Cross-Layer Optimization for Off-Network Public Safety Communications in 4G LTE and 5G NR

Collin Brady

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2023

Reading Committee:
Sumit Roy, Chair
Payman Arabshahi
Tom Henderson

Program Authorized to Offer Degree:
Electrical and Computer Engineering

University of Washington

Abstract

Cross-Layer Optimization for Off-Network Public Safety Communications in 4G LTE and 5G NR

Collin Brady

Chair of the Supervisory Committee:
Sumit Roy
Electrical and Computer Engineering

As public safety migrates from existing digital Land Mobile Radio (LMR) networks to Third Generation Partnership Project (3GPP) based FirstNet, an issue of key concern is coverage. To handle off-network scenarios, 3GPP has introduced Sidelink, a communication link that facilitates direct communication between user equipment (UE). In this thesis, we analyze and propose improvements to the two standards which constitute the Sidelink, Fourth Generation (4G) Long Term Evolution (LTE) Proximity Services (ProSe) and Fifth Generation (5G) New Radio (NR) Cellular Vehicle-to-Anything (C-V2X).

As both standards are relatively new and poorly understood, we attempt to model them in terms of their Key Performance Indicators (KPIs). These models allow us to validate the performance of the network simulator 3 (ns-3) implementations of these standards and to determine how best to set the Physical (PHY) and Medium Access Control (MAC) layer parameters to optimize the KPIs. The challenge in any real-world scenario is that some number of PHY parameters that contribute to the performance of a KPI (e.g., the number of UE participating in the ad hoc network) are hidden from UEs, hindering the optimal setting of PHY/MAC parameters. To deal with this challenge, we use the previously developed models to aid in learning the hidden parameters during operation and set PHY/MAC parameters according to those approximations, thereby improving performance.

In ProSe, we examine the performance of the ProSe Device-to-Device (D2D) direct discovery process in out-of-coverage scenarios. We model individual discovery periods as a slotted random access protocol with half-duplex (HD) UE and the entire discovery process as a modified coupon collectors problem. We use the open-source network simulator 3 (ns-3) to validate our model and evaluate the discovery process's performance as a function of the size of the resource pool, UE density, and transmission probability. We establish there exists an optimal transmission probability that minimizes discovery time for a given network configuration and develop a method to allow UEs to learn the optimal transmission probability during discovery by estimating the hidden parameters.

For C-V2X, we present a novel distributed blind retransmission algorithm in out-of-coverage (mode 2) scenarios. Our contribution is intended as an enhancement to standard (3GPP) specified sensing-based Semi-Persistent Scheduling (SPS) resource re-allocation by opportunistically reusing blind retransmissions to improve average per-user throughput for UE-to-UE transmissions. We initially develop a detailed cross-layer model for (per user) throughput of 5G NR mode 2, supported by simulations using the reputable open-source ns-3 (www.nsnam.org) network simulator. This model and simulation analysis provided important insights into the underlying causative trends and led to two novel distributed adaptive algorithms for resource selection: an initial standards-compliant Dynamic Retransmission (D-Re) and a final standards non-compliant Opportunistic Retransmission (O-Re) algorithm. We show that both D-Re and O-Re render average per-UE throughput robust even in the absence of direct knowledge of 5G network parameters, while O-Re significantly improves the averaged per-UE throughput at high UE densities.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

Chapter 1

Abbreviations

3GPP: 3rd Generation Partnership Project

4G: 4th Generation

5G: 5th Generation

C-V2X: Cellular Vehicle-to-Anything

DMRS: Demodulation Reference Signal

D-RE: Dynamic Retransmissions

FDD: Frequency Division Duplex

HD: Half Duplex

LMR: Land Mobile Radio

LTE: Long Term Evolution

MAC: Medium Access Control

NS-3: Network Simulator 3

NR: New Radio

O-RE: Opportunistic Retransmissions

PROSE: Proximity Services

SPS: Semi-Persistent Scheduling

SINR: Signal to Interference and Noise Ratio

UE: User Equipment

Table 1: Chapter 1: List of Important Symbols

Symbol	Definition
θ	Transmit probability for one UE in LTE ProSe Discovery
N_{Se}	Number of transmissions (including HARQ retransmissions) for each transport block in NR C-V2X SPS

Chapter 2

Abbreviations

3GPP: 3rd Generation Partnership Project

4G: 4th Generation

CCTC: Centre Tecnològic Telecomunicacions Catalunya

CDF: Cumulative Density Function

D2D: Device-to-Device

EPC: Evolved Packet Core

FDD: Frequency Division Duplex

HD: Half Duplex

LTE: Long Term Evolution

MCM: MAC Collision Model

MAC: Medium Access Control

NIST: National Institute of Standards and Technology

NS-3: Network Simulator 3

PRB: Physical Resource Block

PSBCH: Physical Sidelink Broadcast Channel

PSCCH: Physical Sidelink Control Channel

PSDCH: Physical Sidelink Discovery Channel

PSSCH: Physical Sidelink Shared Channel

PMF: Probability Mass Function

RAN: Radio Access Network

SGD: Stochastic Gradient Descent

UE: User Equipment

Chapter 3

Abbreviations

3GPP: 3rd Generation Partnership Project

5G: 5th Generation

BSM: Basic Safety Message

C-V2X: Cellular Vehicle-to-Anything

D-RE: Dynamic Retransmissions

FTR: Finite Transmission Range

Table 2: Chapter 2: List of Important Symbols

Symbol	Definition
N_f	Number of resource block pairs in the resource pool
N_t	Number of sub-frame sets in the resource pool
N_r	Number of PRB in the resource pool, $N_r = N_f N_t$
N_{UE}	Number of UE in the discovery process
θ	Transmit probability for one UE
N_p	Number of UE that are previously discovered from the perspective of a reference UE
N_{DCT}	Number of periods until discovery completion time
K	The number of UE discovered in a single discovery period, messages from UE already discovered are not considered discoveries
N_m	Total number of discovery messages decoded in a single discovery period
θ^*	The value of θ which maximizes $E[N_m]$ as a function of θ
θ^A	The value of θ converged to by the adaptive θ algorithm
N_{TCT}	The number of periods for the adaptive θ algorithm to conclude training

HD: Half Duplex

HARQ: Hybrid Automatic Repeat Request

MAC: Medium Access Control

MCS: Modulation and Coding Scheme

NS-3: Network Simulator 3

NR: New Radio

O-RE: Opportunistic Retransmissions

PRR: Packet Reception Ratio

PRB: Physical Resource Block

PSCCH: Physical Sidelink Control Channel

PSSCH: Physical Sidelink Shared Channel

RSRP: Reference Signal Receive Power

RB: Resource Block

RRI: Resource Reservation Interval

SPS: Semi-Persistent Scheduling

SL: Sidelink

SCI: Sidelink Control Information

SINR: Signal to Interference and Noise Ratio

TDD: Time Division Duplex

UE: User Equipment

V2V: Vehicle-to-Vehicle

ZID: Zone ID

Chapter 4

Abbreviations

4G: 4th Generation

5G: 5th Generation

C-V2X: Cellular Vehicle-to-Anything

D-RE: Dynamic Retransmissions

FDD: Frequency Division Duplex

HD: Half Duplex

LTE: Long Term Evolution

Table 3: Chapter 3: List of Important Symbols

Symbol	Definition
N_{Se}	Number of transmissions (including HARQ retransmissions) for each transport block
R_c	Reselection counter
T_{RRI}	Packet inter-transmission time, Resource Reservation Interval in seconds
ρ_{UE}	UE density, UE/m
γ_{SPS}	Total threshold on Reference Signal Received Power (RSRP) for a PRB to be excluded during SPS reselection, dBm
γ_{FTR}	Detection threshold, dBm
$d_{t,r}$	Distance between the transmitter and receiver, m
$\Lambda(d_{t,r}, N_{Se})$	Calculated per-UE throughput, $packets/(UE \cdot s)$
$P_{PRR}(d_{t,r}, N_{Se})$	Packet Reception Ratio
$\delta_{HD}(N_{Se})$	Probability of half duplex error event
$\delta_{FTR}(d_{t,r}, \gamma_{FTR})$	Probability of finite transmission range error event
$\delta_{COL}(d_{t,r}, N_{Se})$	Probability of collision error event
$\Lambda_{Sim}(d_{t,r}, N_{Se})$	Per-UE throughput generated from ns-3 simulations, $packets/(UE \cdot s)$
$\overline{\Lambda}_{Sim}$	Distance average per-UE throughput from ns-3 simulations from $0 \leq d_{t,r} \leq x$, $packets/(UE \cdot s)$
N^*_{Se}	The value of N_{Se} which maximizes $\overline{\Lambda}_{Sim}$
$\hat{\rho}_{UE}$	Estimate of ρ_{UE}, UE/m
d_{edge}	Distance from the center of the zone occupied by UE_t which constitutes the "edge" UEs[1], m
$N_{Se,O}$	Number of duplicate packets sent using the O-Re algorithm
N_{R_c}	Reselection counter of the overwritten packets when using the O-Re algorithm

NR: New Radio

NR-U: Unlicensed NR

O-RE: Opportunistic Retransmissions

PROSE: Proximity Services

PRR: Packet Reception Ratio

SPS: Semi-Persistent Scheduling

UE: User Equipment

Chapter 4: List of Important Symbols

Table 4: List of Important Symbols

Symbol	Definition
X	Resource Keep Percentage
R_c	Reselection counter

ACKNOWLEDGMENTS

I'd like to thank the University of Washington (UW) for the opportunity to pursue my Ph.D. and the resources to pursue it. I'd like to thank my Advisor, Dr. Sumit Roy, who provided me with endless hours of guidance on my thesis project. He provided me with a second chance at grad school when I probably didn't deserve one but showed me how to formulate good problems, produce good results, and write in a way that informed readers. I'd also like to thank Dr. Roy for helping me beyond academia with assistance securing a position in my future career. I'd like to thank Dr. Henderson for his endless support on all matters ns-3. Dr. Henderson was an invaluable resource (and one with infinite patience) who, without, I could never have made full use of ns-3, which is integral to producing results supporting my thesis. I'd like to thank Dr. Ritcey for helping me to secure a second chance at grad school after a bumpy beginning. I'd like to thank the other members of my advisory committee, Dr. Heimerl, Dr. Arabshahi, and Dr. Moazeni, for their time supervising me. I'd like to thank Dr. Kuga and Dr. Farquharson at UW for providing me with some practical life advice that helped shape the direction of my graduate studies in a positive way. I also thank the ECE graduate advisors (all of them over the years) for their advising and help in my Ph.D. program. I'd also like to thank my professors at my undergraduate university, the University of Illinois at Chicago. I'd like to highlight those professors who helped me get into graduate school by writing letters of recommendation, Dr. Erricolo, Dr. Ansari, and Chris Kuypers (RIP), along with some other professors who kept me engaged throughout undergrad, like Dr. Goncharoff (thanks for keeping me awake), Dr. Caliskan (Your puns still aren't funny sadly), and Dr. Metlushko (I've tried to remember that it's all super simple stuff).

During my time at UW, I did two internships at MIT Lincoln Laboratory (summers of 2018/2019, I got lucky to sit out 2020). I'd like to thank everyone at Lincoln Labs but highlight Dr. Carl Fossa, my group head, who has made the hopefully wise choice to continue my employment there, and Dr. Siamak Dastangoo, who mentored me through both internships. I'd like to thank all my labmates at UW Funlab, who helped me to acclimate at UW generally and in research more specifically.

I'd like to thank my friends and family for your continuous support throughout my graduate studies. There are far too many of you to name specifically, but thank you, Tom Phillips, Mike McConnell, and Matt Saxon. You all held me together through the roughest moment in my life (so far!). Thank you to the Seattle Magic the Gathering community (RIP Jordan Aisoka) for providing me with endless entertainment and the ability to unwind with other strange people like me. And last but in a physical sense least, I'd like to thank my cats Taiyaki and Mochi, who, by nature of being cats, can never read or comprehend this.

DEDICATION

To my father, to whom I told I would never be an electrical engineer.

Chapter 1

INTRODUCTION

1.1 Background

Release 12 of the 3rd Generation Partnership Project (3GPP) defined the concepts of Sidelink (SL) for direct device-to-device (D2D) communications, underpinning Proximity Services (ProSe) to support public safety applications, [2]. Public safety users have ultra-high-reliability requirements and are often called into situations with no or degraded network coverage. ProSe addressed this by creating mechanisms for D2D communications and user equipment (UE) relaying (a single hop patching off-network UE onto the network through an on-network UE). However, given the small market share of public safety users, ProSe-enabled chips are scarce [3]. The recent more promising development is a 3GPP proposal to adapt ProSE features for vehicular communications, which started with LTE-V (Rel-14) and has now matured into Fifth Generation (5G) New Radio (NR) based Cellular Vehicle-to-Anything (C-V2X).

Aside from economic issues, LTE ProSe also suffered from latency and throughput issues, see Tab. 2.1, caused by the channel only being accessible periodically and in sequence (control then data) in LTE modes 1 and 2. NR C-V2X, needing higher performance, elected to use what were originally LTE modes 3 and 4[1] to enable the simultaneous transmission of control and data to reduce latency and increase throughput. However, even with these improvements, C-V2X does not support all of the features from ProSe, most importantly relaying and its required discovery mechanism. A key concern with the transition to cellular technologies is reducing coverage relative to existing Land Mobile Radio (LMR) networks.

[1]Note that in NR C-V2X, and in Chapter 3, these transmission modes are referred to as modes 1 and 2 respectively.

From [4], "3GPP systems deliver less coverage area when compared to LMRs. Furthermore, they can also experience relatively low data rates and dropped communications at the cell edge". This lack of cellular coverage is especially noticeable in low-population or remote areas where existing cellular coverage is minimal. This leads explorations into the ProSe discovery mechanism to be of continuing relevance, as some sort of analogous feature needs to be built into C-V2X to ensure full functionality for public safety users.

Prior work on off-network ProSe discovery can be broadly divided into several categories; 1) modeling ProSe discovery has included [5–7] which include both mathematical and simulation-focused studies into the effects of various aspects of the discovery mechanism, and 2) proposed improved discovery algorithms, [8–10], including piggybacking discovery on another preexisting signal (in this case the demodulation reference signal (DMRS)) [8], applying spread spectrum like techniques to improve decode probabilities [9], and modulating discovery parameters during runtime to minimize discovery completion time, [10].

Use Case	Payload (bytes)	Tx Rate (messages/s)	Maximum End-to-End Latency (ms)	Reliability %	Data Rate (Mbps)	Required Communications Range (m)
Vehicle Platooning	50-6000	2-5	10-25	>90	≤65	80-350
Advanced Driving	SL:300-12000 UL: 450	SL: 10-100 UL: 50	10-100	>90	SL: 10-50 UL: .25-10 DL: 50	360-700
Extended Sensors	1600	10	3-100	>90	10-1000	50-1000
Remote Driving	16000-41700	33-200	5	99.999	UL: 25 DL: 1	≥ 1000

Table 1.1: NR C-V2X Use Cases and Requirements, [1]

Although NR C-V2X may, by its nature, have naturally lower latency and higher throughput than LTE ProSe, that does not mean it can yet achieve the required metrics for its imagined use cases. Tab. 1.1 shows the use cases as originally presented in [1]. Although these use cases are of a vehicular nature, they provide insights into the goals for various

metrics, which are also of relevance to public safety users; for the purposes of this thesis, the most relevant being reliability and data rate. The greatest challenge in off-network communications is the distributed nature of channel access, potentially causing collisions, see Fig. 3.2. NR C-V2X attempts to address this by introducing a distributed channel access mechanism called semi-persistent scheduling (SPS). The SPS scheme is used to monitor the channel and transmit only where the device perceives it to be available, similar in concept, but not execution, to carrier sense multiple access.

Studies of the C-V2X generally focus on the SPS scheme and can be categorized as; 1) efforts to model SPS throughput or latency through a variety of means [11–13], 2) simulation focused campaigns that attempt to understand the same trends as modeling, [14–17], through the creation of novel simulation frameworks [14, 15, 17], or through the use of preexisting ns-3 implementations ([18]) like [16], 3) or algorithmic improvements to the SPS algorithm, [11, 19–24], including modulating the resource reservation rate, [11, 21, 23], through the use of reinforcement learning, [19, 24], or through the use of the zone ID to partition the channel in some way based on a spatial aspect, [20, 22].

Despite the prior work on both LTE ProSe discovery and NR C-V2X SPS, there are still important unexplored problems. Although both technologies have separate problems, a lack of accurate and feature encompassing modeling limits our ability to understand the process and properly improve it. In both cases, existing modeling tends to oversimplify the process, thus precluding certain insights be gained from them. Further, previously proposed algorithms tend to rely on either significant changes to the standards[2] or not work in practice because of some restrictive assumptions[3]. These deficiencies lead to the desire to improve 1) modeling of and 2) algorithms for both LTE ProSe discovery and NR C-V2X SPS.

Our work on both LTE ProSe and NR C-V2X are linked through their goals of minimizing latency. For Prose, the connection to latency is quite direct. The goal of Chapter 2 is to

[2]Introducing spread spectrum, [9], modifications to the DMRS, [8].

[3]foreknowledge of some hidden parameters, [6, 20, 22], additional features not present in off-network contexts, [19, 24], lack of control of certain parameters by the MAC [11, 21, 23].

minimize the discovery process completion time. This is an overt form of latency (the time between beginning discovery and decoding the last discovery message) minimization. My work in Chapter 3's relation to throughput is more circuitous. As We will show in Chapter 3 the key method of improving NR C-V2X throughput is by improving the packet reception ratio. The key latency metric in NR C-V2X is packet inter-reception time, or the time taken between receiving unique packets[4] Increasing the packet reception ratio, in addition to improving throughput, has the added effect of reducing the packet inter-reception time because new packets are more often received, thus minimizing latency.

1.2 Summary of Research Contributions

As part of studying cellular public safety communications, we evaluate two key technologies, Discovery in LTE ProSe and SPS in NR C-V2X. In LTE ProSe discovery, the objective is to minimize the discovery completion time, that is, the time taken to decode discovery messages from all nearby UEs. For NR C-V2X, our explorations of SPS are focused on improving the KPIs presented in [25], specifically packet reception ratio and throughput.

In Chapter 2, we consider the LTE ProSe discovery problem. To analyze the discovery completion time, that is, the number of discovery periods for a reference UE to decode packets at least once from all other UEs in the local region, we begin with a multinomial model of packet decodes based on a simple MAC collision assumption (packets using time-frequency resources exclusively are decoded with probability 1, and 0 otherwise). This model of a single discovery period allows us to extend this model to the entire discovery process using a modified coupon collectors problem. We then use our model to discover 1) the effects of the duplexing assumption on LTE ProSe discovery, specifically by how much relaxing the half-duplex effect (the ability for UEs to transmit and decode packets simultaneously) to a frequency division duplex effect (UEs can transmit and decode at the same time, so long at the other UE does not use the same frequency) affects the discovery completion time, and 2)

[4]This is because there is no HARQ and no repeated packets in broadcast mode.

how the choice of underlying "set-and-forget" discovery parameters can be used to optimize the discovery process, given certain hidden PHY parameters (chiefly UE density) are known. We show that there exists a parameter θ, which governs the probability of transmission in a discovery period for a given UE density, that minimizes the discovery completion time. An algorithm is presented using the stochastic gradient descent method, which is capable of learning the underlying unknown UE density using our model of discovery completion time after a training phase. With this, we can adjust θ during run time to minimize discovery completion time under any circumstances. We use both Matlab and ns-3 to validate the modeling work and test the performance of our proposed algorithm, comparing it to both the standards algorithm and one proposed in [10]. We show that the proposed algorithm improves performance over both the standards algorithm with poorly set θ (a challenge given unknown UE density) and over the algorithm presented in [10].

