Low Overhead Codebook Design for mmWave Roadside Units Placed at Smart Intersections

Bryse Flowers, Xinyu Zhang, Sujit Dey
Center for Wireless Communications
University of California, San Diego
{bflowers, xyzhang, dey}@ucsd.edu

***Abstract*—In order to meet the high data rate requirements of emerging roadway use cases, mmWave vehicular communications will be needed. This work studies the ability of vehicles to communicate with a Roadside Unit (RSU) placed at an intersection. Practical mmWave radios utilize a codebook, a discrete set of analog beams, that is periodically searched during runtime to find the optimal beam to use for each receiver. This search creates overhead as the wireless channel is not used for communication while this beam search is happening. This work focuses on reducing the overhead of beam training by optimizing the site-specific codebook design of a RSU. Owing to the sparsity of the mmWave channel and the user distribution for vehicles, it is found that 85% of beams can be removed from the codebook with zero-impact. By carefully selecting the usage of wide beams the codebook size can be further reduced to just 64 beams while still providing omni-directional coverage for an intersection. Other research thrusts have focused on attempting to augment or remove beam training entirely; however, this necessitates a change to the PHY layer. Codebook optimization achieves approximately 80% of the communications performance that would be achieved if beam training overhead could be completely removed while only requiring a radio configuration update. Thus, this work finds that today's commercial mmWave radios are sufficient for deployments in RSUs. To validate the proposed codebook optimization algorithm, a detailed mmWave ray tracing framework that encompasses 3D environmental information and material properties of reflectors is developed.**

***Index Terms*—Connected Vehicles, mmWave Networks, Machine Learning**

I. INTRODUCTION

Advancements in Autonomous Vehicles (AVs) and Advanced Driver Assistance Systems (ADASs) are ushering in a new era of roadway safety and efficiency beyond what human drivers are able to achieve alone. However, the current generation of these technologies are primarily geared towards decision making using only the information gleaned from on-board sensors and prior knowledge (such as an HD Map). In order to meet ambitious policy goals, such as zero roadway deaths [1], it may be necessary for vehicles to work in harmony, both with one another, and with the roadway infrastructure (e.g. Street IoT cameras and LiDARs). This allows the utilization of multiple vantage points to collaboratively perceive the environment in order to create enhanced roadway awareness as well as for vehicles and infrastructure to coordinate their actions to increase roadway safety, not only for the vehicles, but also for other roadway users such as pedestrians and bicycles which are unable to directly benefit from technological advances in machine perception and autonomy. This roadway collaboration is the motivation for solutions such as Cellular Vehicle-to-Everything (C-V2X) communications; yet, while current use cases of C-V2X are limited to exchange of small messages, future use cases that require the exchange of dense sensor information, such as camera feeds or point clouds, will necessitate data rates – potentially GBPS per user [2] – that current C-V2X solutions are simply unable to provide. Millimeter-Wave (mmWave) communications are increasingly popular solutions for providing high data rates and low latency links and have been included in the latest 5G and Wi-Fi standards.

Despite the clear benefits of mmWave, many of its drawbacks were previously thought to be exacerbated by vehicular use cases. For instance, while the increased path loss of mmWave can be more than overcome with the usage of directional beams from phased arrays, the high mobility of vehicles could lead to frequent beam misalignment and a simple beam training strategy (i.e. sweeping all possible beams to find the best configuration) could incur such a large overhead that mmWave communications is no longer fruitful. In order to eliminate, or greatly reduce, the beam training overhead, complex beam management solutions were developed that utilized deep learning to derive beam management policies by inferring the beams to use from vehicle locations [3] or camera data [4]. However, this work demonstrates that, with the proper codebook design, *simple beam management strategies are still effective for mmWave Roadside Units (RSUs)*.

Naive codebook designs must cover the entire angular space; yet, when mmWave radios are deployed to a static location, they can leverage the sparsity of mmWave channels and user distributions for fine tuning those codebooks. One research thrust is to design codebooks for low overhead channel estimation that can be used to select optimal beams from a larger codebook without necessitating beam training [5]. While this presents a promising research direction, it would require changes to the PHY layer to utilize this new channel estimation. Another thrust is to directly optimize the codebooks for communications by leveraging the static nature of a RSU to adapt to its environment and user distribution. This second approach has the advantage of being able to be easily deployed onto current Commercial Off-the-Shelf (COTS) radios, as many already support codebook configuration, and for that reason is the approach taken by this work. Prior

978-1-7281-7586-7/21/$31.00 ©2021 IEEE

work has focused on codebook adaptations that increase the Received Signal Strength (RSS) of users [6] and by extension this increases the data rate that each user can utilize.

