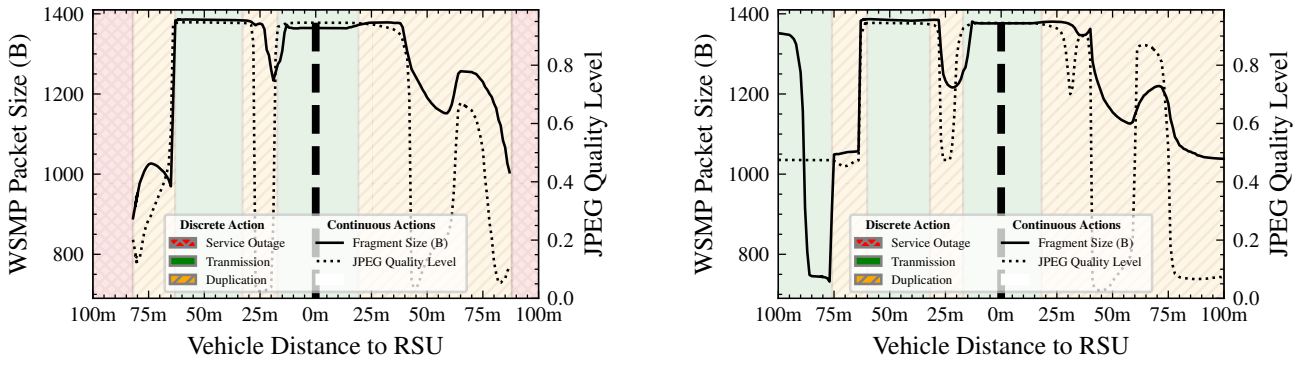
The only portion of the observed policies that is not intuitive based upon an understanding of the underlying deployment is the western edge of Fig. 10b. The edges of the study area have poor channel quality that leads to unreliable transmissions; ergo, it seems rational that many of the mechanisms to increase reliability would kick in such as using duplication and lowering the quality of images and size of WSMP packets. Yet, Fig. 10b shows that is not what was learned. It is suspected that the Agent simply failed to converge to a useful policy in this region of the state space. Instead, the Agent simply took random actions as it was still exploring - the 50% quality level representing the mean of a uniform distribution policy on the JPEG quality level.

VI. Ablation Studies

This section pressure tests the design decisions of this work by performing a set of ablation studies, whose results are outlined in Table I. First, the specific reward function used is swapped with other possible formulations found in literature showing that, as RL is used, the specific definition is unimportant to the success of the framework for optimizing any metric. Then, the impact of reductions to the state and action spaces are explored that could be thought to aid in generalization or casting the problem into a purely application adaptation approach. Finally, the impact of the more communication resource intensive application studied in this work on background traffic is explored. Unless stated otherwise, for all experiments, the methodology developed in Section V, `sorcerer-rc`, with a target risk threshold of 50% is used as a base with small modifications described in each subsection.

[6]Two networks are trained to estimate the Q-function and the minimum Q-estimation between the two is used.

[7]The cause of the C-V2X radios periodically silently stopping transmission is unclear but is believed to be due to a momentary loss of the GPS lock that is required for operation.



(a) Target Risk 30%

(b) Target Risk 50%

Fig. 10: Mean policy extracted from `sorcerer-rc` Agents using a sweep of the roadway study area, at 1m increments, with the observed background traffic set to the mean observed value. Shaded/textured regions of the background represent the greedy discrete action decisions and each line represents the mean of the continuous policy for that action. 10a depicts the policy learned when a target risk threshold of 30% is set while 10b showcases a target risk threshold of 50%.

TABLE I: Results of the ablation studies targeting different QoE metrics and removal of portions of the action or state space. Except for the static configurations, a separate model is trained to optimize each definition of QoE or permutation of the action/state spaces.