In Chapter 3, we consider the problem of maximizing throughput and packet reception ratio, which has the effect of minimizing latency. The focus of this model is on the ways in which packets fail to be decoded; through half-duplex effects, UEs are assumed to be half-duplex and can't transmit and receive simultaneously, through finite transmission range error, UEs must perform a form of energy detection before packet decoding, low receive power results in a form of energy detection error, and collisions, when two or more UEs transmit using the same time-frequency resources another listening UE can fail to decode any if the SINR fall too low. The majority of the modeling is focused on collision errors, an issue of high importance in off-network communications. With the model of throughput, we learn that the parameter N_{Se}, which controls how many duplicate copies of a packet are transmitted at each transmission opportunity, can be used to affect the packet reception ratio through repetition coding gains and further that an optimal value exists for N_{Se} for a known UE density. We propose an algorithm called dynamic retransmission (D-Re) to tune N_{Se} during runtime. The procedure and premise are similar to the algorithm proposed for LTE ProSe discovery. By learning the underlying UE density, we can tune a parameter that affects the density of packets in the channel. The primary innovation over our discovery algorithm is

the elimination of the training phase, with the D-Re algorithm training continuously during operation. Going beyond the D-Re algorithm, we propose a second, non-standards-compliant algorithm called opportunistic retransmissions (O-Re). The O-Re algorithm involves selectively overwriting the packets of UEs at the edge of a given UE's ability to decode packets under certain circumstances. The goal of this overwriting is to, over the entire sequence of packets transmitted in the same resource, reduce the probability of collisions by forcing the time-frequency resource to have a continuous occupant. We use both Matlab and ns-3 to validate the modeling work and test the performance of our proposed algorithm, comparing it to the standards algorithm. We show that the D-Re algorithm improves performance over a poorly chosen N_{Se} but can never exceed the standards algorithms performance with a perfectly set N_{Se} while we further show that the O-Re algorithm maintains the robustness qualities of the D-Re algorithm it can exceed the throughput of the standards algorithm at high UE densities where the probability of overwriting UEs at the edge is higher.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. In Chapter 2, we model and analyze the ProSe discovery process, proposing robusitification techniques and evaluating the effects of certain receiver assumptions. In Chapter 3, we model and analyze NR C-V2X throughput, proposing two algorithms to improve throughput. This thesis includes the materials from the author's previously published paper, [26–28], and one accepted paper, [29].

Chapter 2

4G LTE PROXIMITY SERVICES DEVICE-TO-DEVICE DIRECT DISCOVERY: ANALYSIS AND ENHANCEMENTS

2.1 Motivation

A key component of ad-hoc network formation is the device discovery process in Fourth Generation (4G) Long Term Evolution (LTE), wherein a user equipment (UE) learns about the presence of other UEs in the vicinity and their capabilities. LTE Sidelink discovery occurs over many "discovery periods" with periodicity *discPeriod* [30] which can take values in binary multiples of 0.32 s (i.e., 0.32, 0.64, 1.28, 2.56, 5.12, and 10.24 s). LTE device-to-device (D2D) is composed of three steps: synchronization, discovery, and data communications. Synchronization occurs over the Physical Sidelink Broadcast Channel (PSBCH) via broadcast signals. Discovery messages use the Physical Sidelink Discovery Channel (PSDCH). Finally, control messages and data transmissions using the Physical Sidelink Control Channel (PSCCH) and the Physical Sidelink Shared Channel (PSSCH), respectively. Control messages and data transmissions repeat periodically with periods 40, 80, 160, or 320 ms [31]. The number of 1 ms sub-frames and pairs of twelve 15 kHz wide sub-carriers, called *resource blocks*, devoted to each transmission type is defined before off-network sidelink transmission begins by the network administrator. We show an example of this in Fig. 2.1.

During synchronization, UEs in a local area come to a consensus over timing to enable subsequent transmission of data and control packets. They accomplish this by using synchronization reference signals transmitted by the UEs themselves [32]. During discovery, UEs announce their capabilities and discover those of other UEs in the vicinity. This information is used during the subsequent communications phase to determine what content to send to the other UEs.

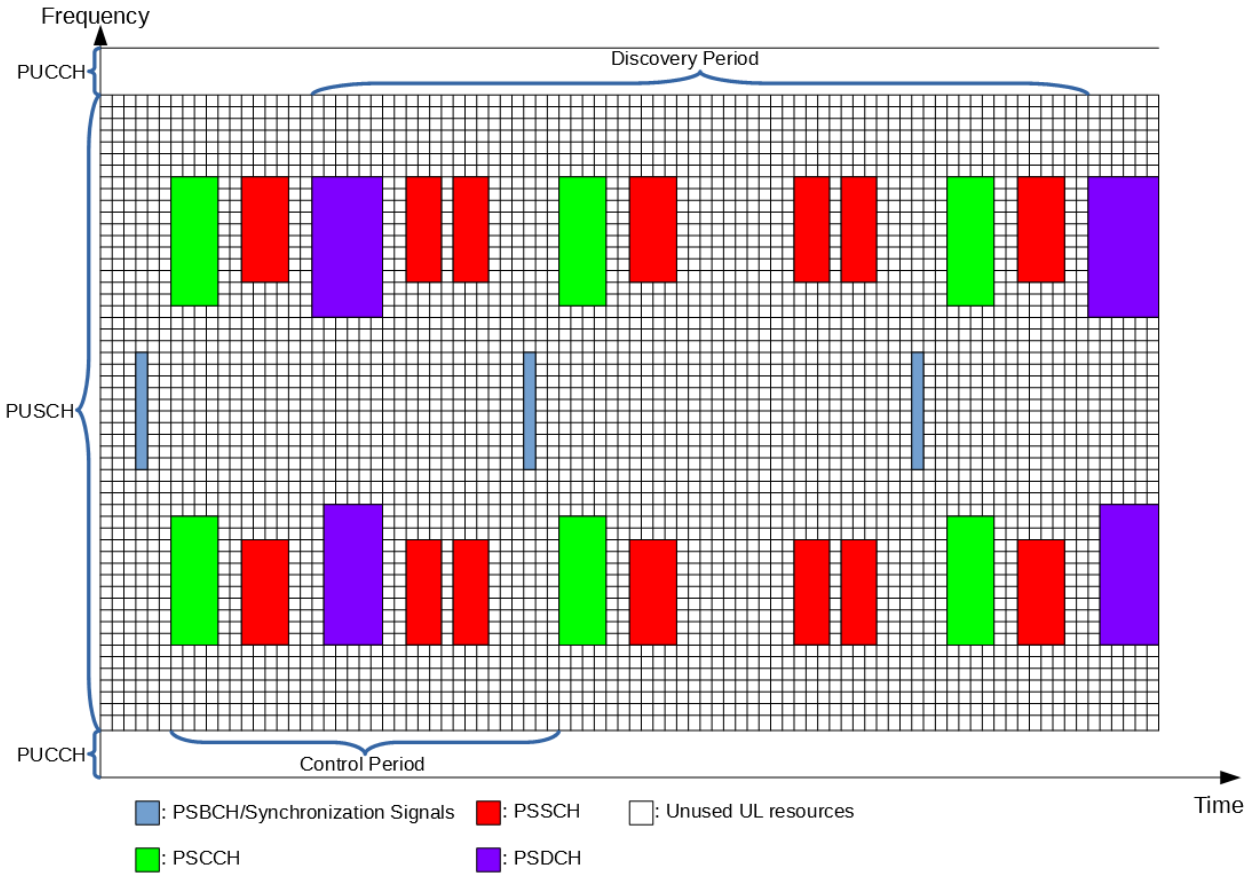


Figure 2.1: The Structure of the Sidelink Physical Channels (© 2021 IEEE)

Because the sidelink channels use the same frequency resources as the uplink, it is reasonable to assume that network providers would disable the sidelink while UEs are connected to the network[1]. Thus for a set of out-of-coverage UEs, they must first perform synchronization and then discovery before data transmission can begin. We assume in this Chapter that UEs have completed the synchronization step.

The key performance metric for the discovery process is discovery completion time, the time required for a reference UE to discover all other nearby UEs. Typically, completing

[1]This is with the possible exception of relaying, i.e., an in-network UE acting as a relay for an out-of-coverage UE, which is outside the scope of this Chapter.

discovery on the order of minutes is reasonable, which corresponds to 187.5 discovery periods during a one-minute duration with the shortest *discPeriod*.

In this chapter, we first analyze the efficiency of the D2D discovery process using mathematical modeling and extensive network simulations. This modeling allows us to gain insights into the discovery process and how best to set θ if all parameters (including N_{UE}) are known to minimize discovery completion time.

This chapter's primary contribution is an improved discovery algorithm to select θ after a training phase - rather than using a pre-selected θ. This training phase allows each UE to learn the optimal θ when all parameters (most notably N_{UE}) are unknown. We will show that our algorithm improves upon the existing 3rd Generation Partnership Project (3GPP) algorithm by being more robust to a lack of foreknowledge of N_{UE} and improves upon other attempts to optimize θ during discovery by providing for a better scaling with N_{UE}. We will show that by using our algorithm to select θ, discovery completion time scales as if N_{UE} was known for highly congested systems.

Major materials from this chapter are presented in the author's previous publication [26], [27][2].

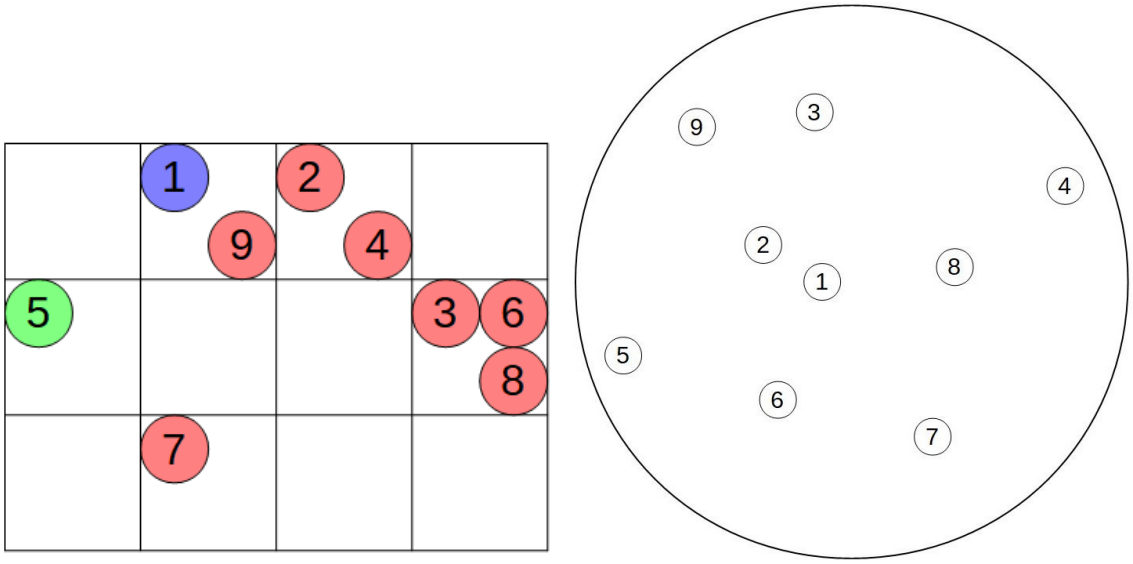
2.2 Modeling Discovery

2.2.1 Modeling One Discovery Period

The D2D discovery process in mode 2 consists of individual discovery periods. During each period, all UEs attempt to transmit on a set of shared resources called the discovery pool. In wired local area networks, it is assumed that collision information is available upon transmission. However, wireless UEs cannot detect collisions. Thus, no mechanism exists to modify subsequent behavior based on collisions. Instead, UEs transmit with probability $\theta \leq 1$ determined before beginning the process, as a preemptive means of collision avoidance, since setting $\theta < 1$ reduces the effective number of contending UEs in any period. In

[2]

principle, θ can be varied period-to-period to optimize the end-to-end discovery process; this is discussed later in Section 2.5. The discovery pool is a set of $N_r = N_f N_t$ physical resource blocks (PRB) represented as a matrix of N_f rows (a pair of twelve 15 kHz sub-carriers, each of which is called a resource block) and N_t columns (1 ms sub-frames). Resource pool sizes are pre-configured by the network administrator. At the beginning of each period, each UE determines whether it will transmit with probability θ. The active UEs then randomly choose one of the available N_r to transmit once per discovery period. Two or more UEs selecting the same PRB leads to interference at every receiver.



(a) Example Discoveries- HD UE and MCM; $N_t = 4, N_f = 3$

(b) Example Physical Distribution of UE in Figs. 2.2a

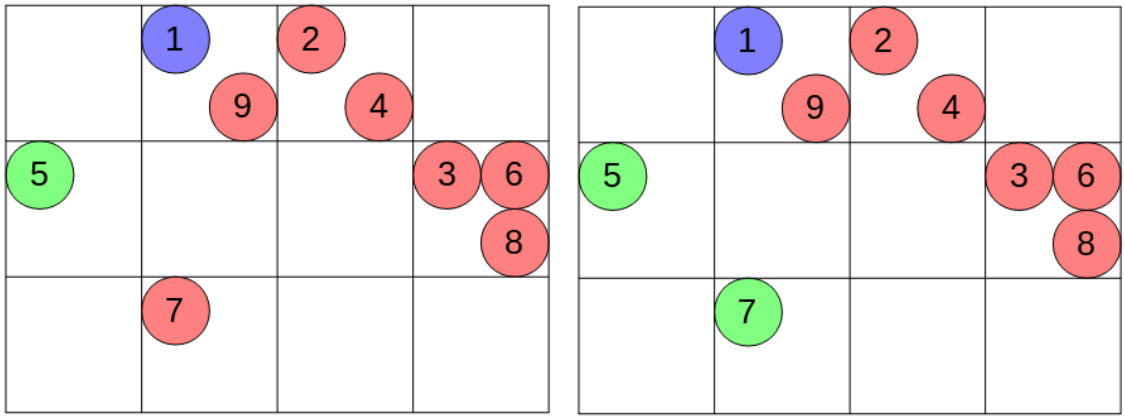
Figure 2.2: Example Discoveries for a Given Physical Distribution (© 2021 IEEE)

All UEs are assumed to be placed randomly (distributed uniformly) on a disk of radius R that represents a single 'collision' domain. If one UE transmits to another in a 'collision' domain, the message is assumed decodable with probability one assuming no interference. If two or more UEs select the same PRB for transmission, we model discovery message decode

probability using the typical legacy assumption, the MAC Collision Model (MCM). The MCM assumes that whenever multiple simultaneous messages collide, none are decodable at any receiving UE. Further, the *Half-Duplex* (HD) nature of UEs [33] implies that all UEs which select the same sub-frame to transmit cannot decode each other's discovery messages.

Fig. 2.2a shows the potential discoveries made based on the MCM with HD UE from the perspective of UE 1 (shown in blue) based on the physical distribution in Fig. 2.2b. The discovered UEs are marked in green, and undiscovered UEs are marked in red. Under the MCM, UE 1 is incapable of discovering any UE in a multiply occupied PRB regardless of the physical distribution of the UEs.

An alternative to HD UEs would be to enhance them by allowing them to be Frequency Division Duplex (FDD) devices. The effects of this are shown in Fig. 2.3, which provides an example of the differences between HD and FDD UEs.



(a) HD assumption (b) FDD assumption

Figure 2.3: Example Discoveries, HD vs. FDD (© 2019 IEEE)

Fig. 2.3a shows the resulting discoveries assuming MCM and HD from the perspective of UE 1, shown in blue. UEs shown in green have their discovery message correctly decoded by UE 1, while those in red have their discovery messages collide. In Fig. 2.3b, we show the

effects of FDD on the same PRB selections; UE 7 would be discovered in this case, while under the HD assumption, it remains undiscovered.

After each discovery period, UEs log *all new UEs*[3] discovered in that period. Each period is identical and memoryless since no parameters are changed based on prior results. Since UEs have no way of confirming that other UEs have discovered them, they never cease to transmit with probability θ, and the process repeats indefinitely.

Probability of k Discoveries in one Discovery Period

The number of discoveries made by a randomly chosen reference UE in a single discovery period is given by the conditional probability $P_D(k|N_p;\, N_{UE},\, N_r,\, \theta)$ that represents the event of k new UE discoveries by the reference UE conditioned on the current *state*: N_p of them being previously discovered, where the resource pool size N_r, N_{UE} potential transmitters, and transmit probability θ are key system parameters. We model the choice of PRB by a transmitting UE - assuming FDD - as a set of two independent multinomial random variables with occupancy matrices $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$ that represent UEs which are undiscovered or previously discovered by the reference UE, respectively.

Computation of $P_D(k|N_p;\, N_{UE},\, N_r,\, \theta)$ is facilitated by splitting into two primary sub-events: (a) the event that the reference UE transmits and (b) the event that the reference UE doesn't transmit, denoted by TX and TX', respectively,

$$\begin{aligned} P_D(k|N_p;\, N_{UE},\, N_r, \theta) = \theta P_D(k|N_p,\, TX;\, N_{UE},\, N_r,\, \theta) \\ + (1-\theta) P_D(k|N_p,\, TX';\, N_{UE},\, N_r,\, \theta). \end{aligned} \tag{2.1}$$

In the event (b) that the reference UE does not transmit, we can ignore the duplex effect. Since in an individual period, all UEs transmit independently with probability θ. It implies that (i) the number of undiscovered UEs that transmit, $N_{u,tx}$, is distributed as $Bin.(N_{UE} - N_p - 1, \theta)$ and (ii) the number of previously discovered UEs that transmit, $N_{p,tx}$, is distributed $Bin.(N_p, \theta)$. We can now reformulate $P_D(k|N_p,\, TX';\, N_{UE},\, N_r,\, \theta)$ in terms of

[3]By implication, messages from UEs that are already discovered in prior periods are discarded.

$P_D(k|N_{u,tx},\, N_{p,tx};\, N_r)$, the probability of the reference UE discovering k other UEs given $N_{u,tx}$ undiscovered UEs and $N_{p,tx}$ previously discovered UEs transmit in a period is

$$P_D(k|N_p, TX';\, N_{UE},\, N_r,\, \theta) = \sum_{n_{u,tx}=0}^{N_{UE}-N_p-1} \sum_{n_{p,tx}=0}^{N_p} P(N_{u,tx} = n_{u,tx}) \cdot P(N_{p,tx} = n_{p,tx}) P_D(k|N_{u,tx} = n_{u,tx},\, N_{p,tx} = n_{p,tx};\, N_r). \tag{2.2}$$

In the event (a) that the reference UE transmits, some of the other UEs may be undiscoverable by the reference UE due to the FDD effect. Specifically, this means that if any other transmitting UE, undiscovered or previously discovered, select the same PRB as the reference UE, they will be undiscoverable by the reference UE. We define the number of previously undiscovered and previously discovered UEs which both transmit and select the same PRB as the reference UE to be $N_{u,d}$, and $N_{p,d}$, respectively. Because UEs select PRB uniformly and independently, $N_{u,d}$ is distributed $Bin(N_{u,tx}, \frac{1}{N_r})$ and $N_{p,d}$ is distributed $Bin(N_{p,tx}, \frac{1}{N_r})$. If we were to formulate this model using the HD assumption, these would be distributed $Bin(N_{u,tx}, \frac{1}{N_t})$ and $Bin(N_{p,tx}, \frac{1}{N_t})$ instead. Similar to $P_D(k|N_p,\, TX';\, N_{UE},\, N_r,\, \theta)$, we now reformulate $P_D(k|N_p,\, TX;\, N_{UE},\, N_r,\, \theta)$ in terms of $P_D(k|N_{u,tx},\, N_{p,tx};\, N_r)$ via

$$P_D(k|N_p, TX;\, N_{UE},\, N_r,\, \theta) = \sum_{n_{u,tx}=0}^{N_{UE}-N_p-1} \sum_{n_{p,tx}=0}^{N_p} P(N_{u,tx} = n_{u,tx}) \times P(N_{p,tx} = n_{p,tx}) \sum_{n_{u,d}=0}^{n_{u,tx}} \sum_{n_{p,d}=0}^{n_{p,tx}} P(N_{u,d} = n_{u,d}) P(N_{p,tx} = n_{p,tx}) \times P_D(k|N_{u,tx} = n_{u,tx} - n_{u,d},\, N_{p,tx} = n_{p,tx} - n_{p,d};\, N_r - 1). \tag{2.3}$$

To determine $P_D(k|N_{u,tx},\, N_{p,tx};\, N_r)$, we next consider how the $N_{u,tx} + N_{p,tx}$ transmitting UEs distribute their messages into the available N_r PRB.