In this work, we propose to create low overhead codebook designs by pruning away unused beams, and incorporating wide area beams when needed. Our codebook pruning algorithm leads to low beam searching overhead, as it uses less than 15% of beams from a naive codebook design. In addition, our algorithm judiciously uses wide beams that cover the user distributions where the RSS is already high and thus directivity gains can be traded for longer beam coherence times (the time between beam switches). By minimizing the overhead of beam training, the effective data rate of each user is increased, because a greater portion of channel time is devoted to communications rather than channel sensing. To verify the adaptive codebook design, we develop a high-fidelity simulation framework for mmWave enabled Connected Vehicles, which combines a wireless ray tracer with an open-source autonomous driving simulator (CARLA [7]). Our simulator faithfully models the mmWave signal propagation, as well as its interaction with different environmental objects (i.e. foliage, street posts, and building facades) and material types (i.e. glass, metal, and concrete). It allows for easy scripting and visualization of customizable scenarios.

This work focuses only on codebook design for mmWave RSUs and does not consider beam forming performed by the vehicles which can add additional overhead but provides higher RSS and can reduce inter-cell interference [8]. This is done because the hardware configurations (i.e. panel sizes and placements on vehicles) of users cannot be assumed as they are likely to vary. Furthermore, this work only considers codebook optimization for a single mmWave RSU for simplicity, but future works may explore whether there are benefits to jointly optimizing codebooks of nearby RSUs.

II. System Model

Our solution framework targets mmWave nodes placed along the roadside (RSUs) in order to facilitate high throughput, low latency, communications with vehicles. The RSU could enable both Vehicle-to-Infrastructure (V2I) or Vehicle-to-Network (V2N) applications such as collaborative perception or remote driving. Many works consider an RSU with a single panel that provides up to $180°$ coverage. In this work, we consider a deployment at an intersection that necessitates omni-directional coverage and therefore considers a multi-panel mmWave node [9] in order to provide this coverage (specifically, 4 panels are used with rotations of $90°$ between each panel as suggested in [10]). Each panel is assumed to contain a 16×16 phased array operating at 60 GHz and transmitting at 43 dBm, the maximum allowable by the FCC at this frequency [11]. The RSU is mounted at a height of 10.5m above the roadway.

A. Beam Management Protocol

Beam management protocols in use today typically have 5 steps: i) Base Station (BS) initiates a beam sweep where it transmits a reference packet, ii) User Equipments (UEs) listen to the beam sweep to determine which transmit beam provides the highest RSS, iii) the BS listens on each of the swept beams for responses from UEs, a random access channel, indicating their best beam for communication, iv) (optionally) beam refinement occurs, and finally v) communication occurs between BS and UE using the beams found through this training procedure. Step iv, beam refinement, can either allow the BS to determine a better beam to use (such as going from wide to narrow beams in a two level search scheme) or allow the UE to determine a suitable beam. The current work does not model this step because Codebook Optimization makes beam refinement by the BS unnecessary – thus, allowing simpler beam management protocols.

B. Achievable Data Rates

We model the MAC layer throughput by considering the bit rate achievable for a given RSS and the available percentage of channel time for communication after beam management. In short, the effective data rates are calculated with

$$\hat{R} = (1 - \text{OH})R(P_{rx}) \quad (1)$$

where $R(\cdot)$ is provided by a rate table that defines the Modulation and Coding Scheme (MCS) to use at a specified RSS. Beam training overhead (OH) is calculated as the percentage of channel time that is used for beam training. More specifically,

$$\text{OH} = \frac{(1 + N_{slots})N_{beams}T_{meas}}{T_{sweep}} \quad (2)$$

where the measurement time, $T_{meas} = 17.84\mu s$ as defined in the 5G NR standard, and the number of random access slots per beam, N_{slots}, was assumed to be 4. The number of beams swept, N_{beams}, and the beam sweep interval, T_{sweep}, are both floating variables used in performance evaluation. The 5G NR default configuration is a beam sweep interval of $20ms$ while the default interval for WiFi is $100ms$. 5G specifies that only 64 beams can be swept per interval which means that typically not all possible beams can be swept; while the current work utilizes this constraint for determining the maximum possible codebook size, it also relaxes it for easier comparison with the performance of larger codebooks.