Action Space			*State Space*	*Reward Function (QoE)*		
JPEG Quality	**WSMP Packet Size**	**Outage/ Duplicate**	**Access to Precise Location**	**Goodput** (12)	**QoE$_{lin}$** (14)	**QoE$_{log}$** (15)
Adaptive	Adaptive	✔	✔	**0.66**	**0.44**	**0.53**
Adaptive	Adaptive	✔	✗	0.58	-	-
Adaptive	Adaptive	✗	✔	0.57	-	-
Adaptive	1300	✗	✔	0.54	-	-
High (95)	1390	✗	N/A	0.53	0.33	0.32
Medium (50)	1390	✗	N/A	0.47	0.21	0.28
Low (0)	1390	✗	N/A	0.21	0.06	0.16

The light gray text indicates the action or state space is unchanged from `sorcerer-rc`
An ✗ indicates discrete actions or state values are withheld
Average rewards are bolded to indicate the best performer within that column

A. Varying Definition of QoE

This work chose to optimize the instantaneous goodput; however, other works have used differing definitions of QoE from the user's perspective when viewing a video. For example in [31], varying notions of image quality is summed with penalties for re-buffering and changes to the encoding rate. While the latter two penalties aren't relevant to this work the definitions of image quality can be explored. The first definition of QoE, termed QoE$_{\text{lin}}$ in [31], is simply defined as the linear function of the encoding bit rate

$$\text{QoE}_{\text{lin}} = \mathbb{1}_D \frac{B}{B_{max}} \tag{14}$$

This work utilizes the frame size as the encoding bit rate and normalizes it to be between 0 and 1. A second definition of QoE used in prior works is termed QoE$_{\text{log}}$ and captures the notion of diminishing returns as image quality is increased. This was defined in [31] as $\log(B/B_{min})$ which is bounded by 0 for the lowest bit rate and logarithmic as bit rate increases. However, this definition does not capture the fact that a frame could potentially be lost due to the lack of a re-transmission mechanism, therefore this work modifies QoE$_{\text{log}}$ as

$$\text{QoE}_{\text{log}} = \log(1 + \mathbb{1}_D \frac{B}{B_{min}}) \tag{15}$$

which is bounded by 0 when the frame is not received and still logarithmic as bit rate increases. The mean performance during evaluation for each definition of reward is presented in Table I for this work (i.e. `sorcerer-rc`) on the top row and the comparison static link configurations in the bottom three rows. As can be seen, regardless of the specific metric chosen for use as a reward function, this work is able to learn a policy that significantly outperforms a static configuration of the C-V2X link on the metric in question.

B. Ability to Generalize Using Only Distance

This work utilizes the latitude and longitude of the vehicle in its state space. This enables learning rich policies that could learn the specific wireless propagation characteristics of the RSU study area; however, the consequence is that, while the methodology will transfer to any RSU deployment, the specific learned policies will not generalize. Therefore, while this work does not have access to multiple RSU deployments, it can remove the specific location information (e.g. latitude and longitude) from the state space and force the Agent to learn a policy that is symmetric around the RSU by only using the distance to the RSU. While this experiment will not precisely show the ability to generalize, it will present a lower bound on the performance degradation caused by the simplified state space. Table I shows that using distance alone, indicated by an ✘ in the "Access to Precise Location" column, can outperform both static configurations of the C-V2X link and the other ablations but that this generalization incurs a significant cost in the achievable performance. This reinforces the choice to utilize GPS coordinates as well for a richer state space that allows the Agent more freedom to adapt the transmission policy.

C. Constraining to Bit Rate Adaptation

This work developed a rich action space that logically spans multiple layers of the protocol stack. While adjusting the JPEG quality levels can be analogous to an ABR algorithm that would typically run within the application layer, choosing the WSMP packet size, selective usage of a form of blind HARQ, and choosing to declare a service outage are functionalities that more closely align with the lower layers of the protocol

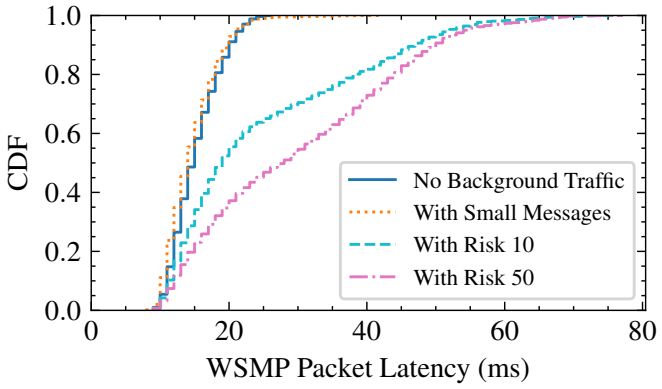


Fig. 11: Cumulative Distribution Function (CDF) of observed WSMP packet latency under varying types of background traffic. Small messages are emulated as described in Section IV-C and image transmission is modeled using `sorcerer-rc` with varying risk thresholds. The CDF describes the percentage of received WSMP packets whose latency is below the value indicated on the X-Axis. While the presence of small messages has effectively zero impact on latency, the presence of image transmissions greatly impacts the observed worst case latency.