UE PRB Choices: Multinomial Model

The $N_{u,tx}$ undiscovered transmitting UEs independently selecting one of N_r PRB randomly can be considered sampling with replacement $N_{u,tx}$ times from a set of size N_r. Let $\mathbf{X}_{u,tx}$

be the occupancy vector of length N_r where the i^{th} element of $\mathbf{X}_{u,tx}$ corresponds to the number of undiscovered UEs which transmit using the i^{th} PRB. $\mathbf{X}_{u,tx}$ is then a multinomial random variable with a number of trials equal to $N_{u,tx}$, and N_r events each with probability $1/N_r$, i.e., $\mathbf{X}_{u,tx}$ is distributed $Mult.(N_{u,tx}, [1/N_r, ..., 1/N_r])$. Similarly $\mathbf{X}_{p,tx}$ is distributed $Mult.(N_{p,tx}, [1/N_r, ..., 1/N_r])$. We separately account for the occupancy of undiscovered and previously discovered UEs and rewrite $P_D(k|N_{u,tx},\ N_{p,tx};\ N_r)$ by summing over all possible $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$, shown as

$$P_D(k|N_{u,tx},\ N_{p,tx};\ N_r) = \sum_{\mathbf{X}_1} P(\mathbf{X}_{u,tx})P(\mathbf{X}_{p,tx})P(k|\mathbf{X}_{u,tx}, \mathbf{X}_{p,tx}). \tag{2.4}$$

In (2.4) $\mathbf{X}_1$ is the set of all $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$ which can be formed with $N_{u,tx}$ UEs selecting from N_r PRB or $N_{p,tx}$ UEs selecting from N_r, respectively[4] and $P(k|\mathbf{X}_{u,tx}, \mathbf{X}_{p,tx})$ is the probability of k discoveries given a set of occupancy vectors $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$.

For the MAC collision model, the above implies that $P(k|\mathbf{X}_{u,tx}, \mathbf{X}_{p,tx}) = 1$ if and only if there are exactly k entries in $\mathbf{X}_{u,tx}$ which have $X_{u,tx}(i) = 1$ and $X_{p,tx}(i) = 0$ (i.e. k PRBs are occupied only by one undiscovered UE). If we used a physical (PHY) collision model, where messages were not automatically discarded if the PRB contained more than one message, we would need to consider all PRB occupied by at least one undiscovered UE as potential discoveries.

Using (2.4) in (2.1), we arrive at a final computational algorithm after some simplification as

[4]In actual computations $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$ are unwieldy. The size of $\mathbf{X}_1$ grows rapidly with $O(N_{UE}^{N_r})$. Hence, we use a simpler but equivalent variable formed by removing all of the entries of $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$ where both are equal to 0 that allows for much quicker computation.

$$P_D(k|N_p;\, N_{UE},\, N_r,\, \theta) = \sum_{n_{u,tx}=0}^{N_{UE}-N_p-1} \sum_{n_{p,tx}=0}^{N_p} P(N_{u,tx} = n_{u,tx}) P(N_{p,tx} = n_{p,tx}) \times$$
$$\Big[\theta \sum_{n_{u,d}=0}^{n_{u,tx}} \sum_{n_{p,d}=0}^{n_{p,tx}} P(N_{u,d} = n_{u,d}) P(N_{p,d} = n_{p,d}) \sum_{\mathbf{X_2}} P(\mathbf{X}_{u,tx}) P(\mathbf{X}_{p,tx}) P(k|\mathbf{X}_{u,tx}, \mathbf{X}_{p,tx}) \tag{2.5}$$
$$+ (1-\theta) \sum_{\mathbf{X_1}} P(\mathbf{X}_{u,tx}) P(\mathbf{X}_{p,tx}) P(k|\mathbf{X}_{u,tx}, \mathbf{X}_{p,tx}) \Big],$$

where $\mathbf{X}_2$ is the set of all $\mathbf{X}_{u,tx}$ and $\mathbf{X}_{p,tx}$ which can be formed with $n_{u,tx} - n_{u,d}$ UEs selecting from $N_r - 1$ PRB or $n_{p,tx} - n_{p,d}$ UEs selecting from $N_r - 1$ PRBs, respectively.

2.2.2 Modeling the Entire Discovery Process

After each discovery period, UEs decode N_m discovery messages and log the K messages which originate from previously undiscovered UEs[5], then updates its cumulative list of discovered UEs.

The probability of discovering K UEs, $P_D(K)$, and the probability of decoding N_m discovery messages, $P_{decode}(N_m)$, are statistically identical if and only if the number of previously discovered UEs, N_p, is zero and all other system parameters (N_t, N_f, θ, etc.) remain constant,

$$P_{decode}(N_m|N_{UE}, N_r, N_t, \theta) = P_D(K|N_p = 0; N_{UE}, N_r, N_t, \theta). \tag{2.6}$$

If the discovery process conforms to the standard, each period is memoryless and identical, and no parameters are changed. Furthermore, because there is no feedback, UEs never cease to transmit, and the process repeats indefinitely.

We focus on the discovery process for a single (randomly chosen) UE; the time (in numbers of periods) to discover all other UEs, the discovery completion time, is N_{DCT}. We model the discovery process as a modified coupon collectors problem. The Markov chain describing

[5]By implication, UEs discard messages received from UEs that were previously discovered.

this process is shown in Fig. 2.4. In each period, the reference UE can discover K, $0 \leq K \leq min(N_r, N_{UE} - 1)$ new UEs.

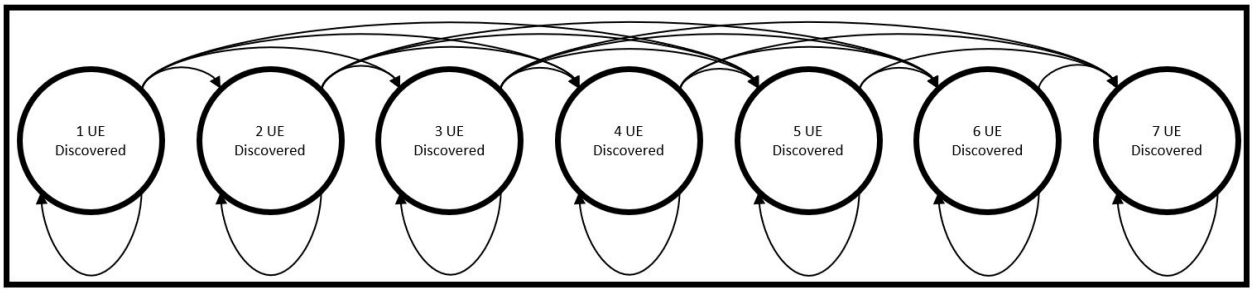


Figure 2.4: Markov Chain Describing the Discovery Process, $N_{UE} = 8$, $N_r = 4$ (© 2021 IEEE)

Since the 3GPP standard requires a constant θ across all discovery periods, we call this the Fixed Transmission Algorithm to establish a baseline.

We define the state transition matrix as $\mathbf{\Phi}$ with state transition probabilities $\Phi_{i,j} = P_D(i + j \,|\, i;\, N_{UE},\, N_r,\, N_t,\, \theta)$ from Eq. (2.5). Note that a) UEs cannot discover more than $N_{UE} - N_p$ UEs (no more to discover) or b) more than N_r (only one UE can be discovered per PRB). Because of this $\Phi_{i,j} = 0$ for all $j - i > min(N_{UE} - N_p, N_r)$. Additionally, once a UE has discovered other UEs, it can not undiscover them, so $\Phi_{i,j} = 0$ for $j < i$. An example $\mathbf{\Phi}$ is shown when $N_{UE} = 3$ in Eq. (2.7),

$$\mathbf{\Phi} = \begin{bmatrix} P_D(0 \,|\, 0;\, N_{UE},\, N_r,\, N_t,\, \theta) & P_D(1 \,|\, 0;\, N_{UE},\, N_r,\, N_t,\, \theta) & P_D(2 \,|\, 0;\, N_{UE},\, N_r,\, N_t,\, \theta) \\ 0 & P_D(0 \,|\, 1;\, N_{UE},\, N_r,\, N_t,\, \theta) & P_D(1 \,|\, 1;\, N_{UE},\, N_r,\, N_t,\, \theta) \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.7)$$

The discovery completion process ends with the Markov model's first entry into the absorbing state. Hence, determining the mean discovery completion time is equivalent to determining the mean time to absorption. Assuming we begin at state $N_p = 0$, the Cumulative Distribution Function (CDF) of N_{DCT} is given by

$$P(N_{DCT} < n) = (\mathbf{\Phi}^n)_{0,N_{UE}-1}. \tag{2.8}$$

Because $\mathbf{\Phi}$ contains a single absorbing state, we can partition it into the following sub-matrices [34],

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{Q} & \mathbf{r} \\ \mathbf{0}_{N_{UE}-1} & 1 \end{bmatrix}, \tag{2.9}$$

where $\mathbf{Q}$ is an $(N_{UE}-1) \times (N_{UE}-1)$ matrix representing the transient state transitions, $\mathbf{r}$ is a length $(N_{UE}-1)$ column vector representing the probability of entering the absorbing state, and $\mathbf{0}_{N_{UE}-1}$ is a length $(N_{UE}-1)$ row vector of zeroes. The fundamental matrix is then

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1}, \tag{2.10}$$

where $\mathbf{I}$ is the identity matrix. The individual elements of $\mathbf{N}$, $n_{i,j}$, represent the mean number of times state j is visited, given the chain began at state i. So, the mean number of periods until discovery completion time is

$$E[N_{DCT}] = \sum_{j=0}^{N_{UE}-1} n_{0,j}. \tag{2.11}$$

2.2.3 Model Insights

Given that the resource pool configuration N_r, and N_t is pre-determined, and N_{UE} is outside the network designers control, the only controllable parameter during discovery is θ; hence, the design of algorithms for learning the optimal θ during discovery is of key interest. Using Eq. (2.11), we can a priori calculate the optimal θ which minimizes $E[N_{DCT}]$, θ^*, provided N_{UE} is known. We will show that in addition to minimizing $E[N_{DCT}]$, θ^* also maximizes the per-period $E[K]$ independent of N_p, implying that constant θ^* achieves our design goal.

We then seek an *iterative* algorithm that learns θ^* using information gained during each discovery period for the scenarios where N_{UE} is unknown.

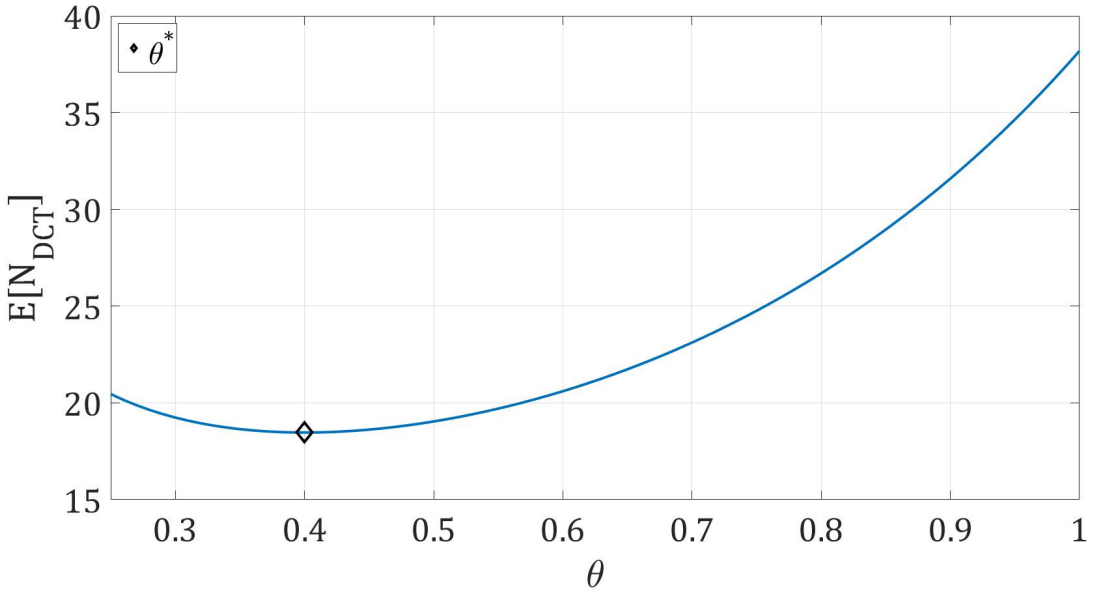


Figure 2.5: $E[N_{DCT}]$ as a Function of θ, MCM, Calculated, $N_{UE} = 10$, $N_r = 4$, $N_t = 4$ (© 2021 IEEE)

Fig. 2.5 shows the behavior of $E[N_{DCT}]$ as a function of θ for $N_{UE} = 10$, $N_r = 4$, and $N_t = 4$ with the MCM calculated using Eq. (2.11). For this parameter combination, the minimum $E[N_{DCT}]$ occurs at $\theta^* = 0.4$, a unique value in $0.25 \leq \theta \leq 1$ regardless of the parameters N_{UE}, N_r, or N_t. The existence of a minimizing θ^* is understood as follows: for $\theta > \theta^*$ the system will be overloaded, leading to more collisions, while for $\theta < \theta^*$ the system will be underloaded, leading to unused PRB. This is confirmed in Fig. 2.6 that shows individual period $E[K]$ as a function of N_p for various θ. These are found by iteratively computing Eq. (2.5) for different θ and computing expectations of the resulting Probability Mass Function (PMF). For this parameter combination, the max occurs at $\theta = 0.4$ for all values of N_p, i.e., the optimal θ does not change with N_p.

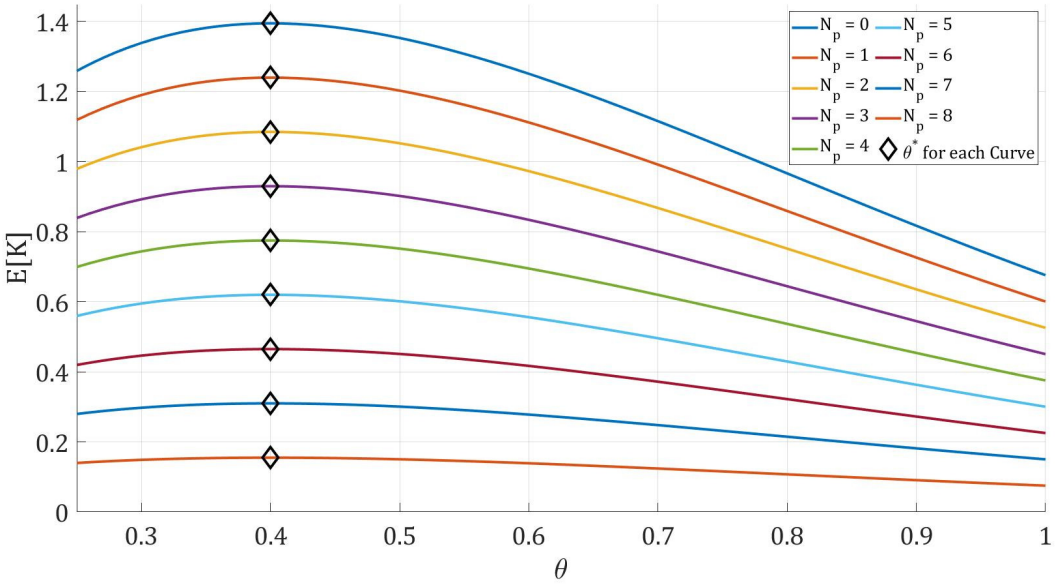


Figure 2.6: $E[K]$ as a Function of θ, MCM, Calculated, $N_{UE} = 10$, $N_r = 4$, $N_t = 4$ (© 2021 IEEE)

That the θ^* maximizing $E[K]$ does not change with N_p can be understood by breaking the discovery process into two parts; decoding discovery messages and identifying *new* (previously undiscovered) UEs from those discovery messages. Because the transmission of discovery messages is both decentralized and memoryless, the number of messages decoded at a node is random. So, the only way to increase the number of discoveries is to increase the number of discovery messages decoded. The number of discovery messages decoded in a single discovery period, N_m, is independent of the number of previously discovered UEs. This means that choosing the value of θ myopically (for a given N_r, N_t, and N_{UE}) that maximizes $E[K]$ in a period will also lead to minimizing $E[N_{DCT}]$.

Selecting θ^* relies on knowledge of N_{UE}, N_r, and N_t. In Section 2.5.1, we explore the solution when N_{UE} is known to provide a lower bound. While in Section 2.5.2, we propose a solution when N_{UE} is unknown and show that our algorithm converges to the same solution as when N_{UE} is known.

2.3 Simulation Methods to Cross Validate Mathematical Modeling and ns-3 Implementation of D2D Discovery

2.3.1 Discovery Process with Constant θ: Simulation Overview

Our mathematical model provides practical insights for *small* systems; for any other reasonable scenario, we must rely on simulations of the D2D process. While Eq. (2.5) applies in principle for systems of any scale, the number of summands scale as $O(N_{UE}!)$. This scaling makes computation for large values of N_{UE} practically impossible.

Simulations of LTE D2D sidelink use its implementation in the network simulator ns-3 [35][6]. ns-3 is increasingly adopted by industry and federal labs for wireless network performance evaluation, primarily due to the leading-edge implementations of new network protocol stacks unavailable in other network simulators. The LTE Sidelink extension is a work-in-progress supported by the US National Institute of Standards and Technology (NIST) [36]. In conjunction with the Centre Tecnològic Telecomunicacions Catalunya (CTTC) Barcelona and U. Washington, the current effort builds on the original LTE-LENA Evolved Packet Core (EPC) and Radio Access Network (RAN) stack features [37] in ns-3 to integrate sidelink capabilities. The NIST code provides a full implementation of the discovery process, placing UEs, determining transmitters and their PRB, resolving collisions, and all other aspects of LTE transmission. We modified the code to allow for parameter changes via scripting and altered the end condition from a set timer to completion after all UEs have discovered all other UEs[7].

The ns-3 simulation output data is extracted and processed using Matlab. The ns-3 simulation traces reports all event timestamps, which UEs are transmitting, the corresponding PRB used, and which UEs have decoded the discovery message. For all resource pool sizes used for this Chapter, the resource pool ends well before the beginning of the next discovery period. Because of this, there are large gaps between timestamps, which indicate breaks be-

[6] ns-3 is the leading open-source network simulator funded by the US National Science Foundation (NSF) for academic researchers.

[7] All code required to generate and analyze the data found can be found at https://git.io/JvTDZ.

tween periods. Using this trace information and simulation meta-data, we count the number of periods taken for discovery completion time for each UE in each trial.

Our ns-3 simulation scenario is comprised of UEs placed over a circular disc with no hidden nodes, i.e., any transmission by a UE can be heard by all other UEs. Further,

- UEs are distributed randomly (uniformly),
- All UE transmissions are assumed to be perfectly synchronized with slot boundaries,
- UEs transmit using LTE band 14.

We set each UEs transmit power to $P_t = -10$ dBm, and we assumed the noise figure on receive to be $F = 0$ dB. This setting results in the coverage range for UE transmission to have a radius $R = 5000\ m$, so any two UEs within a separation distance R will be able to decode each other's discovery message with probability 1 in the absence of interference.

During simulation, we varied N_{UE} from $\lfloor N_r/4 \rfloor$ to $N_{UE} = 3N_r$ to evaluate the system under low and high loads. We set the number of trials N for averaging to $N = \lceil 10000/N_{UE} \rceil$ as a function of N_{UE}. For each (N_{UE}, N_r, N_t) triplet (Fig. 2.7a), UEs are placed randomly on a disk of radius 5000 m, (Fig. 2.7b).