III. Modeling Dynamic mmWave V2I Scenarios

There are two predominant types of methodologies for channel modeling. In Sub-6 GHz communications, stochastic channel models are widely used for evaluating algorithm performance. They're computationally efficient, but lack the ability to show how leveraging site specific information can be exploited for enhanced radio control or site planning. In mmWave communications, ray tracing has been shown to be quite accurate in modeling a wireless environment [8]. It is conditioned upon the specific "3D Map" that defines the geometry of a scene, and so can model a specific cell site instead of a generic channel model – allowing demonstration of techniques that predict optimal beams from UE locations [3]. Going beyond the wireless domain, the environment

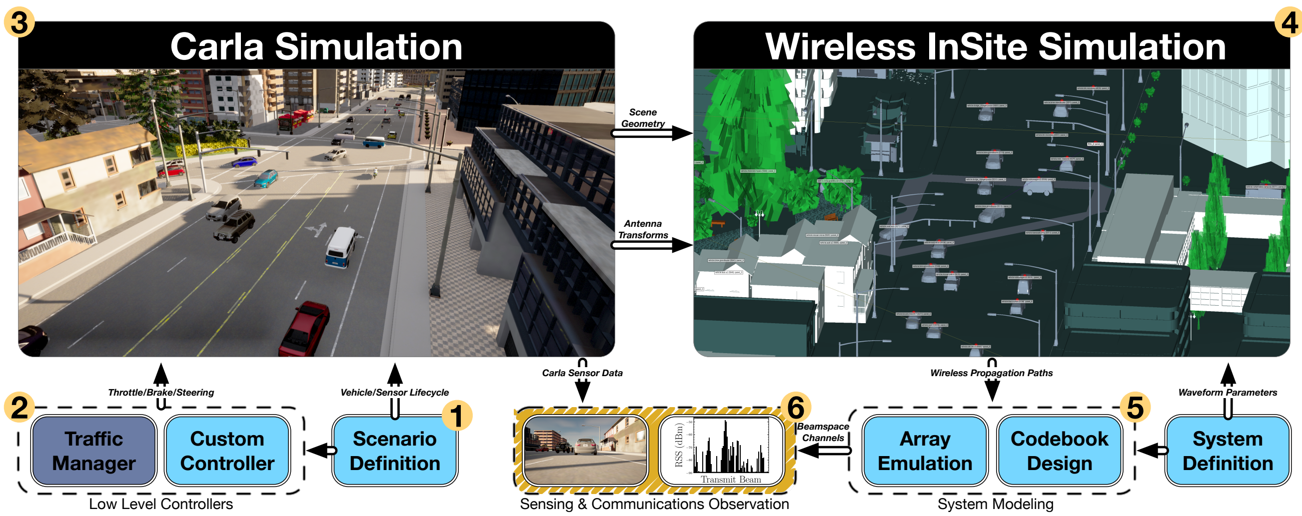


Fig. 1: Block diagram of the simulation setup for mmWave data generation.

models can also be extended to provide additional sensing modalities such as cameras and LiDARs for exploring how multi-modal sensing can aid in radio control [6], [12].

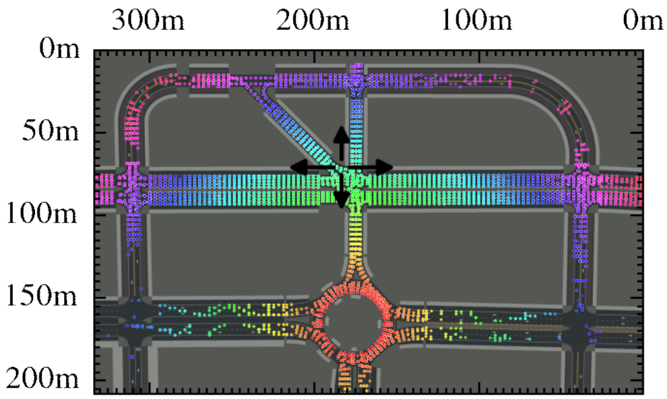
A. Methodology

An overview of the developed simulation and data generation methodology is provided in Figure 1. The key contribution is the ability to simultaneously leverage two widely used simulators, CARLA [7], an Open Source Autonomous Driving Simulator that provides the models and mobility simulations of the physical world, and Wireless InSite [13], a ray tracing simulator that provides wireless channels between transmitters and receivers in a specific 3D environment. Each simulator is ran at discrete time steps in a synchronous fashion, allowing for extraction of sensing (i.e. RGB/LiDAR/Depth) observations and observations about the wireless system's operation (i.e. a beamspace RSS Matrix) that describe the same instant of time in the same scenario.