stack. To evaluate localizing the Agent's impact to only the logical application layer, this work constrains the action space to only choosing the JPEG quality level. Table I shows that this constrained action space only narrowly outperforms, in terms of goodput, a static configuration of the C-V2X link (while also achieving an advantage in FDR by successfully delivering 10% more frames). Therefore, the decision to include WSMP packet size in the action space in `sorcerer` is certainly beneficial and the additional extensions of `sorcerer-rc` can greatly increase FDR beyond what ABR alone can provide.

D. Characterizing Impact on Traditional V2X Applications

Finally, this work proposes using the C-V2X Sidelinks for a more data intensive application. This will undoubtedly congest the network and lead to degrading performance of the safety critical applications that currently use it. To evaluate the extent of this effect, this section presents a study that is the logical inverse of the one presented in the rest of the work. The sending of image data is considered as the background traffic and the impact on latency and delivery rates of smaller WSMP packets are measured. 1000 packets are sent for each WSMP packet size in $\{200, 300, 400, 500\}$ at a rate of 10 WSMP packets per second with their configurations randomized. The latency of each WSMP packet is recorded and aggregated results are displayed in Fig. 11. Although the presence of background traffic emulated as small messages, such as BSMs or CAMs, has nearly zero functional impact on the observed latency of WSMP packets, the presence of image transmission nearly doubles the observed latency. The usage of a more risk averse policy, which is generally more conservative in its transmissions, can mitigate the average latency impact to traffic from today's use cases; e.g. $\approx 60\%$ of transmissions still achieve sub 20ms latency. However, the worst case latency still remains the same when high quality images are being transmitted (for instance, when the vehicle is nearby the RSU). The worst case latency is a crucial metric for safety critical messages; therefore, enabling learning of a transmission policy that mitigates these potential negative impacts to other users of the C-V2X spectrum is an area for future work.

VII. Conclusion

This work evaluated the capabilities of today's commercial C-V2X radios for supporting the challenging task of transmitting sensor data from vehicles to RSUs with the specific use case of image data studied. To support this evaluation, a C-V2X testbed was deployed on the UCSD campus. It was found that despite the fact that today's C-V2X was designed for the primary purpose of reliably transporting small messages, C-V2X, operating in Mode 4, can support sharing of image data from vehicles to the roadside edge with ≈ 50ms frame latency. This real-time sensor data sharing can enable advanced use cases such as collaborative perception or computation offloading to become reality. However, this analysis also explicitly exposed a limitation of today's C-V2X: the technology is meant for broadcast transmission and lacks the feedback necessary for adapting transmission parameters to wireless channel and network conditions.

This work developed a RL framework for blind adaptation of transmission parameters from the available out-of-band information. Specifically, the JPEG quality level and WSMP packet sizes were adapted on a per-frame basis, using real-time estimates of vehicle location and network congestion. The RL framework was trained and evaluated with the C-V2X HIL which shows this technology is mature enough for real-world deployment. While this application of RL was shown to provide a 6.8% improvement in the effective throughput, the FDR was still lower than 50%. Therefore, this work explored the selective usage of a form of blind HARQ for improving reliability. Moreover, this work presented extensions to SOTA RL that allowed for a constraint on FDR to be provided and, if this constraint could not be met, to simply cease transmission instead of wasting power and spectrum resources. This advanced methodology was shown to achieve a 23.3% improvement in effective throughput over simply sending high-quality images while still delivering 9.9% more frames than when the policy is to only send the lowest possible quality images. Further, this work showed that this methodology could be used to learn a transmission policy that successfully delivered 87.1% of the frames it sent - *RL is capable of enabling highly reliable transmission of image frames from vehicle to RSU while maximizing the ratio between image quality and latency*. This can all be done by simply layering network intelligence onto currently commercialized C-V2X radios. No modifications to C-V2X standards or radio firmware are needed to achieve these results.