The simulator conducts a discovery period as follows: each UE generates a standard U[0,1] random variable independently and compares it to θ. If the randomly generated number is less than θ, the UE is considered 'active,' meaning it seeks to transmit (Fig. 2.7c). Each active UE selects one of the N_r PRBs uniformly and then transmits with $P_t = -10\ dBm$ (Fig. 2.7d). All UE receivers scan the PRB pool to decode the discovery message using the MCM (Fig. 2.7e). After each period, all UEs compile a list of successfully received discovery messages and their associated PRB. The simulation ends after all UEs have successfully received a discovery message from all other UEs at least once (Fig. 2.7f). Currently, UEs in NS-3 are HD devices. To emulate the behavior of FDD UEs, we map all resource pools of size (N_f, N_t) to pools of size $(1, N_f \times N_t)$ with contiguous sub-frames. By doing this, we

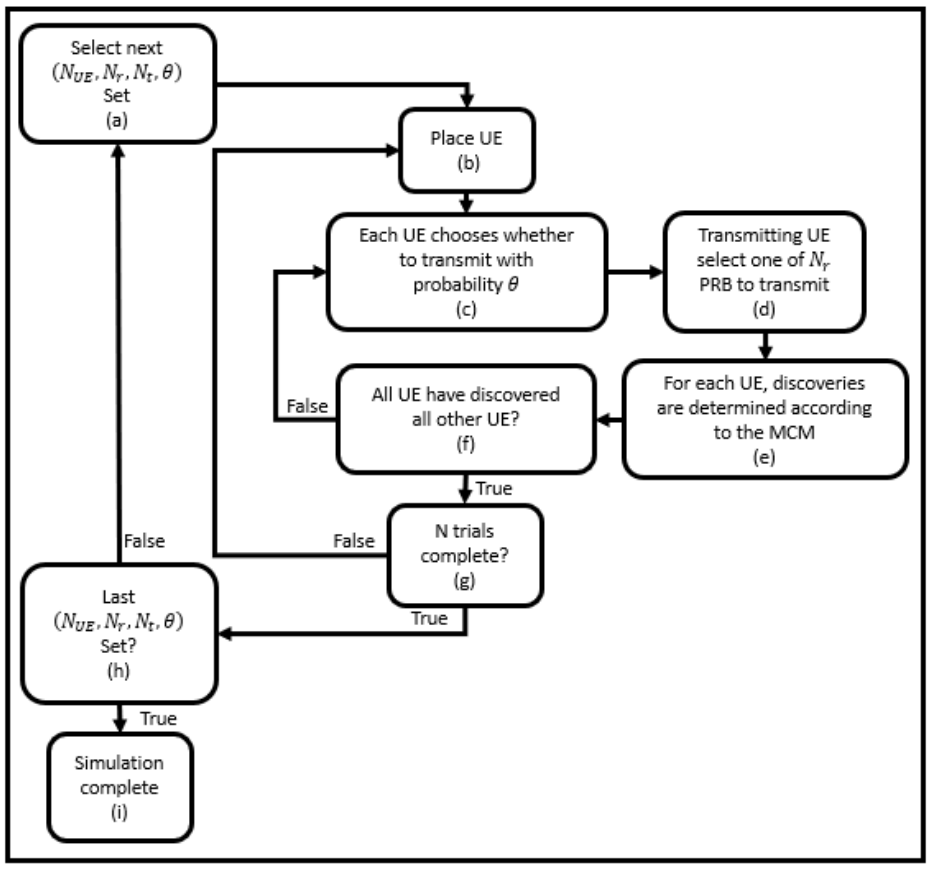


Figure 2.7: Simulation Flowchart: Constant θ Simulations (© 2021 IEEE)

eliminate the possibility of two UEs selecting the same subframe but not the same resource block.

We reposition the UEs and repeat steps (Fig. 2.7c - 2.7f). After N trials are completed, (Fig. 2.7g), we move to the next values of (N_{UE}, N_r, N_t) triplet (Fig. 2.7h) to obtain the full suite of results (Fig. 2.7i).

Tab. 2.1 contains the time taken to complete an individual trial for the simulations. Trial completion times were found by running 10000 trials and dividing the total time taken by the number of trials to find the per-trial average duration.

For trial completion time, the resource pool size and structure have some impact. Larger

$(N_{UE}, N_r, N_t, \theta)$	Trial Completion Time (μs)	$(N_{UE}, N_r, N_t, \theta)$	Trial Completion Time (μs)
(10,10,10,1)	12.96	(20,10,10,1)	25.37
(10,20,10,1)	12.48	(30,10,10,1)	58.78
(10,30,10,1)	12.47	(40,10,10,1)	162.04
(10,40,10,1)	12.70	(20,10,10,0.75)	18.98
(10,20,20,1)	10.05	(20,10,10,0.5)	15.78
(10,30,30,1)	9.07	(20,10,10,0.25)	14.50
(10,40,40,1)	8.64		

Table 2.1: Simulation Trial Completion Time, Constant θ Simulations ()

resource pool sizes lower the trial completion time as N_t (the number of sub-frames) increases while increasing the number of resource blocks N_f has a marginal effect. We conclude that increasing the resource pool size and changing the configuration reduces the trial completion time as $O(\frac{1}{N_t})$. The main driver of complexity is the value of N_{UE}. Increasing N_{UE} increases the trial completion time at rate $O(N_{UE}^2)$.

From Tab. 2.1, we can also observe that the trial completion time scales quadratically with θ. This scaling occurs because values of θ below one act to effectively reduce N_{UE} in a discovery period. For the parameters in Tab. 2.1, reducing θ reduces the simulation time because the system is congested and the optimal θ is small. In general, the trial completion time grows with $O((1 + \frac{|\theta-\theta^*|}{\theta^*})^2)$, the distance between the chosen θ and optimal θ^*.

Overall the MCM model has complexity,

$$T_M(N_{UE}, N_t, \theta) \in O\left(\frac{\left(c + \frac{|\theta-\theta^*|}{\theta^*}\right)^2 N_{UE}^2}{N_t}\right), \tag{2.12}$$

where c is some constant.

From the output logs, we can extract period-by-period discovery data when a particular UE was discovered, along with data for the whole discovery process, to determine how many periods elapse until a UE has discovered every other UE. Next, we focus on the number of periods, N_{DCT}, until a reference UE has discovered all other UEs, the discovery completion time. In the results section, we examine CDFs of N_{DCT} and the impact of varying N_r and N_{UE}.

2.3.2 Discovery Process with Variable θ: Simulation Overview

To simulate the discovery process while varying θ period-to-period, we recreated the key elements of the ns-3 simulation in Matlab.

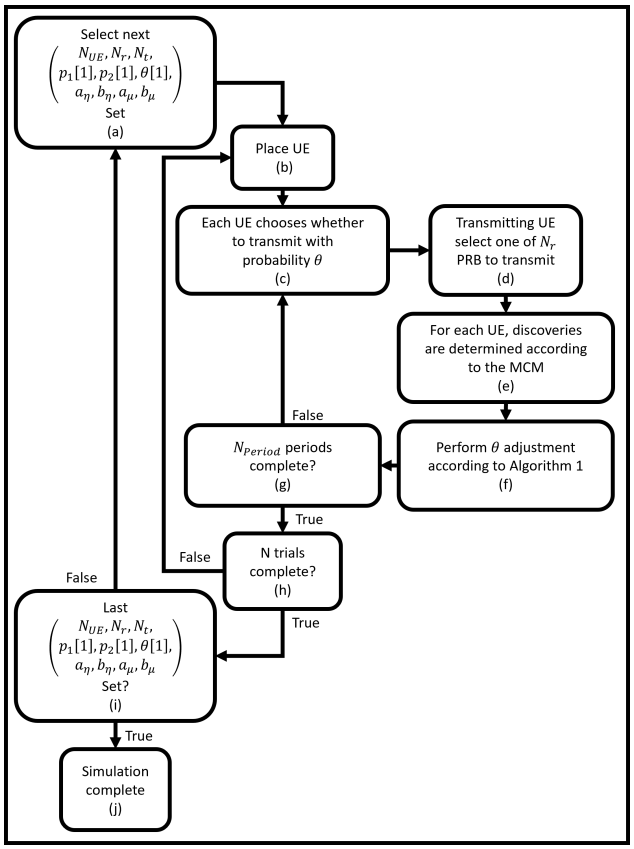


Figure 2.8: Simulation Flowchart: Adaptive θ Simulations (© 2021 IEEE)

The Matlab implementation follows The same steps of the ns-3 implementation of discovery (UE placement, Fig. 2.8b, transmitting with probability θ, Fig. 2.8c, selection of a PRB randomly, Fig. 2.8d, resolving discoveries according to the MCM, Fig. 2.8e) along with making the same structural assumptions as the ns-3 model (UEs are synchronized, UEs are half-duplex, etc.) with three key differences.

Firstly the Matlab simulation iterates over many more variables (Fig. 2.8a) than the ns-3 simulation (Fig. 2.7a). The added variables (the stochastic gradient descent (SGD) initialization parameters $p_1[1],\ p_2[1],\ \theta[1]$, and the algorithms tuning parameters $a_\eta,\ b_\eta,\ a_\mu,\ b_\mu$) are all related to Alg. 1 and naturally would not appear in an implementation without the adaptive algorithm. The large number of variables can result in simulations that take an unacceptably long time to complete. So, in practice, we limited ourselves to exploring subsets of those parameters for individual simulation runs.

Secondly, we changed the terminating condition on individual simulation trials. In ns-3, the simulation terminates when all UEs have discovered all other UEs (Fig. 2.7f). While in Matlab, the simulation terminates when N_{Period} discovery periods have passed (Fig. 2.9g). We made this change because, for specific combinations of simulation variables, discovery can conclude before the adaptive algorithm's training concludes. As our goal in the Matlab simulation was to evaluate the adaptive algorithm's performance, especially its convergence properties, concluding the simulation before training was complete was not acceptable. One could construct an ending condition that relied upon ending when both UEs' training was complete, and all UEs had discovered all other UEs. However, we chose the condition in Fig. 2.8 to facilitate a more straightforward analysis.

Finally and most importantly, the Matlab simulation includes a block that implements Alg. 1, Fig. 2.8f. The contents of that block are summarized in Alg. 1 and in Fig. 2.9

Tab. 2.2 contains the time taken to complete an individual trial for the variable θ simulations. Similar to the constant θ simulations, the trial completion times were found by

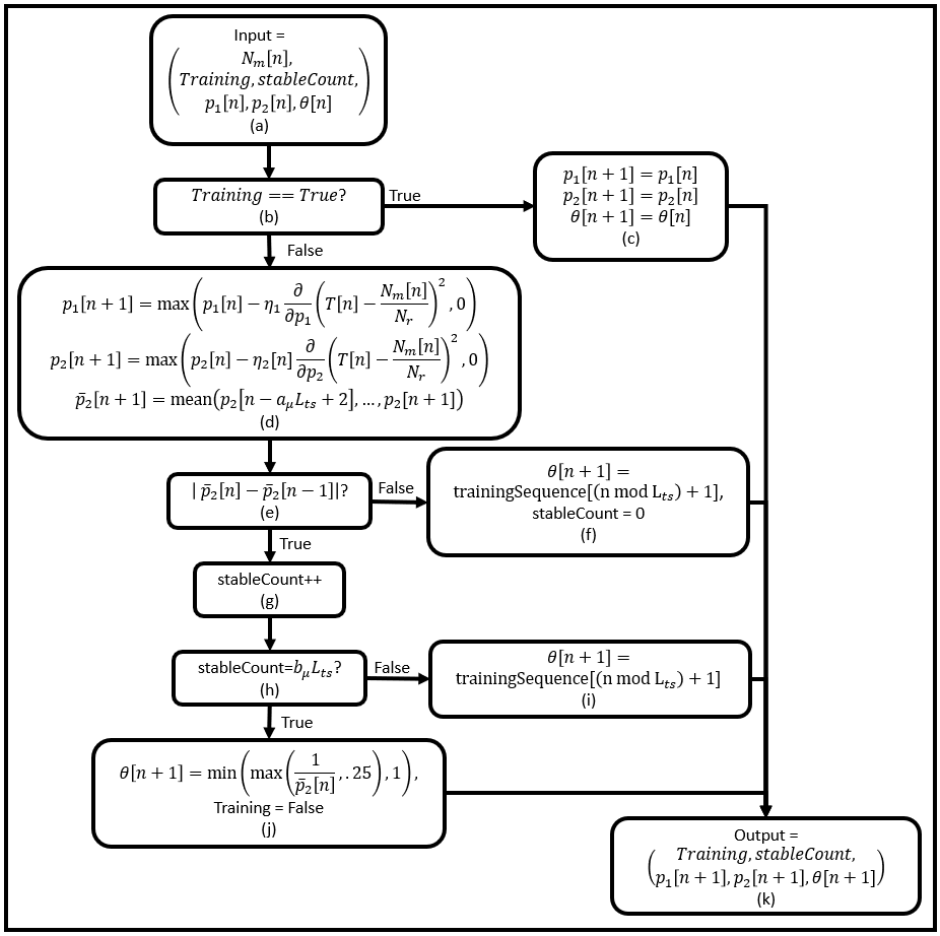


Figure 2.9: Simulation Flowchart: Algorithm 1 Matlab Implementation (© 2021 IEEE)

running 10000 trials with 1000 periods[8] and dividing the total time taken by the number of trials to find the per-trial average duration.

In the case of the variable θ simulations, the trial completion time is much less influenced by any of the parameters under test. That is because the primary driver becomes the number of periods the simulation is run rather than the simulation parameters. Overall the trial completion time grows linearly with both N_r and N_{UE}.

Any real-world implementation of this algorithm would also include a condition to re-enter

[8]This number was chosen to guarantee convergence of the adaptive algorithm.

$(N_{UE}, N_r, N_t, \theta)$	Trial Completion Time (μs)	$(N_{UE}, N_r, N_t, \theta)$	Trial Completion Time (μs)
(10,10,10)	1.98	(10,30,30)	3.12
(10,20,10)	2.86	(10,40,40)	3.42
(10,30,10)	3.31	(20,10,10)	2.37
(10,40,10)	3.62	(30,10,10)	2.79
(10,20,20)	2.75	(40,10,10)	3.40

Table 2.2: Simulation Trial Completion Time, variable θ Simulations (© 2021 IEEE)

the training phase, likely through conditions on the performance of discovery[9]. However, as this Chapter's focus was the performance of the algorithm in terms of convergence and discovery completion time in a static environment, such a condition is outside of the scope of this Chapter.

2.3.3 Validating The Mathematical Model

Before evaluating our adaptive algorithm's performance, we first sought to validate the performance of our Matlab simulation environment as shown in Fig. 2.10. To accomplish this, we compared the results for the discovery process with constant θ obtained from Matlab simulation with those obtained from both ns-3 simulations and our analytical modeling in Section 2.2.

[9]For instance, if N_m/N_r fell $x\%$ below the same value when training completed for y of discovery periods then training should be restarted.

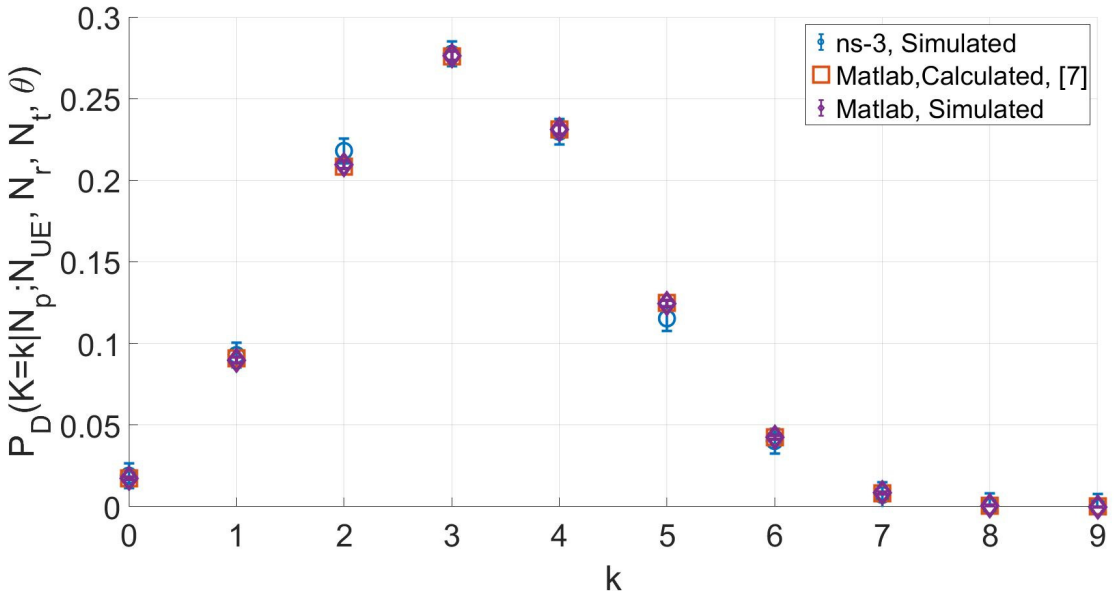


Figure 2.10: $P_D(K|N_p = 0; N_u = 15, N_r = 10, N_t = 10, \theta = 1)$ (© 2021 IEEE)

2.4 Comparison of HD to FDD

Fig. 2.11 compares the expected value of N_{DCT} for a resource pool size of ($N_t = 6$, $N_f = 4$) as we vary N_{UE}[10]. The results are plotted in terms of the number of discovery periods that can be converted to time by assuming that each period corresponds to $0.32s$. As expected, $E[N_{DCT}]$ for FDD begins to noticeably differ from HD for large N_{UE} relative to $N_r = 24$.

The difference between the two assumptions (HD vs. FDD) on $E[N_{DCT}]$ is initially negligible; at $N_{UE} = 6$, the difference is only .6 periods. When $N_{UE} > 37$, however, the difference grows to more than 3 periods (1 sec.) on average. When $N_{UE} = 96$, the difference of averages grows to 41.4 periods.

The differences between HD and FDD are starker when $N_t < N_f$. In Fig. 2.12 we show simulation results for the PMF of N_{DCT} when $N_t = 3$, $N_f = 6$, and $N_{UE} = 40$.

[10]When comparing HD to FDD, note that the resource pool configuration for fixed N_r is irrelevant for FDD but makes a difference for HD. For instance, ($N_t = 6$, $N_f = 4$) and ($N_t = 4$, $N_f = 6$) are equivalent for FDD, but not so for HD.

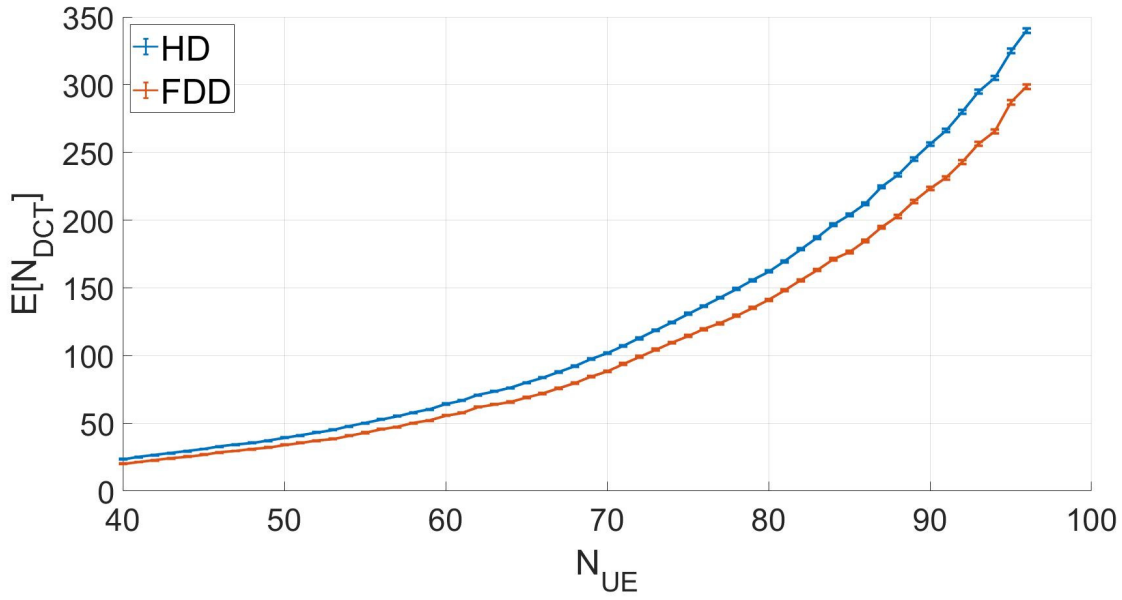


Figure 2.11: Mean Discovery Completion Time $E[N_{DCT}]$, HD vs. FDD, $N_t = 6$, $N_f = 4$, from Simulation (© 2019 IEEE)

In Fig. 2.12, both PMFs are non-symmetric, having longer tails, with the FDD case having significant gains over the HD. In Fig. 2.13, the evolution of the CCDF vs. discovery periods ($P\{N_{DCT}\} > n_{DCT}$) as we increase N_{UE} while holding the resource pool configuration fixed, is shown. The corresponding PMFs are not symmetric, tending towards completing discovery in shorter numbers of periods but with long tails. The tail behavior can be seen in 2.13 to be quite different between the two assumptions.