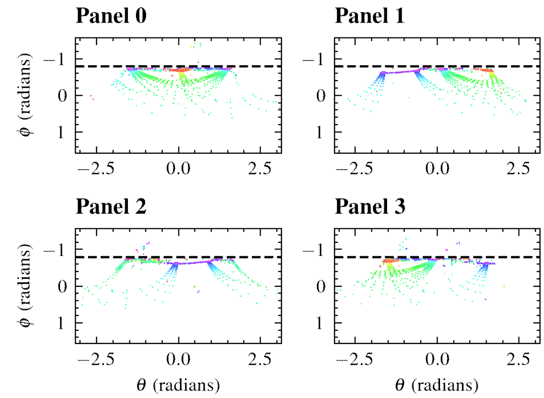
CARLA has a set of community defined maps, or "towns", that allow for simulation of vehicles in various urban, suburban, rural, or highway scenarios. The current work uses "Town03", which is a urban scenario featuring a 5-way intersection and a roundabout. Rather than simulate the entire town, which would be needlessly computationally expensive as a single mmWave radio cannot provide coverage for the entire town, a study area is defined that centers at the 5-way intersection (the deployment scenario can be viewed in Figure 2a). CARLA facilitates multiple methods for defining a scenario, three different ways were used in the current work: i) initializing traffic randomly within the study area and using CARLA's autopilot to drive the vehicles until they leave this area, ii) creating a simple waypoint following algorithm that utilized CARLA's low level control (e.g. throttle/brake/steering) to drive a pre-defined route, and iii) utilizing CARLA's wrappers around the OpenDRIVE map to cover the entire possible user distribution with receive antennas. Two modifications were necessary to CARLA in order to add the ability to co-simulate wireless communications. First, we need to extract the static meshes of the environment at any given time step. To meet this requirement, we add an additional Remote Procedure Call (RPC) server to the CARLA simulator that would provide all, or a queried subset, of the meshes within the simulator. A Python client, running in lock step with CARLA's already exposed Python API, can then query the simulator to extract the static meshes within the environment – this includes the $<x, y, z>$ coordinates of the triangular meshes along with meta-data that aids in material hinting as will be outlined below. Second, CARLA provides the capability of placing sensors anywhere in the environment, or even attaching them to other actors such as vehicles. In order to reuse this functionality, we create a "dummy" sensor within the server for mmWave that would allow for tracking the locations and orientations of all antennas placed within the environment, even as they changed due to vehicle mobility.

At each discrete time step, the scene geometry is extracted from the CARLA simulation and used to create project feature files for a Wireless InSite simulation[1]. Each of the meshes have a set of material hints extracted as well that help to form an imperfect matching to potential properties of them in the Radio Frequency (RF) domain. For instance, meshes labeled with asphalt or concrete can be matched with a concrete material in the Wireless InSite simulation; similarly, foliage or palm can be matched with a foliage material. Therefore, the developed simulation has material diversity consisting of brick, concrete, foliage, grass, dry earth, glass, metal, and wood – each material has a unique color in the Wireless InSite simulation shown in Figure 1. Each RSU panel is modeled as a separate transmit antenna and each of the vehicles in the simulation include a single receive antenna placed on the roof of the vehicle. The location and orientation of these antennas are extracted from the CARLA simulation and juxtaposed with a set of common parameters such as the transmit power in

[1]The static features within the environment (e.g. buildings) can be collected once and only dynamic objects (e.g. vehicles) need to be updated at each step.



(a) Road map of the study area considered in the current work.



(b) Paths within the Angle of Departure for each RSU panel.

Fig. 2: The RSU is placed next to a 5-way intersection and includes four panels, with their headings indicated as arrows in 2a, utilized to provide omni-directional coverage. The scatter plot on top of the road map indicates potential UE locations that could be covered by this RSU (i.e. they contain propagation paths with the potential for positive SNR). The color scheme indicates the distance from the origin and is meant to provide continuity between 2a and 2b to visualize the relationship between the angular domain, where beamforming decisions are made, and the spatial domain, where vehicles exhibit mobility.

order to create the set of transmitters and receivers for the Wireless InSite simulation. Wireless InSite then provides the propagation paths between each transmit and receive pair and these paths are post processed to create beam space channel matrices used in performance evaluation.

B. Related Work

Broadly, modeling dynamic scenarios using ray tracing requires the interfacing of five interconnected components: i) a geometry definition (meshes, materials, BS transforms, etc.), ii) a mobility model (traffic simulation, drone mobility, pedestrian movement, etc.), iii) a ray tracing engine (i.e. Wireless InSite [13]), iv) hardware emulation (i.e. phased array beam performance), and (optionally) v) rendering multi-modal sensing data describing the current scene. The current work extends prior work by presenting new approaches to components i, ii, and v.