This work took great effort to evaluate and augment C-V2X in a realistic scenario: using commercial C-V2X radios to conduct a measurement campaign, training and evaluating RL Agents with HIL, emulating varying network loads from an external radio, etc. Yet, some potential issues may only be exposed at scales beyond the current capabilities of the UCSD C-V2X testbed. For instance, deploying multiple RSUs in varying deployment scenarios (e.g., highways or denser urban environments) would enable evaluation in a broader

range of speeds and wireless channel conditions. Overlapping RSU coverage ranges would allow for exploring soft handoffs. While C-V2X Mode 4 operates in a broadcast mode that has no hard association with any specific RSU, the RL methodology developed in this work optimizes for specific links and, therefore, may have to be extended with some form of implicit link selection logic in scenarios where multiple RSUs are in the communication range of the vehicle. This scale would also enable further exploration of the most crucial limitation identified in this work: the transmission of images causes significant network congestion that can nearly double the latency for other users of the C-V2X spectrum.

REFERENCES

[1] Y.-J. Ku, B. Flowers, S. Thornton, S. Baidya, and S. Dey, "Adaptive C-V2X sidelink communications for vehicular applications beyond safety messages," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, pp. 1–6, June 2022.

[2] Qualcomm, Inc, "V2X Technology Benchmark Testing." https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/fcc_usdot_cv2x_-_v2.14_w_video1.pdf, 2018. Accessed: 2023-07-30.

[3] A. Festag, "Cooperative intelligent transport systems standards in europe," *IEEE Comms. Magazine*, vol. 52, pp. 166–172, December 2014.

[4] F. Perry, "Overview of dsrc messages and performance requirements," in *UFTI DSRC and Other Communication Options for Transportation Connectivity Workshop*, 2017.

[5] 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, pp. 30–39, Dec 2017.

[6] 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, pp. 1972–2026, thirdquarter 2021.

[7] V. Va, T. Shimizu, G. Bansal, and R. W. H. Jr., "Millimeter wave vehicular communications: A survey," *Foundations and Trends in Networking*, vol. 10, no. 1, pp. 1–113, 2016.

[8] W. Zheng, A. Ali, N. González-Prelcic, R. W. Heath, A. Klautau, and E. M. Pari, "5G V2X communication at millimeter wave: rate maps and use cases," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–5, May 2020.

[9] N. J. Myers, Y. Wang, N. González-Prelcic, and R. W. H. Jr, "Deep learning-based beam alignment in mmwave vehicular networks," 2019.

[10] V. Va, T. Shimizu, G. Bansal, and R. W. Heath, "Online learning for position-aided millimeter wave beam training," *IEEE Access*, vol. 7, pp. 30507–30526, 2019.

[11] M. Alrabeiah, A. Hredzak, Z. Liu, and A. Alkhateeb, "ViWi: A deep learning dataset framework for vision-aided wireless communications," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–5, May 2020.

[12] R. Fukatsu and K. Sakaguchi, "Millimeter-wave V2V communications with cooperative perception for automated driving," in *2019 IEEE 89th Vehicular Technology Conf. (VTC2019-Spring)*, pp. 1–5, April 2019.

[13] FCC, "Use of the 5.850-5.925 GHz Band," Report and Order 35 FCC Rcd 13440 (16), Federal Communications Commission (FCC), 11 2020.

[14] M. Chen, R. Chai, H. Hu, W. Jiang, and L. He, "Performance evaluation of C-V2X mode 4 communications," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, March 2021.

[15] 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, pp. 1155–1166, Feb 2019.

[16] 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)*, pp. 43–48, Dec 2018.

[17] S. Thornton and S. Dey, "Machine learning techniques for vehicle matching with non-overlapping visual features," in *2020 IEEE 3rd Connected and Automated Vehicles Symp. (CAVS)*, pp. 1–6, Nov 2020.

[18] R. Guo, S. Keshavamurthy, and K. Oguchi, "Simultaneous object detection and association in connected vehicle platform," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 840–845, 2018.