2.4.1 Trends in FDD

We next examined the trends for N_{DCT} as a function of both N_{UE} and N_r for $\theta = 1$. Fig. 2.14 shows both $E[N_{DCT}]$ and the 95^{th} percentile of N_{DCT} as a function of N_r for $N_{UE} = 10$, 50, and 100. We see that at roughly $N_{UE}/N_r = 1.5$, both metrics change at a much faster rate than for $N_{UE}/N_r < 1.5$. Fig. 2.14 also demonstrates the consequences of selecting a resource pool size that is inappropriate for the number of UEs expected. For example, if the

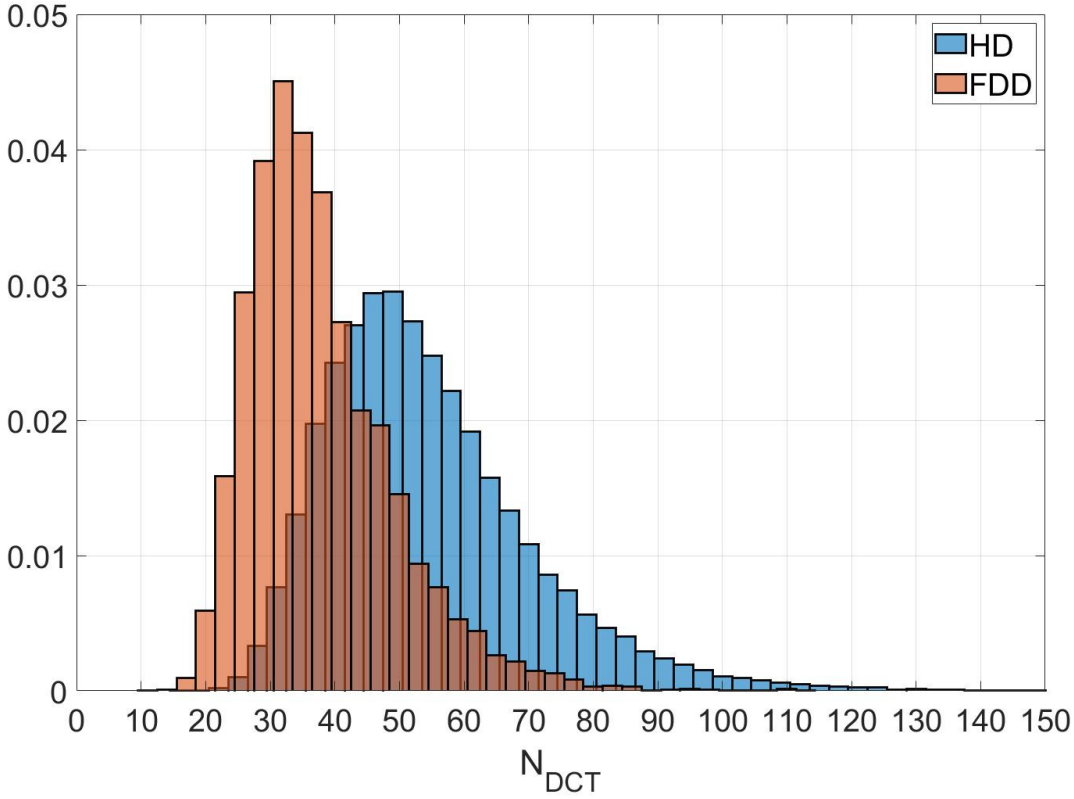


Figure 2.12: Comparison of the PMF of N_{DCT}, HD vs. FDD, from Simulation (© 2019 IEEE)

designer selects a resource pool appropriate for $N_{UE} = 10$, like $N_r = 20$, the system would take on average about 5 discovery periods to complete if $N_{UE} = 20$ or 50 discovery periods if $N_{UE} = 50$. This translates to (using the shortest discovery period allowed in the standard of $0.32\,s$) 1.6 or 16 seconds lost, respectively.

However, from Fig. 2.13, the distribution of N_{DCT} is such that the 95^{th} percentile of the discovery completion time is significantly greater than the mean. Accordingly, in Table 2.3, we have collected a set of minimum N_r to meet performance criterion based both on $E[N_{DCT}]$ and 95^{th} percentile.

For completing discovery in under 3 seconds on average, we see that selecting N_r to be

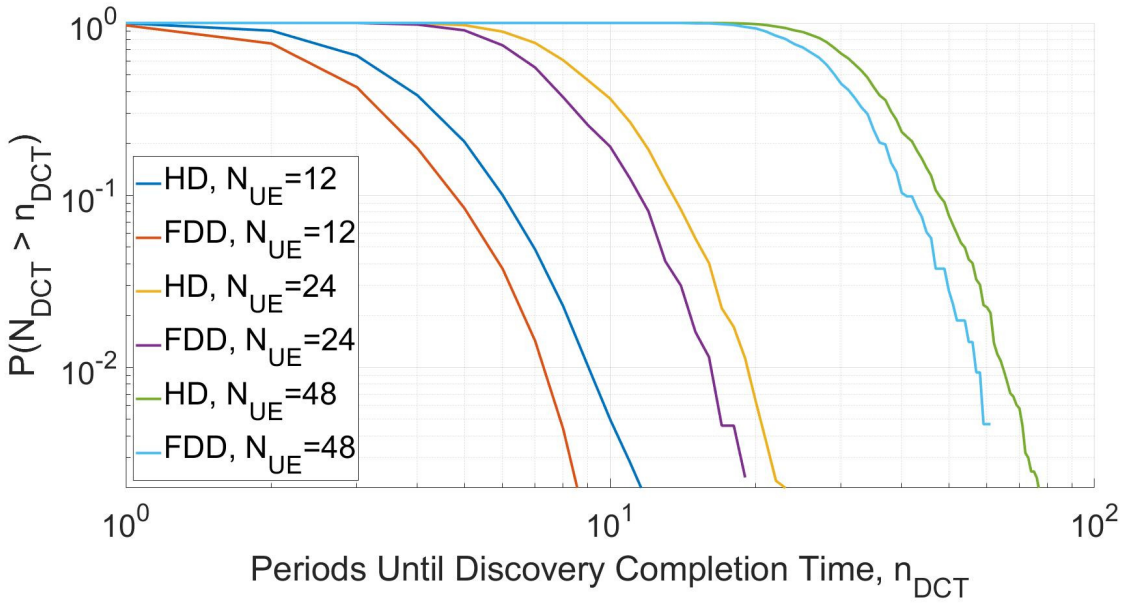


Figure 2.13: CCDF of N_{DCT}, HD vs. FDD, $N_t = 6$, $N_f = 4$, From Simulation (© 2019 IEEE)

N_{UE}	<3 s on average	<10 s on average	<3 s 95% of the time	<10 s 95 % of the time
10	8	5	13	6
20	18	10	31	12
30	29	15	49	18
40	42	20	72	25
50	55	26	89	32
60	68	31	116	39
70	81	37	132	45
80	94	43	160	53
90	110	48	185	59
100	112	53	210	67

Table 2.3: Recommended Minimum N_r Given an Expected N_{UE} for Several Design Criteria for FDD (© 2019 IEEE)

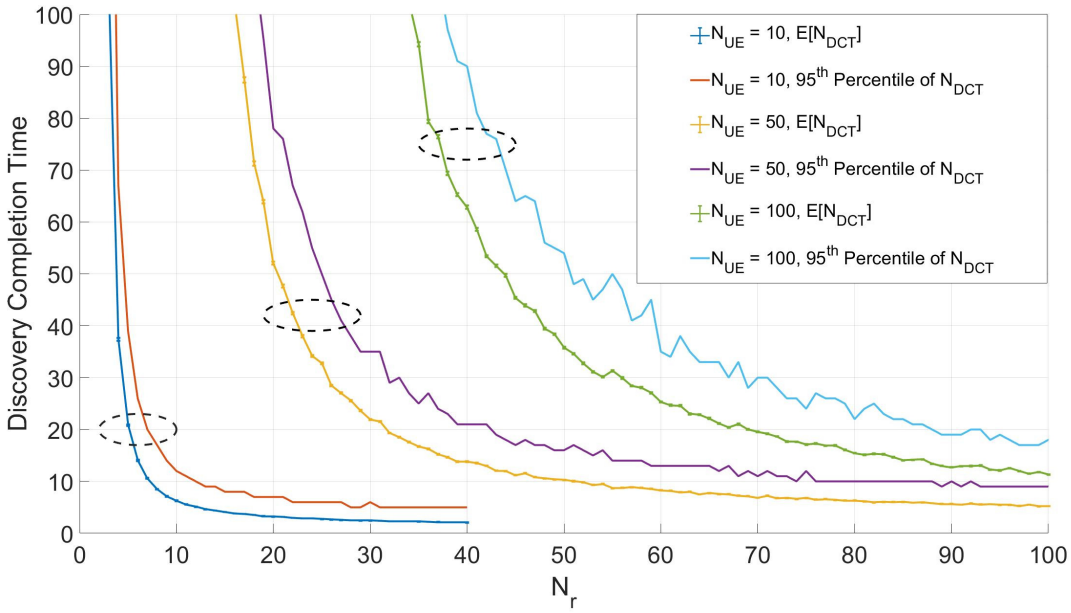


Figure 2.14: $E[N_{DCT}]$ for Various N_{UE} as a function of N_r, FDD Only, From Simulation (© 2019 IEEE)

slightly greater than N_{UE} suffices. But when trying to ensure that discovery takes less than 3 seconds 95% of the time, N_r must be set to slightly more than twice N_{UE}. For a more lenient performance metric: < 10 seconds on average - N_r must be set to roughly $N_{UE}/2$. To achieve discovery in under 10 seconds 95% of the time, we recommend setting N_r to $2N_{UE}/3$. These recommendations serve as a lower bound on the required resource pool sizes using HD UEs.

For the remainder of this Chapter, UEs will be assumed to be HD.

2.5 Optimizing Discovery

2.5.1 Optimization with Perfect Information

If the value of N_{UE} is known, any UE could determine θ^* via computing argmax of Eq. (2.5) or argmin Eq. (2.11) over $0.25 \leq \theta \leq 1$. Table 2.4 shows the values of θ^* for various

combination of N_{UE}, N_r, and N_t. Determining an exact relationship between θ^* and (N_{UE}, N_r, N_t) is not tractable, however we can see from Table 2.4 that when $N_r > N_{UE}$, $\theta^* \rightarrow 1$ except for low values of N_t and N_{UE} must be much greater than N_r (e.g. $N_{UE} = 20$, $N_r = 4$) before $\theta^* = 0.25$. This approach is not implementable in practice and only serves to show a lower bound on the discovery completion time.

		N_{UE}					N_{UE}		
N_r	N_t	5	10	20	N_r	N_t	5	10	20
4	2	0.62	0.36	0.25	16	2	0.9	0.75	0.54
4	4	0.8	0.4	0.25	16	4	1	1	0.68
8	2	0.8	0.56	0.34	16	8	1	1	0.76
8	4	1	0.71	0.38	16	16	1	1	0.8
8	8	1	0.8	0.4	20	2	0.90	0.8	0.61
12	2	0.87	0.68	0.45	20	4	1	1	0.8
12	3	1	0.85	0.51	20	5	1	1	0.85
12	4	1	0.95	0.54	20	10	1	1	0.95
12	6	1	1	0.57	20	20	1	1	1
12	12	1	1	0.6					

Table 2.4: θ^* for various combinations of N_{UE}, N_r, and N_t ()

2.5.2 Optimization with Hidden Features

In the case of unknown N_{UE}, it must be a) either estimated and the value of θ^* computed based on that estimate or b) the value of θ^* directly estimated from simulated data. We have chosen the strategy of directly estimating θ^* through the use of an SGD algorithm.

Setup: During discovery period n, a UE transmits with probability $\theta[n]$ and decodes $N_m[n]$ discovery messages. These data, when normalized by N_r, are used to drive an SGD

algorithm with parameters $p_1[n]$, $p_2[n]$, and α, which approximates a normalized version of the curve shown in Fig. 2.6 by the use of an initial training phase. Because the exact function which describes the curve in Fig. 2.6 for $N_p = 0$ is not tractable, we propose to approximate it via

$$T[n] = p_1[n]\theta[n]e^{-p_2[n](\theta[n]-\alpha)}. \tag{2.13}$$

Note that the parameter $p_1[n]$ acts like a scale parameter, reflecting how often discovery messages are decoded, while the $p_2[n]$ acts like a location parameter, as the max of $T[n]$ occurs at $1/p_2[n]$.

The structure of Eq. (2.13) is apropos for several reasons. First, $E[k]$ increases more quickly before the peak than its rate of decrease after, as can be seen in Fig. 2.6. Note that $E[k]$ is roughly equal at $\theta = .25$ and $\theta = .6$ for all values of N_p. This behavior is present regardless of the specific values of N_{UE}, N_r, or N_t. Second, using Matlab's curve fitting tool, a good fit was computed for $E[K|N_p = 0]/N_r$ vs. θ for many combinations of parameters (N_{UE}, N_r, N_t)[11]. Third, it has a unique global maximum at $\theta = \frac{1}{p_2}$, which will prove convenient during training. And finally, the function has partial derivatives with respect to p_1, p_2, which changes sign over the domain of θ.

While the first and second conditions are relatively easy to meet (and are met by other functions), the third is a key property for successfully finding the value of θ^*. Because θ is bounded by $0.25 \leq \theta \leq 1$, it follows that $p_1 > 0$, $p_2 > 0$. This is because any reasonable fit of $E[K|N_p = 0]$ as a function of θ will have a global maximum only when the above holds.

To estimate the parameters of $T[n]$, we selected the least squares objective function

$$Q(p_1, p_2)[n] = \left(T[n] - \frac{N_m[n]}{N_r}\right)^2 = \left(p_1[n]\theta[n]e^{-p_2[n](\theta[n]-\alpha)} - \frac{N_m[n]}{N_r}\right)^2. \tag{2.14}$$

With Eq. (2.14), the updates for (p_1, p_2) are given by

[11] Although the function need only approximate $E[k]$ for $N_p = 0$, Eq. (2.13) also approximates $E[k]$ for all other values of N_p.

$$
\begin{aligned}
p_1[n+1] &= p_1[n] - \eta_1 \frac{\partial}{\partial p_1} Q(p_1, p_2)[n] \\
&= p_1[n] - \eta_1 \left(2\theta[n] e^{-p_2[n](\theta[n]-\alpha)} \left(T[n] - \frac{N_m[n]}{N_r} \right) \right),
\end{aligned}
\tag{2.15}
$$

and

$$
\begin{aligned}
p_2[n+1] &= p_2[n] - \eta_2 \frac{\partial}{\partial p_2} Q(p_1, p_2)[n] \\
&= p_2[n] - \eta_2 \left(-2p_1[n]\theta[n](\theta[n]-\alpha) e^{-p_2[n](\theta[n]-\alpha)} \left(T[n] - \frac{N_m[n]}{N_r} \right) \right),
\end{aligned}
\tag{2.16}
$$

where η_1, η_2 are the learning rates for p_1, p_2 respectively. Normalizing N_m by N_r sets a bound on the number of messages received ($N_m/N_r \in [0,1]$), which makes the initialization of p_1 easier.

Discussion: By examining Eqs. (2.15) and (2.16), the significance of the term $(\theta[n]-\alpha)$ term in Eq. (2.13) becomes apparent. When $\theta > \alpha$, the parameters can only move in either the $(\hat{p_1}, -\hat{p_2})$ or the $(-\hat{p_1}, \hat{p_2})$ direction (opposite direction), whereas when $\theta < \alpha$ the parameters can only move in either the $(\hat{p_1}, \hat{p_2})$ or the $(-\hat{p_1}, -\hat{p_2})$ direction (same direction). If Eq. (2.13) lacked the $(\theta[n]-\alpha)$ term, then the updates for p_1, p_2 could only move in the opposite direction because of the bounds on θ, p_1, p_2.

This means that for initial conditions at $(1,1)$, the minimum of the objective function was achieved at $(.5,.5)$; absent the $(\theta[n]-\alpha)$ term, the minimum number of steps to reach the minimum would be 2[12]. The value of α can be chosen within the bounds of θ and affects the subsequent training. Training is performed by setting θ to any sequence of values comprised of point pairs, where the pairs are evenly spaced both above and below α[13] and then repeating until the training phase is complete.

---

[12]In practice, it would take more as $\frac{\partial}{\partial p_2} T[n] = (-p_1[n](\theta[n]-\alpha)) \frac{\partial}{\partial p_1} T[n]$ if one update is non-zero the other tends to be as well, which results in slow movement along the $(-\hat{p_1}, -\hat{p_2})$ and $(\hat{p_1}, \hat{p_2})$ directions.

[13]For example, if $\alpha = 0.5$ a valid training sequence might be $\theta = [0.25, 0.4, 0.6, 0.75$. The sequence contains two pairs, $(0.25, 0.5)$ and $(0.4, 0.6)$, which are respectively 0.25 and 0.1 below and 0.25 and 0.1 above.

After some experimentation, it was determined that 1) the best training sequences were ones that switched above and below α quickly, and 2) the best training sequences spanned the entire domain of θ. As long as these conditions are met, the length of the training sequence was not particularly impactful. Hence, for simulations, we used $\alpha = 0.625$ and the training sequence $\theta = [1, 0.75, 0.5, 0.25]$, since $\alpha = 0.625$ lies at the midpoint of the domain of θ, making training sequences easy to generate.

During training, the learning rate η_2 is adjusted each period to enable quick convergence to the general location of the minimum of the objective function, then to tune in on the correct solution. Overall $\eta_2[n]$ is:

$$\eta_2[n] = \eta_{2,i} \operatorname{erfc}((n - a_\eta)/b_\eta) + \eta_{2,f}, \tag{2.17}$$

where $\eta_{2,i}$ is the initial value of η_2, $\eta_{2,f}$ is the final value of η_2, and a_η, b_η are scaling parameters. η_1 is not tuned in the same way for several reasons: first, the lack of a θ^2 term makes the p_1 update naturally larger in size than the p_2 update. Second, the range of values p_1 is smaller than p_2, implying that the initial guess of $p_1[1]$ is likelier to be closer to optimal p_1 than $p_2[1]$. Note that the initial value of $p_1[1]$ represents an apriori guess of N_{UE}. Hence, larger initial values of $p_1[1]$ closer to true N_{UE} implies that the algorithm will converge more quickly. Similarly, the initial value $p_2[1]$ represents a guess on θ^*; when $p_2[n]$ is initialized such that $1/p_2[1]$ is close to θ^*, the algorithm converges more quickly.

In a real implementation of Alg. 1 after each discovery period, each UE would perform a single step of the stochastic gradient descent algorithm. Because the minimal value *discPeriod* can take is .32 seconds, and because an individual step of Alg. 1 is computationally simple, it is safe to assume that all UEs will have completed calculating θ for the next discovery period before that discovery period begins.