Some prior works [8], [14] have used 3D models from Open Street Map to create site specific geometry definitions; however, these works utilize simplified models of the environment that do not include potential static blockages such as foliage or street lamps. They model the sides of buildings as completely flat surfaces which could lead to unrealistic reflections, and assume that the entire world is made of concrete and thus lacks material diversity. The current work remedies all of these limitations for a higher fidelity environment at the cost of no longer having a model based on a specific real world location but rather a representative deployment scenario. Additionally, prior works placed the RSU on the side of a single street; the current work models a 5-way intersection that requires omni-directional coverage from the RSU.

All prior works utilize SUMO [15] to model vehicular traffic (the interfacing of SUMO and Wireless InSite was first developed in [14]). We extend the ability to model mmWave communications in large scale representative traffic scenarios with the ability to script *specific* scenarios. Using CARLA's vehicle control API, we conduct a virtual drive test of specific routes using a waypoint following algorithm.

IV. Codebook Optimization

Practical mmWave radios utilize analog beamforming over a pre-defined set of possible beamforming weights – a codebook. At runtime, the radios search within this codebook to find the best possible configuration (i.e. the beam with the highest RSS). This process must be repeated periodically as the decisions made are no longer valid due to rotation/translation of UEs or blockages. Thus, a *good* codebook would provide high RSS to all users, take limited time to search through (e.g. it's small), and be robust to UE mobility.

Existing work mostly adopts generic codebooks with beams equally spaced within, and fully covering, the possible beamforming directions in the angular domain. However, site-specific codebook optimization can occur for an RSU deployment by leveraging knowledge about the UE distribution and wireless propagation characteristics of the environment. A *good* codebook optimization algorithm should be sample efficient and only utilize data that can be easily collected in the field. This section describes an algorithm that meets this criteria – it only needs to sample the UE distribution once and only requires RSS measurements.

A. Pruning Unused Beams

As can be seen in Figure 2b, the UE distribution for an RSU is incredibly sparse. This owes to the fact that the majority of users are concentrated "on the horizon". Thus, a beam pointing down a straight road, such as a city block, will cover a large amount of road area. Indeed this was also observed in our virtual drive test where the beam switches necessary to maintain optimal RSS are infrequent, typically only needing to occur on the seconds scale, when the vehicle is further away

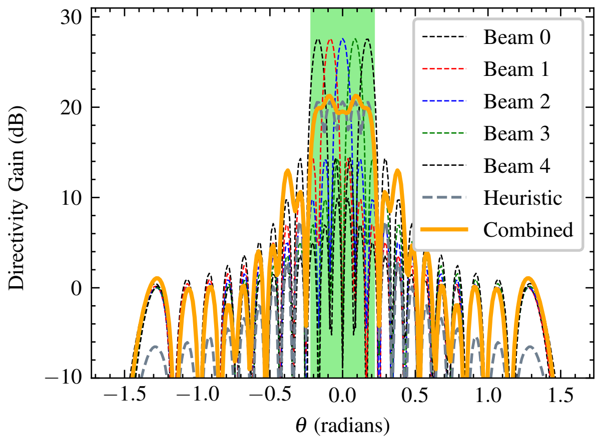


Fig. 3: Example of combining five narrow beams into a single wide beam, covering the same angular area (shaded in green), whose performance can be closely approximated by a heuristic that uses only RSS estimates of the original narrow beams.

from the RSU. The potential UE locations nearby the RSU exhibit more angular diversity, but the lanes of traffic can still clearly be seen in Figure 2b, which lead to the UEs still being relatively tightly clustered within the angular domain. While potential Non Line-of-Sight (NLoS) paths exist, which can be easily spotted by the discontinuities in color of Figure 2b, they are rare. Furthermore, each panel has some overlap in coverage with other panels due to their placement at 90° angles from one another. Therefore, the majority of beams (specifically, 85% in this work) from a naively constructed codebook for RSUs can be pruned as they cover directions with no UE distribution. Determining which beams can be safely pruned is quite simple. If a beam never has the highest RSS within the collected dataset, then it can be removed as it would never be chosen during beam training anyways.

B. Combining Narrow Beams into Wide Beams

While narrow beams provide the highest directivity gains their narrow beam widths provide lower area coverage than a wider beam could; thus a trade off exists between achievable RSS and beam training overhead. This section describes how to create wide area beams and how to estimate their achievable RSS from measurements consisting of only the initial narrow beams. The following section describes the process for determining when to include one or the other within a codebook for optimal RSU performance.