[19] H. Liu, P. Ren, S. Jain, M. Murad, M. Gruteser, and F. Bai, "Fusioneye: Perception sharing for connected vehicles and its bandwidth-accuracy trade-offs," in *2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1–9, 2019.

[20] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, SEC '19, (New York, NY, USA), p. 88–100, Association for Computing Machinery, 2019.

[21] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 514–524, July 2019.

[22] E. E. Marvasti, A. Raftari, A. E. Marvasti, Y. P. Fallah, R. Guo, and H. Lu, "Cooperative lidar object detection via feature sharing in deep networks," *arXiv preprint arXiv:2002.08440*, 2020.

[23] L. Yang, J. Cao, Z. Wang, and W. Wu, "Network aware multi-user computation partitioning in mobile edge clouds," in *2017 46th International Conference on Parallel Processing (ICPP)*, pp. 302–311, IEEE, 2017.

[24] Y.-J. Ku, S. Baidya, and S. Dey, "Adaptive computation partitioning and offloading in real-time sustainable vehicular edge computing," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 12, pp. 13221–13237, 2021.

[25] S. Aoki, T. Higuchi, and O. Altintas, "Cooperative perception with deep reinforcement learning for connected vehicles," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, pp. 328–334, Oct 2020.

[26] B. Gu and Z. Zhou, "Task offloading in vehicular mobile edge computing: A matching-theoretic framework," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 100–106, 2019.

[27] G. Qiao, S. Leng, K. Zhang, and Y. He, "Collaborative task offloading in vehicular edge multi-access networks," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 48–54, 2018.

[28] J. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in HTTP-based adaptive video streaming with festive," *IEEE/ACM Transactions on Networking*, vol. 22, pp. 326–340, Feb 2014.

[29] Y. Sun, X. Yin, J. Jiang, V. Sekar, F. Lin, N. Wang, T. Liu, and B. Sinopoli, "CS2P: Improving video bitrate selection and adaptation with data-driven throughput prediction," in *Proceedings of the 2016 ACM SIGCOMM Conference*, SIGCOMM '16, (New York, NY, USA), p. 272–285, Association for Computing Machinery, 2016.

[30] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, (New York, NY, USA), p. 187–198, Association for Computing Machinery, 2014.

[31] 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*, SIGCOMM '17, (New York, NY, USA), p. 197–210, Association for Computing Machinery, 2017.

[32] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, SIGCOMM '15, (New York, NY, USA), p. 325–338, Association for Computing Machinery, 2015.

[33] X. Xie, X. Zhang, S. Kumar, and L. E. Li, "PiStream: Physical layer informed adaptive video streaming over LTE," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, (New York, NY, USA), p. 413–425, Association for Computing Machinery, 2015.

[34] C. Inc., "High performance V2X enabled roadside unit with edge computing." commsignia.com/products/rsu/, 2022. Accessed: 2022-02-18.

[35] Qualcomm Inc., "C-V2X 9150." qualcomm.com/products/qualcomm-c-v2x-9150, 2022. Accessed: 2022-02-18.

[36] NVIDIA, "High performance AI at the edge — NVIDIA jetson TX2." nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-tx2/, 2022. Accessed: 2022-10-25.

[37] Commsignia Inc., "Powerful V2X Onboard Unit." commsignia.com/products/obu/, 2022. Accessed: 2022-02-18.

[38] 3GPP, "Study on LTE-based V2X services," Technical Report (TR) 36.885, 3rd Generation Partnership Project (3GPP), 7 2016. Version 14.0.0.

[39] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," Technical Specification (TS) 36.213, 3rd Generation Partnership Project (3GPP), 9 2022. Version 16.9.0.

[40] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *CoRR*, vol. abs/1801.01290, 2018.

[41] P.-W. Chou, D. Maturana, and S. Scherer, "Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution," in *International conference on machine learning*, pp. 834–843, PMLR, 2017.

[42] P. Christodoulou, "Soft actor-critic for discrete action settings," *arXiv preprint arXiv:1910.07207*, 2019.

[43] H. Zhou, Z. Lin, J. Li, D. Ye, Q. Fu, and W. Yang, "Revisiting discrete soft actor-critic," *arXiv preprint arXiv:2209.10081*, 2022.

[44] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.