The SGD algorithm is known to converge to the local minimum of the objective function Eq. (2.14) in the vicinity of the initialization. While we cannot claim that this is the global minimum, we show through numerous simulations that Alg. 1 converges to the optimal values in Tab. 2.4 regardless of the initial conditions $p_1[1]$, $p_2[1]$. Tab. 2.5 shows the values of θ^A,

Algorithm 1: SGD based θ^* Determination (© 2021 IEEE)

Input : $\theta[n]$, $N_m[n]$

Initialize: $p_1[1]$, $p_2[1]$, trainingSequence = $[1, 0.5, 0.75, 0.25]$, L_{ts} = length(trainingSequence), stableCount = 0, Training = $True$

for *Each* UE_i *performing discovery* **do**

- **if** *Training* **then**
  - %perform SGD step based on $N_m[n]$
  - $p_1[n+1]$ =max$(p_1[n] - \eta_1 \frac{\partial}{\partial p_1}(T[n] - \frac{N_m[n]}{N_r})^2, 0)$;
  - $p_2[n+1]$ =max$(p_2[n] - \eta_2[n] \frac{\partial}{\partial p_2}(T[n] - \frac{N_m[n]}{N_r})^2, 0)$;
  - $\bar{p_2}[n]$ = mean$([p_2[n - a_\mu L_{ts} + 2], ..., p_2[n+1]])$;
  - **if** $|\bar{p_2}[n] - \bar{p_2}[n-1]| < \epsilon$ **then**
    - stableCount ++;
    - **if** $stableCount = b_\mu L_{ts}$ **then**
      - $\theta = \min\left(\max\left(\frac{1}{\bar{p_2}[n]}, 0.25\right), 1\right)$;
      - Training = $False$;
    - **else**
      - $\theta[n+1]$=trainingSequence$[(n \bmod L_{ts}) + 1]$;
    - **end**
  - **else**
    - stableCount = 0;
    - $\theta[n+1]$=trainingSequence$[(n \bmod L_{ts}) + 1]$;
  - **end**
- **else**
  - %Continue to use the θ learned during training
  - $\theta[n+1] = \theta[n]$;
- **end**

end

the average value of θ once training is complete, for various initial conditions over the range of possible solutions when $\theta^* = 0.5$ and confirms the claim[14].

Finally, training is deemed complete when the change in p_2 is sufficiently small such that the objective function has reached the minimum. Algorithm convergence may be determined in practice by a robust heuristic. We compute the average of the last $a_\mu L_{ts}$ values of p_2, where L_{ts} is the length of the training sequence, and a_μ is a scalar multiple. We then check if

[14]This table can be recreated using any values for N_r, N_t, N_{UE}, or initial conditions using our code found at https://git.io/JvTDZ.

the change is lower than a threshold ϵ for $b_\mu L_{ts}$ discovery periods, where b_μ is another scalar multiple. Once training has been completed, θ is set based on the average of the last $a_\mu L_{ts}$ parameter values.

$p_1[1]$	$p_2[1]$	θ^A	$p_1[1]$	$p_2[1]$	θ^A	$p_1[1]$	$p_2[1]$	θ^A
0.33	1	0.5065	0.66	1	0.5014	1	1	0.4935
0.33	2	0.4959	0.66	2	0.4965	1	2	0.5007
0.33	3	0.5056	0.66	3	0.5036	1	3	0.4938
0.33	4	0.4917	0.66	4	0.5004	1	4	0.4975

Table 2.5: Convergence Values for Alg. 1 when $N_r = 10$, $N_t = 10$, $N_{UE} = 20$, 1000 trials

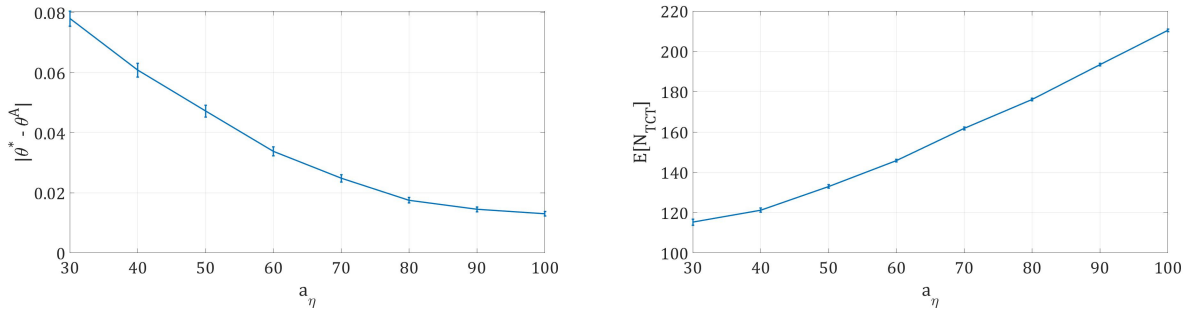
2.5.3 Results

Selection of Algorithm Parameters

We examine the effects of the simulation parameters, a_η, b_η, a_μ, and b_μ on the performance of the algorithm by evaluating two metrics; the error $|\theta^* - \theta^A|$, and the average number of periods to complete training, $E[N_{TCT}]$. Fig. 2.15 Shows the effect of a_η on these metrics.

By increasing a_η we extend the duration where η_2 is large, effectively extending the training. This reduces the error between θ^* and θ^A exponentially and increases $E[N_{TCT}]$ linearly. This pattern will become common among many other parameters. This trade-off is demonstrated in Fig. 2.16 by comparing $E[N_{DCT}]$ vs N_{UE} for various values of a_η.

When a_η is too low, the high error between θ^* and θ^A causes delays in discovery completion. While high values of a_η cause delays in discovery completion by extending training beyond what is needed. We found a good point that balanced these competing factors to be $a_\eta = 30$. For our subsequent simulations, we have chosen to use $a_\eta = 30$.



(a) (b)

Figure 2.15: Effects of a_η on $|\theta^A - \theta^*|$ and $E[N_{TCT}]$, Simulated, $N_{UE} = 20$, $N_r = 4$, $N_t = 4$, $, b_\eta = a_\eta/2$, $, a_\mu = 4$, $, b_\mu = 1$, with 95% Confidence Intervals ()

We next examined the effects of changing both a_μ and b_μ, shown in Figs. 2.17 and 2.18. We determined that when $b_\mu > a_\mu$ $E[N_{TCT}]$ was intolerably large ($E[N_{TCT}] \geq 2000$), thus our evaluation is limited to $b_\mu \leq a_\mu$.

When increasing a_μ and holding b_μ constant as in Fig. 2.17, both the error and $E[N_{TCT}]$ decrease rapidly to their asymptote. $E[N_{TCT}]$ follows this pattern because increasing a_μ implies more averaging, thus making $\bar{p_2}[n]$ smoother. However, as can be seen in Fig. 2.18 b_μ is the primary driver of error, and so long as a_μ is sufficiently greater than b_{mu}, the error will be minimized.

As a function of b_μ, the error increases linearly with b_μ. This pattern may seem counter-intuitive as one would expect that tighter bounds on the end condition would result in more accurate estimates. This pattern results from the fact that when the SGD algorithm is at the minimum, it fluctuates rather than being stable. In contrast, when the parameters are approaching the minimum, they are more stable. Thus, by forcing the period of stability to be longer, the only instances where the algorithm can meet the stopping condition are away from the minimum.

Based upon the results in Figs. 2.17 and 2.18, we selected $a_\mu = 3$ and $b_\mu = 1$ for the

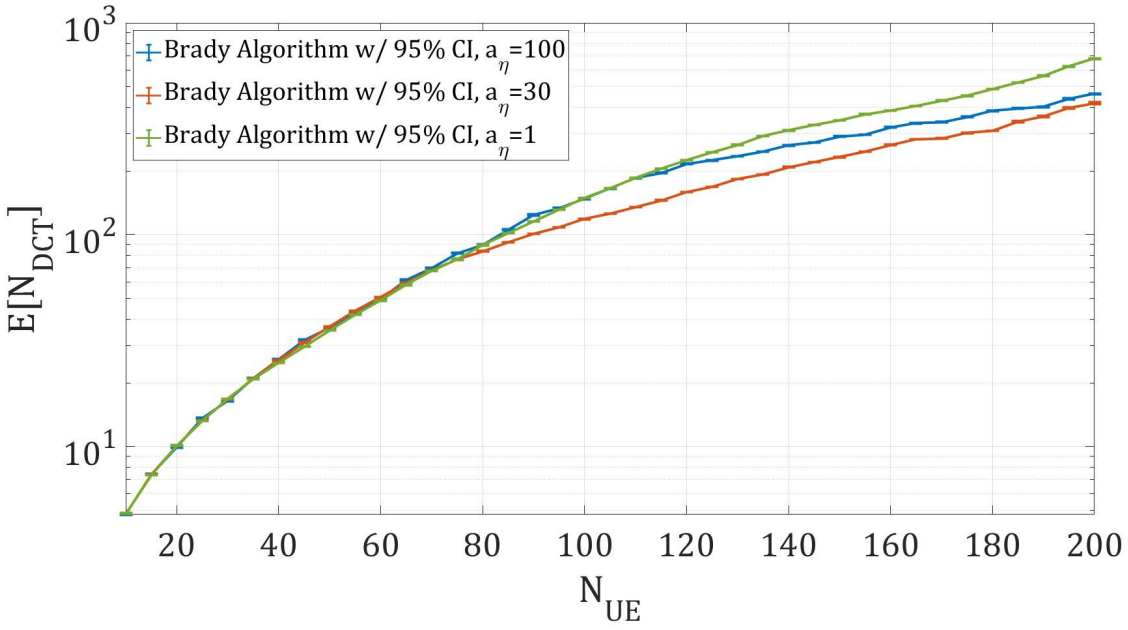


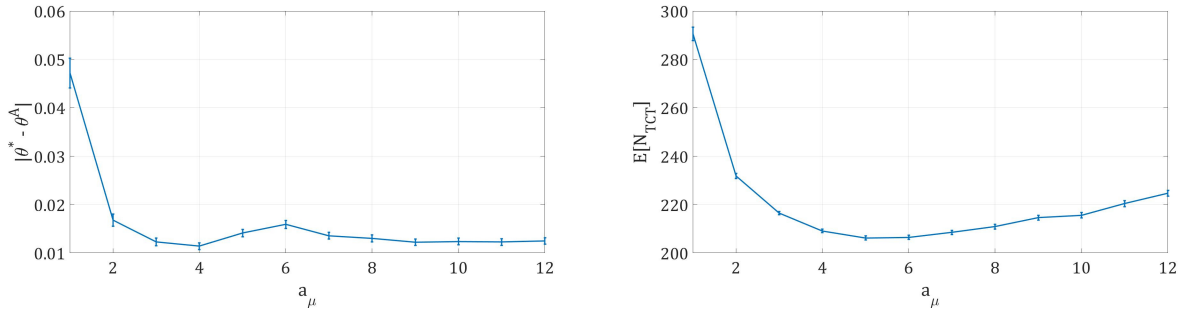
Figure 2.16: $E[N_{DCT}]$ vs N_{UE} for Various Values of a_η (© 2021 IEEE)

remainder of our simulations. Using the selected parameters, we sought to replicate the results of the genie-aided algorithm from Tab. 2.4. The adaptive θ algorithms results are shown in Tab. 2.6

For all parameter combinations in Tabs. 2.4 and 2.6 the difference is within a couple percentage points. The parameters for which our algorithm performs the worst are when θ^* is close to or exactly $\theta^* = 1$. This occurs because when θ^* is near 1 p_2 converges to a distribution with $E[p_2[\infty]] = 1/\theta^*$, however, because all values of p_2 that are below 1 are truncated to 1 $E[\theta^A] <= 1/E[p_2[\infty]]$. This effect is present when $\theta^* \approx 0.25$ but is less pronounced because there is no upper bound on p_2 where there is a lower bound $p_2 > 0$.

Algorithm Performance

To measure the performance of the algorithm, we varied N_{UE} and measured the effects on the expected number of periods to complete discovery, $E[N_{DCT}]$. For all simulations in this



(a) (b)

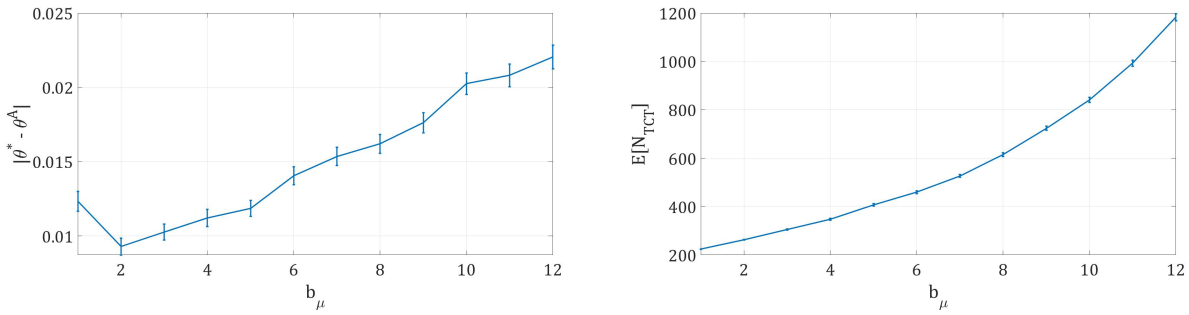
Figure 2.17: Effects of a_μ on $|\theta^A - \theta^*|$ and $E[N_{TCT}]$, Simulated, $N_{UE} = 20,\ , N_r = 4,\ , N_t = 4,\ , a_\eta = 100,\ , b_\eta = 50,\ , b_\mu = 1$, with 95% Confidence Intervals (© 2021 IEEE)

section, we used the parameter values listed in Tab. 2.7

Fig. 2.19 shows the performance of both the standards based constant θ algorithm ($\theta = 1$, and $\theta = 0.25$), along with our adaptive algorithm as a function of N_{UE}. Although we cannot calculate the value of θ^* for large values of N_{UE} it is clear from table 2.4 that when $N_{UE} >> N_r$ $\theta \to 0.25$, as such we assume that the performance of the standards based constant $\theta = 0.25$ algorithm is optimal as $N_{UE} \to \infty$.

Fig. 2.19 shows that for low values of θ, our adaptive algorithm performs worse than the constant $\theta = 1$ algorithm, however not significantly so. This is because, during training, UEs are participating in discovery at an effective constant $\theta = 0.625$. Although this is clearly sub-optimal when N_{UE} is low, the difference between the best and worst case is minimal when $N_{UE} = 10$ $E[N_{DCT}] \approx 2$ for $\theta = 1$ and $E[N_{DCT}] \approx 9$ for $\theta = 0.25$.

As N_{UE} grows and $\theta = 1$ becomes the sub-optimal case, the adaptive algorithm begins to track more closely to the $\theta = 0.25$ algorithm. At first, this is because, during training, the effective constant θ is lower than 1, so it tends to perform better. Indeed it can be seen that from $40 \leq N_{UE} \leq 70$, the curves for the adaptive algorithm and the constant $\theta = 0.25$



(a) (b)

Figure 2.18: Effects of b_μ on $|\theta^A - \theta^*|$ and $E[N_{TCT}]$, Simulated, $N_{UE} = 20,\ , N_r = 4,\ , N_t = 4,\ , a_\eta = 100,\ , b_\eta = 50,\ , a_\mu = 12$, with 95% Confidence Intervals (© 2021 IEEE)

algorithm are divergent, as $\theta^* = 0.25$. However, at $N_{UE} \approx 70$, the adaptive algorithm begins to turn back to and become parallel with the constant $\theta = 0.25$ algorithm's curve. This occurs because at $N_{UE} = 70$ $E[N_{DCT}] \approx 200$, this is approximately the same as $E[N_{TCT}]$ for our algorithm parameters. For $N_{UE} > 70$, the adaptive algorithm acts identically to the constant θ algorithm with $\theta = \theta^*$ with a constant additive factor from training.

Compared to the existing adaptive algorithm from [10] (henceforth the Mosbah algorithm), our algorithm shows significant improvements. As stated in the previous section, the Mosbah algorithm sets θ high and slowly reduces it as it discovers new UEs. This behavior results in an algorithm that works well for low N_{UE} and scales poorly as N_{UE} grows. When N_{UE} is high, the Mosbah algorithm only sets θ to low values once the discovery process is almost done, whereas our algorithm always does this after training, regardless of N_{UE}.

Overall our algorithm performs slightly worse in regions where the Mosbah algorithm performs best. The most significant discrepancy is for $N_{UE} = 75$ when on average, discovery completes in 62.92 periods when using the Mosbah algorithm while taking 76.41 periods

		N_{UE}					N_{UE}		
N_r	N_t	5	10	20	N_r	N_t	5	10	20
4	2	.6475	.3946	.3513	16	2	.8367	.7894	.5259
4	4	.7970	.4275	.3232	16	4	.9881	.9726	.7273
8	2	.8119	.5675	.3558	16	8	.9976	.9915	.8310
8	4	.9547	.7603	.3842	16	16	.9991	.9952	.8659
8	8	.9793	.8271	.4054	20	2	.8155	.8292	.6081
12	2	.8442	.7107	.4341	20	4	.9847	.9886	.8704
12	3	.9615	.8705	.5003	20	5	.9928	.9947	.8973
12	4	.9842	.9219	.5445	20	10	.9985	.9991	.9559
12	6	.9936	.9542	.5938	20	20	.9993	.9993	.9691
12	12	.9974	.9762	.6363					

Table 2.6: θ^A for various combinations of N_{UE}, N_r, and N_t ()

when using our algorithm, a difference of 4.32 seconds[15]. While at the highest N_{UE} we tested, $N_{UE} = 200$, discovery completes in 3062 periods on average when using the Mosbah algorithm while taking 417.9 periods when using our algorithm, a difference of 846.1 seconds[15].

2.5.4 Effects of Relaxing Simultaneity Assumption

Throughout this Chapter, we have assumed that all UEs begin discovery simultaneously due to a sudden power outage. This assumption has the effect of causing all UEs to perform training in unison (each UE transmits with the same θ during each period of training) until some UEs complete training. We feel this assumption is reasonable because discovery periods are periodic with relatively large periods (minimally .32 s). For our simultaneity assumption to hold, all UEs are required to initiate sidelink discovery after a network outage within a

[15] Assuming a discovery period of .32 seconds.

Parameter	Value Used
N_f	2
N_t	10
a_η	100
b_η	50
a_μ	4
b_μ	1
ϵ	.001
$trainingSequence$	[1,.05,0.75,0.25]

Table 2.7: Parameter Values used During Performance Evaluation (© 2021 IEEE)

block of time equal to *discPeriod*.

However, if that assumption were relaxed (not all UEs begin discovery at the same time), this would have implications for the adaptive algorithm's performance. To test these effects, we introduced one of two random shifts, S, to each UE's training sequence. Meaning that in Alg. 1 instead of $\theta[n+1] = trainingSequence[(n \, mod \, L_{ts}) + 1]$ during training $\theta[n+1] = trainingSequence[(n \, mod \, L_{ts}) + 1 + S]$.

We selected two distributions for S, uniform and Poisson with mean .1. These two distributions represented a worst-case scenario, UEs are completely out-of-sync, and a milder scenario, where only a small number of UEs are out-of-sync. The results of these experiments are shown in Fig. 2.20.

In both cases adding a random shift reduces the performance of the adaptive algorithm, although in both cases, it still performs better than the algorithm presented in [10]. The results with random shifts appear similar to those in Fig. 2.16 where the poor choice of a_η resulted in high error between θ^* and θ^A. This behavior makes sense as the shift introduces errors into the measurements taken during training, resulting in selecting poor values of θ^A.