While there are many ways to determine beamforming weights, one of the easiest is the utilization of gradient descent. Every narrow beam can be considered to be providing coverage over a subset of the possible angular directions, i.e. the angular area where the directivity gain is within 3dB of the peak. By combining the angular area covered by multiple narrow beams, a wide beam can be created for their replacement by choosing beamforming weights that maximizes the directivity gain within that same area. Figure 3 shows an example case of creating a single, combined, beam to cover the same area as multiple narrow beams. Specifically, the current work utilizes the following optimization equation and performs the gradient based optimization.

$$\begin{aligned} \max_{\vec{w}} \quad & \mathrm{E}_{\theta\sim\mathcal{C}}[|g(\theta,\vec{w})|^2] - \frac{\mathrm{VAR}_{\theta\sim\mathcal{C}}[|g(\theta,\vec{w})|^2]}{\mathrm{E}_{\theta\sim\mathcal{C}}[|g(\theta,\vec{w})|^2]} \\ \text{s.t.} \quad & |w_i| = 1 \quad \forall i \end{aligned} \tag{3}$$

In Eq. (3), θ represents an angular direction and $\mathcal{C}$ represents the set of angular directions that the beam should cover. It was found that simply maximizing the expected directivity gain, $\mathrm{E}[|g(\cdot)|^2]$, was unstable and thus the normalized variance was used to ensure all directions within the set were equally valued. The constraint, $|w_i| = 1$, ensures that power is not added to the system and enforces that the system being modeled is a uniform amplitude array. The current work does not consider the impact of quantization on phase shifters. Similarly, it is assumed that the antenna elements of a phased array are distributed at $\frac{\lambda}{2}$ spacing and the current work does not model hardware imperfections when creating the beamforming weights or performing the evaluations. The directivity gain of the synthesized wide area beam can be estimated from the directivity gains of the narrow beams it is replacing. A heuristic is used in the current work to estimate the wide area beams performance, $\hat{p}$, as

$$\hat{p} = \frac{\max p_0, \ldots, p_n}{n} \tag{4}$$

where p represents the measurable receive power of the narrow beams. In Figure 3 it can be seen that this is a quite accurate approximation within the coverage region. Although the heuristic underestimates the directivity gain outside of this region, that is not a significant concern as another beam would be used to cover UEs in that direction anyways. In order for this estimate to remain accurate from channel measurements, multi-path effects must remain minimal, which is typically the case in outdoor mmWave due to channel sparsity.

C. Knapsack Optimization

Through a mixture of channel measurements of narrow beams and performance estimation of wide beams, a dataset can be created that estimates the utility of a large set of potential beams over the entire UE distribution served by an RSU. This section details how to select *which* of those potential beams should be included in a codebook for maximum benefit. This can be formulated as a 0/1 knapsack problem with the goal of solving the optimization equation

$$\begin{aligned} \max_{x} \quad & \sum_{i=0}^{n} v_i x_i \\ \text{s.t.} \quad & \sum_{i=0}^{n} x_i \leq N \\ & x_i \in 0, 1 \end{aligned} \tag{5}$$

where v_i is the *value* of beam i, x_i is an indicator of whether this beam is included in the codebook, and the codebook size is restricted to N.

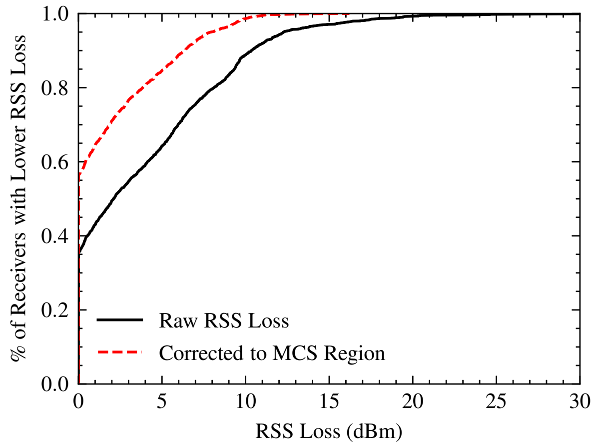


Fig. 4: Cumulative Distribution of reductions in achievable RSS, compared to a standard DFT designed codebook, due to codebook optimization. When a user's RSS falls outside of the defined MCS ranges, such as when RSS is extremely high, there is nearly zero impact to the achievable bit rate – the red line accounts for this to show that over half of receivers will experience no effective degradation in RSS.