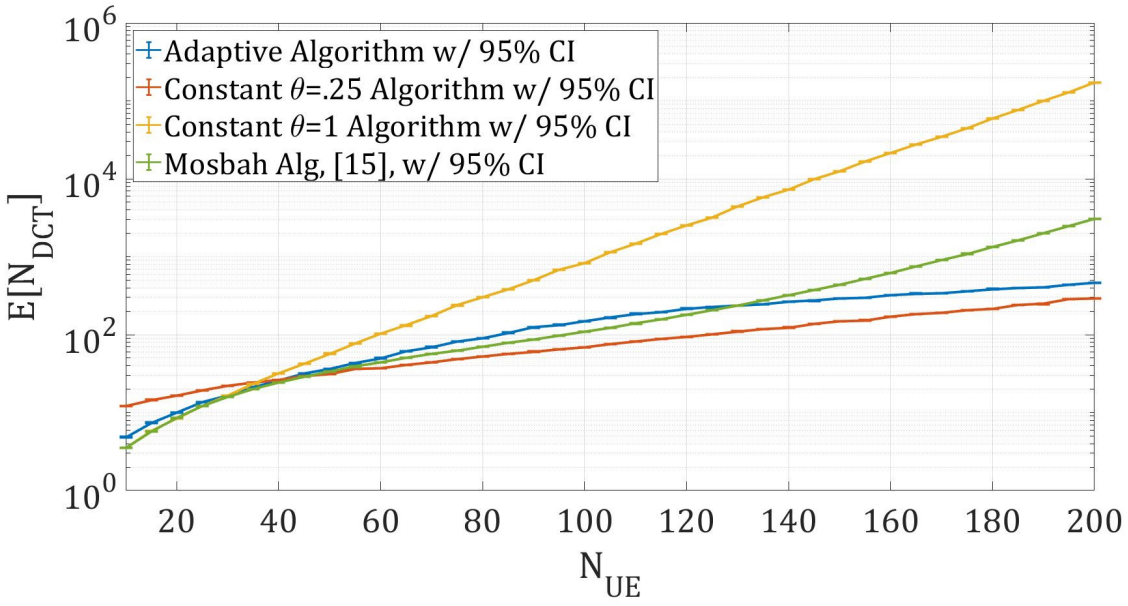


Figure 2.19: Performance Comparison of Algorithm to Constant θ, Simulated, $N_r = 20\, N_t = 10$, with 95% Confidence Intervals ()

To correct for the effect of random shifts, one would need to have knowledge of $\bar{\theta}[n]$, the average transmission probability of all UEs participating in discovery during period n, and replace $\theta[n]$ with $\bar{\theta}[n]$ in Eq. 2.13.

2.6 Summary

We presented an analytical simulation study (using ns-3 and Matlab) examining the LTE sidelink discovery process. We validated the mathematical model for the discovery process using the ns-3 simulations. We used the simulations to examine the difference between the HD and FDD assumptions and the effects of resource pool size and the number of UEs on the discovery completion time. Further, we explored the impact of resource pool size and the number of UEs on the discovery completion time. We used insights gained from our mathematical models to develop a stochastic gradient descent-based algorithm to adapt the transmission probability θ based on measurements during a training phase.

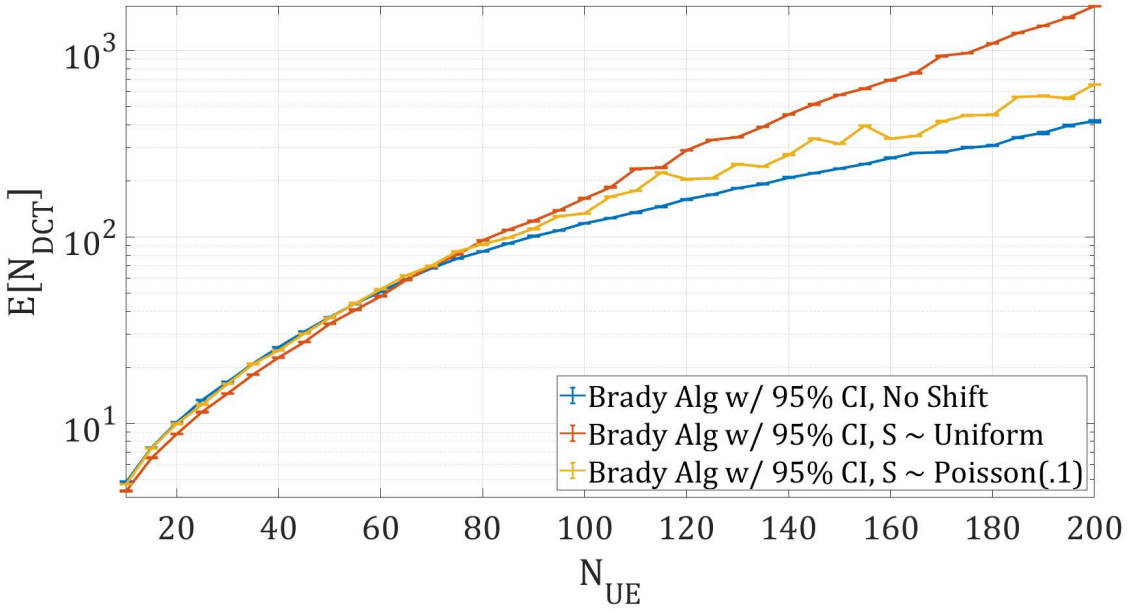


Figure 2.20: Effects of Shift Type (© 2021 IEEE)

Through Matlab simulations, we determined that our proposed algorithm converged to the correct transmission probability regardless of initialization parameters. We determined that our algorithm performs comparably to the algorithm in [10] for $N_{UE} < E[N_{TCT}]$ and significantly better for $N_{UE} > E[N_{TCT}]$. We found that our algorithm scales well with N_{UE}, with $E[N_{DCT}]$ increasing at the same rate as the lower bound, plus a constant offset of $E[N_{TCT}]$ discovery periods.

2.7 Concluding Remarks

The key enhancement made by our algorithm is to provide a method of estimating the UE density during operation so as to avoid a poor selection of discovery algorithm parameters. With the estimated UE density we can select the optimal transmission probability θ for that UE density. This is important, especially for large UE densities, because the difference between the best and worst transmission probabilities (for a given UE density) results in

increases in the discovery completion time of up to approximately 100 minutes for the highest tested UE density.

We importantly found that Setting the transmission probability myopically (that is, the same at the beginning of the discovery process at the end) is optimal, rather than changing it over time. This is because the primary driver of discovery completion time is decoding new discovery messages, whose primary driver is in turn decoding packets. The reason that decoding new discovery messages is primarily driven by decoding new packets is that there is no mechanism to tell other UEs that you have already decoded their discovery message, so they will continue to participate in discovery regardless of if you have decoded their packet[16]. The only real way to control packet decode probability in discovery is to tune the transmission probability, thus reducing collisions. This is also convenient because it is difficult to know how far through the discovery process you are.

Other proposed algorithms, like those in [10], fail to achieve the same level of results as ours because they are worse at estimating UE density. The proposed algorithm in [10] effectively tunes θ over time by estimating the total number of UE to be the number of discovery messages decoded, thus, the algorithm only converges to the correct value once most UEs are already discovered.

[16]Even if there was, the goal of that other UE is for all other UE to decode their packets which it cannot know it has done just through a positive ACK by one UE.

Chapter 3

5G NR CELLULAR VEHICLES-TO-ANYTHING THROUGHPUT: ANALYSIS AND ENHANCEMENTS

3.1 Motivation

3rd Generation Partnership Project (3GPP) Rel-12 defines communications modes 1 (on-network) and 2 (off-network) for Sidelink (SL). In this Chapter, we will focus on mode 2, which enables distributed packet transmission. Cellular Vehicle-to-Anything (C-V2X) introduced a sensing-based Semi-Persistent Scheduling (SPS) Scheme based on distributed channel sensing that requires UEs to independently monitor the shared channel to determine which resources were in use and avoid transmitting using those resources in the near future. A key original application was to support periodic status messages with low latency requirements like Basic Safety Messages (BSMs). In Fifth Generation (5G) New Radio (NR), this has expanded to non-periodic packets (called dynamic transmissions), but for this Chapter, we limit packet transmissions to periodic data types such as BSMs.

Mode 2 resource allocation relies on sensing-based SPS to determine the available Physical Resource Blocks (PRBs) for transmission. This determination is based on a selection window spanning a number of subframes, defined by parameters T_1 and T_2 (see Fig. 3.1). The set of candidate PRBs is chosen based on the SPS algorithm detailed in Sec. III, incorporating "blind retransmission", or duplicates of the original packet[1] to exploit repetition coding gains.

The primary contribution of this Chapter is to introduce two new distributed adaptive algorithms for mode 2 SL operation. The first - Dynamic Retransmissions (D-Re) - is designed to make PHY packet reception more robust to lack of foreknowledge of user equipment

[1]The number of retransmissions is determined before operation.

(UE) density ρ_{UE} and is standards compliant. The second - Opportunistic Retransmissions (O-Re) - exploits knowledge of reselection counter values[2] and zone IDs as an alternative non-standards compliant algorithm to the 3GPP defined baseline SPS reselection algorithm. These algorithms are designed to improve throughput via joint PHY/MAC design to achieve more robust communications in the absence of foreknowledge of ρ_{UE}. The O-Re algorithm improves throughput over the standards specified SPS at high ρ_{UE}. To support these claims, we conduct the first publicly reported network simulations of SPS using the open-source ns-3 simulator's (www.nsnam.org) 5G stack implementations and validate results against analytical models for Packet Reception Rate (PRR) as a function of transmitter-receiver distance $d_{t,r}$, the number of retransmissions N_{Se}[3], the UE density ρ_{UE}, and the average per-UE throughput $\Lambda(d_{t,r}, N_{Se}.)$. This is done first for the 3GPP-specified SPS reselection algorithm to set a performance baseline that subsequently allows us to determine values of N_{Se} that maximize throughput as a function of ρ_{UE}.

Major materials from this Chapter are presented in the author's previous publication [28], [29][4].

3.2 Model for C-V2X Sidelink

3.2.1 Channel Modeling & Resource Allocation in Mode 2

As shown in Fig. 3.1, the NR C-V2X SL mode 2 allocates resources in time-frequency, consisting of N_{sc} sub-channels (frequency) and slots (time), respectively. Each sub-channel is composed of 10, 15, 20, 25, 50, 75, or 100 contiguous resource blocks (RBs) or a set of twelve contiguous sub-carriers. A combination of one sub-channel and one slot constitutes a Physical Resource Block (PRB). 5G NR system resources are configured depending on the choice of numerology $\mu \geq 0$ that determines both the subcarrier spacing and the length of

[2]This is a proposed addition to the 3GPP standards.

[3]For the standards algorithm, the retransmission parameter N_{Se} is set blindly. Our proposed algorithms, in contrast, exploit channel information to select N_{Se}

[4]

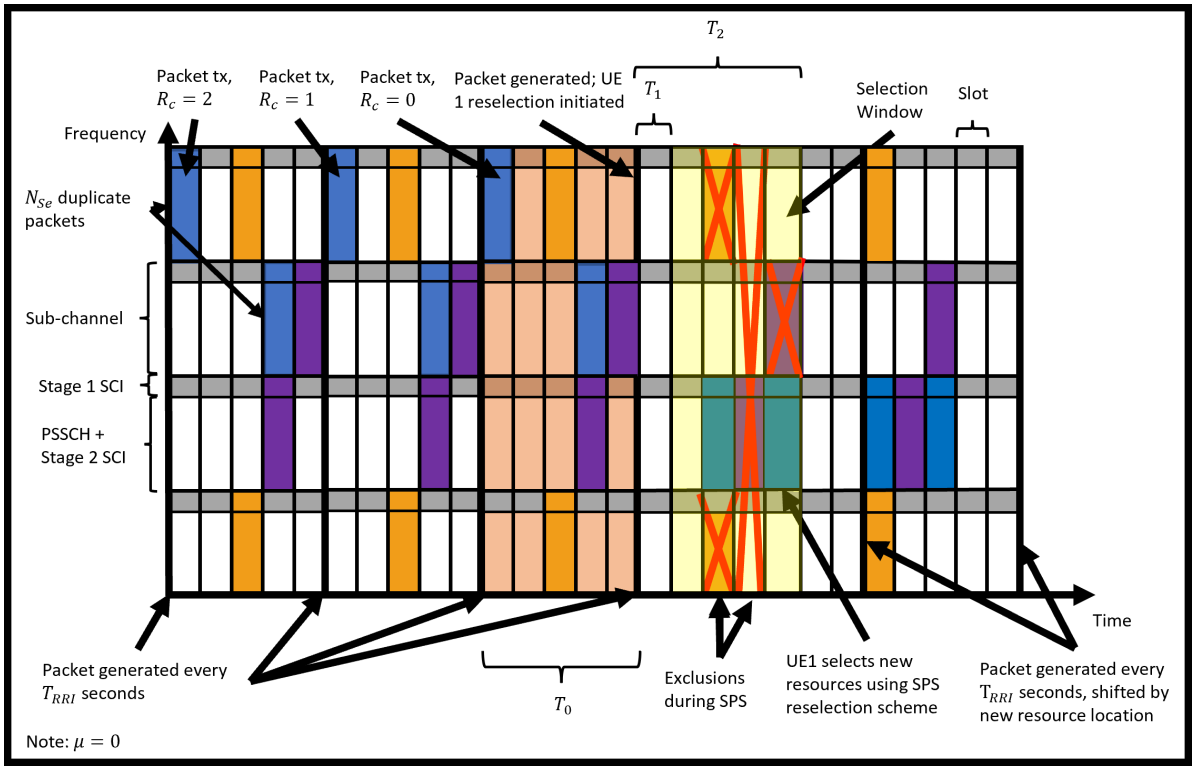


Figure 3.1: Channel Structure of NR C-V2X Mode 2

a slot. C-V2X supports numerologies $\mu = 0, 1$ or 2, we set $\mu = 0$ throughout this Chapter.

Mode 2 packet transmission is comprised of stage 1 Sidelink Control Information (SCI) which uses 2 RBs, and the Physical Sidelink Shared Channel (PSSCH) plus stage 2 SCI, which takes up the remainder of the subchannel. The Stage 1 SCI contains information on the packet's priority, location in time, and frequency of the stage 2 SCI, Resource Reservation Interval (RRI), and what Modulation and Coding Scheme (MCS) is used by the second stage SCI [38]. In this Chapter, we propose that the stage 1 SCI also contain the resource reservation counter value R_c. PSSCH data is sent using allowable MCSs defined in [39]. The second stage SCI contains the MCS of the PSSCH, source, and destination ID (for unicast and groupcast), the new data indicator, and Hybrid Automatic Repeat Request (HARQ) process ID [38]. NR C-V2X allows for the Time Division Duplex (TDD) of the slots between

the sidelink and uplink. Because we are modeling off-network NR C-V2X only, we will assume that all PRBs are devoted to sidelink.

We consider NR C-V2X mode 2 for periodic transmissions of BSMs during infrastructure-less (i.e., in the absence of gNB) operations that are generated and transmitted by each UE once every RRI, using one PRB. UE scheduling fundamentally relies on 3GPP-defined sensing-based Semi Persistent Scheduling (SPS), wherein UEs self-select PRB candidates for use based on sensing results over a window of duration T_0 s. In this Chapter, we set $T_0 = .1s$ (RRI duration). Upon access, a UE transmits N_{Se} duplicates of BSM in every RRI (T_{RRI}) s interval[5], using PRBs selected during the previous reselection window. If the UE decodes any number of duplicate packets, then the packet is considered decoded. For simplicity, we assume no recombining is performed. The UE is allowed a reservation of the PRBs selected for a number of consecutive bursts equal to the reselection counter value R_c, also chosen during the previous reselection.

Fig. 3.1 shows an illustrative example of the channel structure of NR C-V2X mode 2 supporting SPS, with $N_{sc} = 4$ sub-channels, $\mu = 0$ (hence each slot equals a sub-frame of duration .001 s), $T_{RRI} = .005$ s, $T_0 = RRI$ s, $T_1 = 1$ slots, and $T_2 = 5$ slots. Once a UE has transmitted R_c consecutive packet bursts, it must initiate reselection. For the example in Fig. 3.1, the UE whose packets are marked in blue initiates reselection in the window shown in yellow. This duration is defined by the boundary parameters $T_2 > T_1$ slots after the packet, which caused reselection to be triggered, was generated. The SPS reselection algorithm aims to identify candidate PRBs for subsequent use that are determined to be not currently in use by other UEs. To accomplish this, the UE measures the Reference Signal Received Power (RSRP) one RRI before the slot in question in the sensing window (the salmon region). Based on the RSRP, the UE first eliminates all PRBs as potential selections via the condition

[5]RRI can take values from $[.001, .1]$ s or in multiples of .1 s thereafter, between .2 to 1 s. For the remainder of this Chapter, we fix $T_{RRI} = .1$ s.

$$RSRP > \gamma_{SPS} = \gamma_{baseSPS} + \gamma_X. \tag{3.1}$$

In the sensing window (yellow region) in Fig. 3.1, we see three PRBs with red Xs that have been excluded due to Eq. (3.1) being met. In Eq. (3.1), γ_{SPS} is the SPS threshold determined after exclusions have been finalized and equals $\gamma_{baseSPS}$, the base SPS threshold preset based on 3GPP recommendations, enhanced by γ_X, where the increase is required to ensure that the proportion of PRBs remaining for selection in the reselection window is no less than a proportion X[6]. If $\gamma_{baseSPS}$ is initialized to a low value, it can result in potentially denying PRBs for use that could have led to successful data transmission by reference UE_t, even in the presence of secondary transmission. The role of γ_X is to correct this by ratcheting up the threshold as needed in denser environments to allow more resources to be potentially available. To determine γ_{SPS}, a UE first determines the number of exclusions with $\gamma_X = 0$; then, if the proportion of resources remaining for selection relative to the original # is less than X, the UE increases γ_X by 3 dB and repeats the process, until the proportion is above X. For instance, the selection window in Fig. 3.1 contains 16 PRB, so for $X = .2$, the UE would increase γ_X until at least 4 PRB were available for selection.

In addition, the UE also excludes all PRBs which are in slots that the UE previously transmitted in due to the Half Duplex (HD) effect, [1]. In Fig. 3.1, the PRBs that have been eliminated because of the HD effect are shown as a column with a red X in the selection window (yellow region).

After the exclusions are finalized, the UE selects N_{Se} of the remaining PRBs at random for transmission, along with a new (random) value for the reselection counter R_c, chosen as a discrete random variable with limits determined by RRI value. For RRI = .1 s, R_c is selected uniformly between $[5, 15]$.

---

[6] X can take values .1, .2, or .5; we assume $X = .2$.

3.2.2 PHY Layer - Channel Model and UE Layout

Our network topology is an infinite highway, as shown in Fig. 3.2, with ℓ lanes in each direction, populated by nodes placed with density ρ_{UE} per unit distance in each lane. In simulations, we assume that the channel is modeled according to the 3GPP Vehicle-to-Vehicle (V2V) Highway model [25] that incorporates distance-dependant deterministic path loss and a random shadow-fading component.

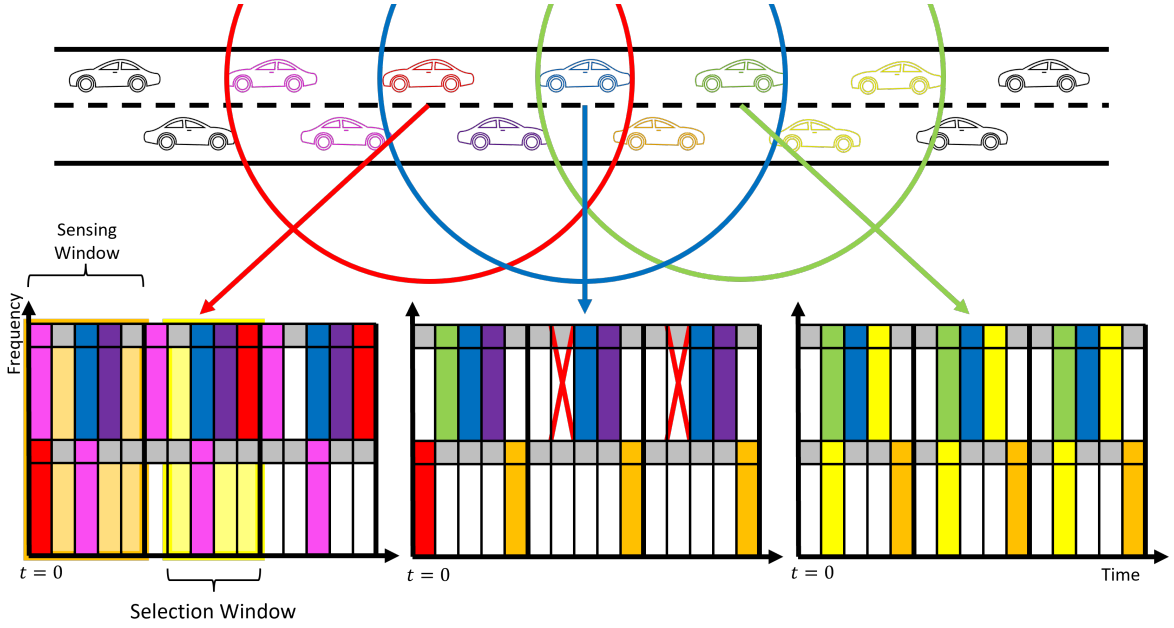


Figure 3.2: Example of Hidden Nodes Causing Collisions

UEs are assumed to perform energy detection to determine if there exist packet(s) in PRBs to decode. This is performed by checking if the RSRP in a PRB is above a threshold γ_{FTR}; if the RSRP exceeds the threshold, the UE attempts to decode the packet(s). If one packet occupies the PRB exclusively, it is decoded with near probability one in the absence of collision or HD errors. The consequence of the detection threshold is the existence of hidden nodes and resulting collisions, as shown in Fig. 3.2.