Determining the value of any beam is difficult. One way would be to utilize some aggregate of the achievable RSS across the entire UE distribution; however, there are two issues with this approach. First, RSS is an imperfect estimator of performance, a better indicator is the bit rate achievable which is a non-linear function of RSS. This could be modeled with a channel capacity equation, but a more accurate representation is to utilize the rate table of a specific protocol – the current work utilizes the 802.11ad rate table. Each protocol defines the MCS to use as a function of SNR or RSS and the MCS roughly indicates the achievable bit rate. Any RSS lying outside of the MCS ranges will not translate to additional performance and therefore utilizing a rate table instead of RSS alone presents a more accurate depiction of RSU performance during operation. Second, a beam only provides value if it is used for communication. If a better beam exists in the codebook then it will be used instead and there is no value to including a beam that can provide a lower rate for a specific location. Therefore, a beam's value is dependent upon the codebook it is included in and this must be repeatedly updated throughout optimization. The value of a beam is then the aggregate additional rate it can provide over the UE distribution.

The algorithm used for knapsack optimization is shown in Algorithm 1. At each epoch, each beam's values are updated based on the currently selected codebook as described above. The beam with the highest value is then added to the codebook and the process repeats. In order to ensure that beams covering the same area (i.e. a wide beam and any of the narrow beams it is meant to replace) are not selected by the algorithm, they are expressly forbidden. The algorithm terminates when the codebook reaches the maximum size. The current work restricts the codebook to a size of 64 and only includes wide area beams meant to replace two narrow beams.

Algorithm 1: Codebook Knapsack Optimization

Input: Per Beam RSS Estimates ($\boldsymbol{P}$), Maximum Codebook Size (N), Rate Mapping Function ($R(\cdot)$)
Result: Selected Beams

```
selected = []
forbidden = []
V = update value (selected, forbidden, P, R(·))
selection = argmax V
while len(selected) < N do
    selected.append(selection)
    forbidden.extend(all beams that overlap selection)
    V = update value (selected, forbidden, P, R(·))
    selection = argmax V
end
```

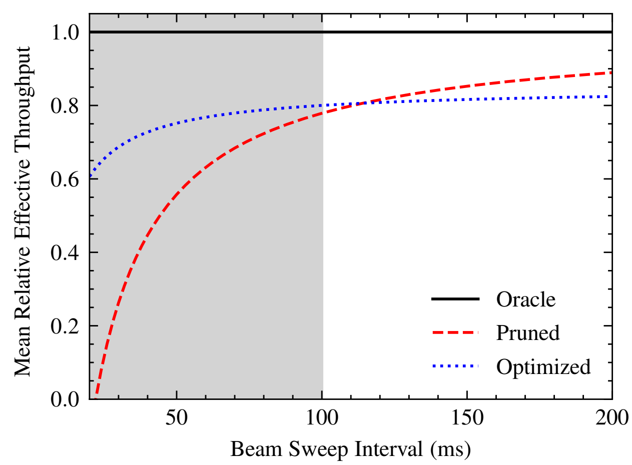
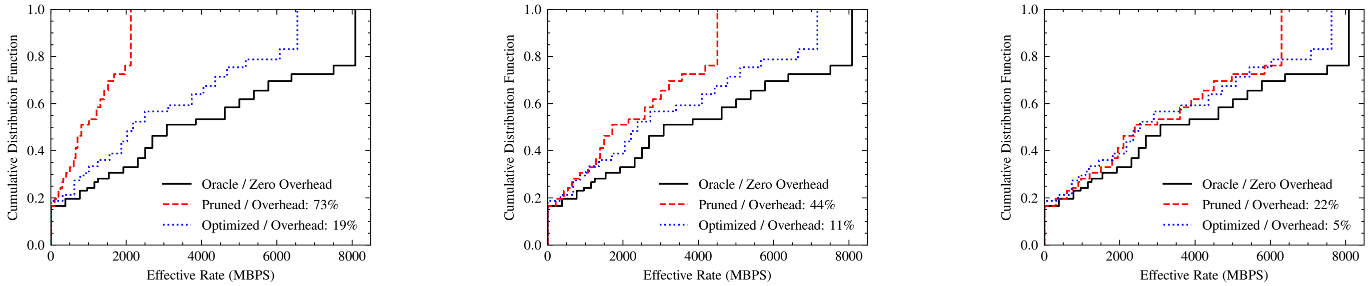


Fig. 5: Mean relative performance compared with an idealized zero-overhead oracle solution. Realistically the beam sweep interval in use will fall within the grey area as this is bounded by the default configuration for 5G (20ms) and WiFi (100ms).

V. Results

There are three key factors affecting the performance of a codebook design: i) the best RSS achievable for the receiver (Figure 4), ii) the loss in communication time due to the overhead of beam training (Figure 5 and 6), and iii) the impact of suboptimal beam decisions caused by mobility. As shown in Figure 4, despite the huge reduction in number of beams (1765 original beams vs. 64 beams in the optimized codebook), over half of receivers will not experience any degradation in RSS. Therefore, using a pruned codebook can achieve nearly 80% of the performance available to an oracle solution (Figure 5) that achieves the best RSS possible but takes zero overhead.

Figure 5 shows that the benefits of lower beam training times from the smaller pruned codebooks have diminishing returns as the beam sweep interval increases. This is due to the overhead of longer beam training times becoming negligible when it occurs infrequently and thus the increased bit-rates due to potentially higher RSS becomes a factor. While the



(a) 30 ms Beam Sweep Interval (b) 50 ms Beam Sweep Interval (c) 100 ms Beam Sweep Interval

Fig. 6: Cumulative Distribution of effective rates, a function of the achievable bit rate due to RSS and the overhead of beam training, among various codebook and beam sweep interval choices.

necessary beam switching times due to mobility can be on the order of seconds, as was discovered in our virtual drive test, it is unlikely that a beam sweep interval would be set higher than 100ms. The beam sweep serves a dual purpose of beam training as well as allowing for the initial attach of UEs to the mmWave network; an excessively long beam sweep interval could lead to delays for UEs entering the network. When below 100ms, the lower overhead from the optimized codebook outperforms the simple pruning strategy that results in zero loss to achievable RSS, but still has high overhead due to the need to measure 248 beams from its larger codebook. While the pruned codebook achieves a huge overhead reduction from a naive codebook design (1765 original beams to 248 pruned beams), it would still be incapable of running at the default 5G beam sweep interval of 20ms, whereas the optimized codebook is still able to achieve non-trivial bit rates even with these frequent beam sweeps.

Only looking at relative performance values can be overly pessimistic about the currently achievable performance. Figure 6 shows that, even though there is still a performance gap between achievable rates with an optimized codebook and an oracle solution, a large percentage of users can still achieve multi-GBPS communications. Advanced V2X use cases that necessitate high data rates, such as those that utilize sharing of raw sensory data for collaborative perception, can be enabled today with only a radio configuration update.

VI. Conclusion

In this work, we have demonstrated that mmWave V2X communication can be improved through an optimized codebook design that achieves low overhead beam management without requiring the need for PHY layer modifications. The codebook optimization algorithm only depends upon beamspace RSS measurements instead of necessitating the estimation of channel state information for all elements of the array. The algorithm is validated through a detailed mmWave ray tracing framework that encompasses 3D environmental information and material properties of reflectors. While codebook optimization alone still leaves additional optimization space relating to beam management, it means that *today's COTS mmWave radios are sufficient for deployments in RSUs*.

Acknowledgement

This material is partially supported by the National Science Foundation CCRI program under Grant No. CNS-1925767 and the UC San Diego Center for Wireless Communications.

References

[1] Vision Zero Network, "Making our streets safer." https://visionzeronetwork.org, 2021. Accessed: 2021-03-15.

[2] 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.

[3] 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.

[4] 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.

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

[6] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Learning beam codebooks with neural networks: Towards environment-aware mmwave mimo," 2020.

[7] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, pp. 1–16, 2017.

[8] S. Wang, J. Huang, and X. Zhang, "Demystifying millimeter-wave v2x: Towards robust and efficient directional connectivity under high mobility," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, MobiCom '20, (New York, NY, USA), ACM, 2020.

[9] Y. Huang, Y. Li, H. Ren, J. Lu, and W. Zhang, "Multi-panel mimo in 5g," *IEEE Communications Magazine*, vol. 56, pp. 56–61, March 2018.

[10] Telecom Infra Project, "Tip playbook for smart cities." https://telecominfraproject.com/tip-playbook-for-smart-cities/, 2021. Accessed: 2021-03-15.

[11] Federal Communications Commission, "15.255 Operation within the band 57– 64 GHz.," October 2002.

[12] A. Klautau, N. González-Prelcic, and R. W. Heath, "Lidar data for deep learning-based mmwave beam-selection," *IEEE Wireless Communications Letters*, vol. 8, pp. 909–912, June 2019.

[13] Remcom, "Wireless EM propagation software." https://www.remcom.com/wireless-insite-em-propagation-software, 2020. Accessed: 2021-03-15.

[14] A. Klautau, P. Batista, N. Gonzalez-Prelcic, Y. Wang, and R. W. Heath Jr., "5G MIMO data for machine learning: Application to beam-selection using deep learning," in *2018 Information Theory and Applications Workshop, San Diego*, pp. 1–1, 2018.

[15] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*, IEEE, 2018.