In Fig. 3.2, we can see three transmission ranges highlighted along with their channels

synchronized beginning at $t = 0$. Note that everything is color-coded, so the blue UE corresponds to the blue transmission range and the blue PRB in the channels. In the example, the red UE performs reselection and randomly selects one of the 4 PRBs it believes to be empty based on the sensing window. However, as we can see from the blue UE's perspective, the PRB it selects is actually occupied by the green UE, causing a collision shown as a red X.

3.3 Per-UE Average Throughput, ($\Lambda(d_{t,r}, N_{Se})$)

The average per-UE throughput at a transmit-receive distance $d_{t,r}$ with N_{Se} blind retransmissions is given by

$$\Lambda(d_{t,r}, N_{Se}) = \frac{1 - (1 - P_{PRR}(d_{t,r}, N_{Se}))^{N_{Se}}}{T_{RRI}}. \tag{3.2}$$

$\Lambda(d_{t,r}, N_{Se})$ is a function of the average packet generation rate $1/T_{RRI}$ and the probability of decoding one of the N_{Se} packets, $P_{PRR}(d_{t,r}, N_{Se})$. The exponent in the numerator of (3.2) represents the diversity gain from blind retransmissions. While the event probabilities of subsequent unique packet receptions are correlated, the receptions are approximately conditionally uncorrelated once reselection occurs.

3.3.1 PRR, $P_{PRR}(d_{t,r}, N_{Se})$

$P_{PRR}(d_{t,r}, N_{Se})$ represents the probability that one of the N_{Se} duplicate packets is correctly decoded. This is expressed in terms of three types of errors that exist in NR-CV2X:

i. Half-Duplex (HD) errors (E_{HD}) occur because UEs are incapable of transmitting and receiving simultaneously in a slot. So, when all N_{Se} packets transmitted by UE_t and UE_r use the same slot, UE_r cannot detect UE_t's packet.

ii. Finite Transmission Range (FTR) errors (E_{FTR}) occur when UEs are unable to detect packets because the receive power falls below the detection threshold γ_{FTR}. This occurs because the UEs are too far apart or channel conditions are poor.

iii. Collision errors (E_{COL}) occur when one or more secondary UEs select the same N_{Se} PRBs used by UE_t to transmit, causing decode failure at intended UE_r. This scenario occurs when UEs are outside the FTR of UE_t (hidden nodes) but can also happen if another UE performs reselection simultaneously with UE_t. We define the three error probabilities as follows:

$$\delta_{HD}(N_{Se}) = P(E_{HD}), \tag{3.3}$$

$$\delta_{FTR}(d_{t,r}, \gamma_{FTR}) = P(E_{FTR}|\ \bar{E}_{HD}), \tag{3.4}$$

$$\delta_{COL}(d_{t,r}, N_{Se}) = P(E_{COL}|\ \bar{E}_{HD},\ \bar{E}_{FTR}), \tag{3.5}$$

where $d_{t,r}$ is the distance between UE_t and UE_r and γ_{FTR} is the detection threshold. Hence P_{PRR} is obtained as

$$P_{PRR}(d_{t,r}, N_{Se}) = (1 - \delta_{HD}(N_{Se}))(1 - \delta_{FTR}(d_{t,r}, \gamma_{FTR}))(1 - \delta_{COL}(d_{t,r}, N_{Se})), \tag{3.6}$$

Half-Duplex Errors, $\delta_{HD}(N_{Se})$

This error represents the probability that the packet sent by UE_t occupies the same slot as one of the N_{Se} packets sent by UE_r for its transmission. If $N_{Se} = 1$, then this probability would simply be $1/(1000 \cdot T_{RRI})$, where T_{RRI} is specified in ms. More generally, $\delta_{HD}(N_{Se})$ is defined in Eq. (3.7), with the first term representing the probability of selecting the same slot as one of UE_r's transmissions while the second term represents the average number of slots occupied by UE_r's transmissions. The second term is less than N_{Se} because UE_r can transmit multiple (up to N_{sc}) times in one slot.

$$\delta_{HD}(N_{Se}) = \left(\frac{1}{1000 T_{RRI}}\right)\left(N_{Se} - \frac{(N_{Se} - 1)(T_2 - T_1 + 1)}{1000 T_{RRI}}\right). \tag{3.7}$$

Finite Transmission Range Errors, $\delta_{FTR}(d_{t,r}, \gamma_{FTR})$

This error represents the probability that the receive power of a packet sent by UE_t to UE_r at distance $d_{t,r}$ falls below the detection threshold γ_{FTR}. We assume that the channel between UE_t and UE_r is identical for all N_{Se} copies of the packet. We define the receive power in dBm as

$$P_r(d_{t,r}) = P_t - L(d_{t,r}) - S_F, \tag{3.8}$$

where P_t is the transmit power in dBm in one PRB, L is the distance-dependant path loss component, and S_F is the additive shadowing loss. As we assume that the channel is modeled according to the 3GPP V2V Highway model [25], with log-normal shadow fading with standard deviation σ_{sf}, $\delta_{FTR}(d_{t,r}, \gamma_{FTR})$ is given by

$$\delta_{FTR}(d_{t,r}, \gamma_{FTR}) = \frac{1}{2} erfc\left(\frac{P_t - L(d_{t,r}) - \gamma_{FTR})}{\sigma_{sf}\sqrt{2}}\right). \tag{3.9}$$

If $P_r < \gamma_{FTR}$, the packet is not detectable, and the transmitting UE is said to be hidden from the receiver.

Collision Errors, $\delta_{COL}(d_{t,r}, N_{Se})$

This term represents the probability that when UE_r attempts to decode the packet from UE_t, all N_{Se} PRB are occupied by at least one interfering transmission. Multiple UEs can occupy the same PRB for several reasons, the most common being that UE_t cannot detect interfering UEs denoted by UE_{int} when selecting PRBs during reselection (akin to the hidden node problem). This error can also occur because UE_t and UE_{int} both initiate reselection simultaneously and select the same PRB.

We thus decompose collision error into two parts; $p_{SIM}(d_{t,int}, N_{Se})$ is the probability that UE_t and UE_{int} select the same PRB to transmit while $p_{INT}(d_{t,r}, d_{int,r})$ is the probability that given that UE_t and UE_{int} select same PRBs, the interference from UE_{int} results in block

decoding error for UE_t packet. Overall, the probability of all N_{Se} packets experiencing a collision error is

$$\delta_{COL}(d_{t,r}, N_{Se}) = 1 - \prod_i (1 - p_{SIM}(d_{t,int}, N_{Se}) p_{INT}(d_{t,r}, d_{int,r}), \tag{3.10}$$

$p_{INT}(d_{t,r}, d_{int,r})$

To determine $p_{INT}(d_{t,r}, d_{int,r})$, we first define the Signal to Interference and Noise Ratio (SINR) in dB as

$$SINR(d_{t,r}, d_{int,r}) = P_r(d_{t,r}) - P_I(d_{int,r}) - N_0, \tag{3.11}$$

where $P_I(d_{int,r})$ is the receive power of the packet from UE_{int} at UE_r and N_0 is the noise power. The probability that the receiver experiences a decode error is

$$p_{INT}(d_{t,r}, d_{int,r}) = \sum_{s=-\infty}^{\infty} BL(s) f_{SINR(d_{t,r}, d_{int,r})}(s), \tag{3.12}$$

where $BL(s)$ is the BLER for SINR $= s$[7], and $f_{SINR(d_{t,r}, d_{int,r})}(s)$ is the probability density function for SINR.

$p_{SIM}(d_{t,int}, N_{Se})$

Packets can simultaneously select the same PRB under the following conditions; (i) both UE_t and UE_{int} have overlapping selection windows, and (ii) both select the same PRB from that overlap, defined as P_O and $P_{Sh|O}$ as the first and second terms in Eq. (3.13) respectively. In the example in Fig. 3.2, the red and green UEs have $P_O = 1$ because $1000 T_R RI = T_0$ and $P_{Sh|O} = 4/5$ because 4 of the 5 slots overlap.

Next, UE_t and UE_{int} must fail to take each other's transmissions into account during reselection. This can occur because UE_t cannot detect any of the packets sent by UE_{int} with

[7]Obtained by invoking the error model from [40].

probability $\delta_{FTR}(d_{t,int}, \gamma_{SPS})^{N_{Se}}$, or if they can, both reselect simultaneously with probability $1 - T_{RRI}(1 - P_{Keep})$. The probability of either of these events occurring is $1 - (1 - T_{RRI}(1 - P_{Keep}))(1 - \delta_{FTR}(d_{t,int}, \gamma_{SPS})^{N_{Se}})$. In the example in Fig. 3.2, the first probability is 1 for the red and green UEs (they cannot detect each other's packets) and equals 0.1 for the probability they both reselect simultaneously.

Finally, if both UEs have overlapping selection windows, and both either reselect simultaneously or are hidden from each other, then the probability of selecting the same PRB simultaneously is the probability that the transmitter randomly selects one of the $\overline{C}_{C_a}(N_{Se}, \gamma_{SPS})$ common candidate PRBs[8] from among their $\overline{N}_{C_a}(N_{Se}, \gamma_{SPS})$ total candidate PRBs $(\overline{C}_{C_a}(N_{Se}, \gamma_{SPS})/\overline{N}_{C_a}(N_{Se}, \gamma_{SPS}))$ times the probability that the interferer selects the same PRB $(1/\overline{N}_{C_a}(N_{Se}, \gamma_{SPS}))$. In the example in Fig. 3.2, for the red and green UEs, this equals $2/5^2$ because there are $\overline{C}_{C_a}(N_{Se}, \gamma_{SPS}) = 2$ common candidates among the $\overline{N}_{C_a}(N_{Se}, \gamma_{SPS}) = 2$ total candidates in the selection window. Hence the final expression for $p_{SIM}(d_{t,int}, N_{Se})$ is given by Eq. (3.13)

$$p_{SIM}(d_{t,int}, N_{Se}) = P_O P_{Sh|O} \Bigg(1 - (1 - T_{RRI}(1 - P_{Keep})) \cdot \left(1 - \delta_{FTR}(d_{t,int}, \gamma_{SPS})^{N_{Se}}\right) \Bigg) \frac{\overline{C}_{C_a}(N_{Se}, \gamma_{SPS})}{\overline{N}_{C_a}(N_{Se}, \gamma_{SPS})^2}. \tag{3.13}$$

P_0 is calculated assuming UE transmissions are uniformly distributed in time. Thus the probability of overlap is the total number of slots they (UE_t and UE_{int}) both occupy divided by the number of slots in a resource reservation interval. Therefore P_O is given by

$$P_O = \frac{2(T_2 - T_1 + 1) - 1}{1000 t_{RRI}}. \tag{3.14}$$

$P_{Sh|O}$ is the probability that both choose one of the $N_{sc}N_{Sh}$ overlapping PRB out of the total N_r they have available to choose from. Therefore $P_{Sh|O}$ is given by

[8]Common candidates being those PRBs which would remain after SPS exclusions for both UEs.

$$P_{Sh|O} = \left(\frac{N_{sc}N_{Sh}}{N_r}\right)^2, \tag{3.15}$$

where N_{sc} is the number of sub-channels, N_{Sh} is the average number of shared resources given the selection windows overlap and is given by

$$N_{Sh} = \frac{(T_2 - T_1 + 1)^2}{2(T_2 - T_1) + 1}. \tag{3.16}$$

Note that γ_X impacts γ_{SPS} in Eq. (3.13) as an internal parameter that effectively shrinks the FTR *exclusively* for the purposes of ensuring a minimum target value for the 'candidate' PRB set. Hence if $\gamma_{FTR} < P_r(d_{t,r}) < \gamma_{SPS}$, the packet can still be decoded in the absence of half duplex or collision errors. Further, because γ_X is the increase in γ_{SPS} required to limit the ratio of occupied to total PRB to X, we can select γ_X via

$$\gamma_X = \underset{\gamma}{\operatorname{argmin}} \left| \frac{\overline{N}_{C_a}(N_{Se}, \gamma_{SPS} = \gamma_{baseSPS} + \gamma)}{N_{sc}\overline{N}_{Sh}} - X \right|, \tag{3.17}$$

where $\overline{N}_{Sh}$ is the average number of overlapping PRB in the selection windows of UE_t and UE_{int}. As will be seen further, due to $\overline{N}_{C_a}(N_{Se})$ dependence on γ_X, the latter can't be obtained in closed form.

The average number of common candidate PRBs, $\overline{C}_{C_a}(N_{Se}, \gamma_{SPS})$, is the total number of PRBs $N_{sc}\overline{N}_{Sh}$, minus the average number of common exclusions, $\overline{C}_{E_x}(N_{Se}, \gamma_{SPS})$.

$$\overline{C}_{C_a}(N_{Se}, \gamma_{SPS}) = N_{sc}\overline{N}_{Sh} - \overline{C}_{E_x}(N_{Se}, \gamma_{SPS}) \tag{3.18}$$

UE_t and UE_{int} can both exclude the same PRBs for two reasons: a) The PRB selected by the $\overline{N}_B(N_{Se}, \gamma_{SPS})$ UEs fall within the FTR or b) Some of the $\overline{N}_{Exc}(N_{Se}, \gamma_{SPS})$ UEs that fall within the FTR of each exclusively happen to select the same PRB for transmission.

In general $\overline{N}_B(N_{Se}, \gamma_{SPS})$ and $\overline{N}_{E_x}(N_{Se}, \gamma_{SPS})$ are

$$\overline{N}_B(N_{Se},\gamma_{SPS}) = \iint\limits_{d_{t,r},d_{int,r}\in\mathbb{R}} \ell\rho_{UE}N_{Se}(1-\delta_{FTR}(d_{t,r},\gamma_{SPS}))\cdot (1-\delta_{FTR}(d_{int,r},\gamma_{SPS}))dd_{t,r}dd_{int,r}, \tag{3.19}$$

and

$$\overline{N}_{Exc}(N_{Se},\gamma_{SPS}) = \iint\limits_{d_{t,r},d_{int,r}\in\mathbb{R}} \ell\rho_{UE}N_{Se}(1-\delta_{FTR}(d_{t,r},\gamma_{SPS}))\cdot (\delta_{FTR}(d_{int,r},\gamma_{SPS}))dd_{t,r}dd_{int,r}. \tag{3.20}$$

To determine $\overline{N}_{C_a}(N_{Se},\gamma_{SPS})$ and $\overline{C}_{E_x}(N_{Se},\gamma_{SPS})$, we assume that if any UE successfully transmits using a PRB, then the selected PRBs will be excluded during SPS.

Finally, the state of any UE performing reselection (the total number of occupied PRBs) is represented by the Markov chain shown in Fig. 3.3 with accompanying transition matrix T.

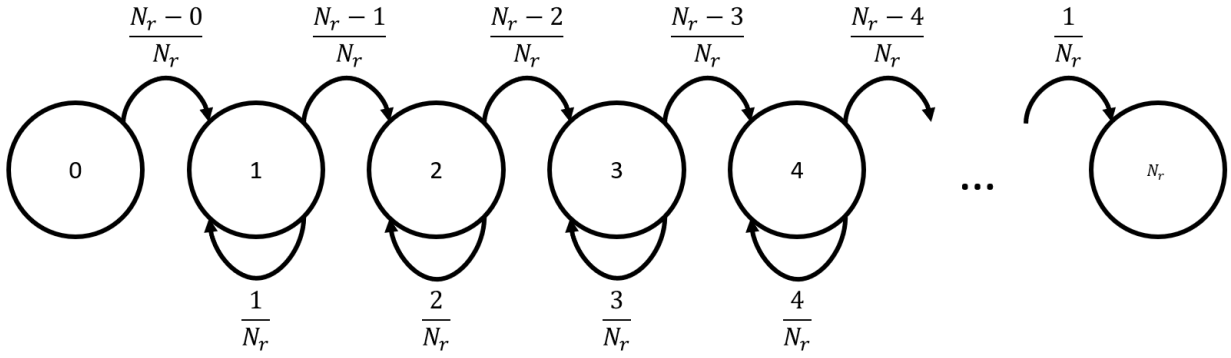


Figure 3.3: Markov Chain representing PRB occupancy, states representing the number of occupied PRB

The probability that the number of excluded PRBs, E_x equals k, given that N UEs are selecting PRBs, there are M PRBs to select from, and S UEs have already chosen PRBs, is given by

$$P_{Ex}(k|N, M, S) = [T_M^N]_{\{S,k\}}, \tag{3.21}$$

which represents the $\{S, k\}^{th}$ element of transition matrix T_M^N. Taking the expectation of Eq. (3.21) we know that $\overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS})$, the expected number of PRBs excluded due to the $\overline{N}_B(N_{Se}, \gamma_{SPS})$ UEs, is

$$\overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS}) = E[E_x|N = \overline{N}_B(N_{Se}, \gamma_{SPS}), M = N_{sc}\overline{N}_{Sh}, S = 0], \tag{3.22}$$

so overall $\overline{C}_{E_x}(N_{Se})$

$$\begin{aligned}
&\overline{C}_{E_x}(N_{Se}, \gamma_{SPS}) = \overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS}) \ + \frac{1}{N_r - \overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS})} \cdot \\
&\Big(E[E_x|N = \overline{N}_{Exc}(N_{Se}, \gamma_{SPS}), M = N_{sc}\overline{N}_{Sh}, S = \overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS})] - \\
&\overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS}))^2 N_r - \overline{C}_{E_x,B}(N_{Se}, \gamma_{SPS}) \Big),
\end{aligned} \tag{3.23}$$

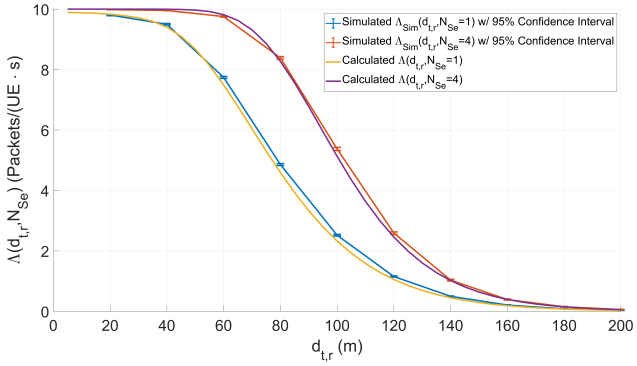
where N_r is the total number of PRBs in the sensing window and is equal to $N_{sc}(T_2 - T_1 + 1)$. Similarly $\overline{N}_{C_a}(N_{Se}, \gamma_{SPS})$ is

$$\begin{aligned}
&\overline{N}_{C_a}(N_{Se}, \gamma_{SPS}) = N_{sc}\overline{N}_{Sh} \ - E[E_x|N = \overline{N}_B(N_{Se}, \gamma_{SPS}) + \\
&\overline{N}_{Exc}(N_{Se}, \gamma_{SPS}), M = N_{sc}\overline{N}_{Sh}, S = 0].
\end{aligned} \tag{3.24}$$

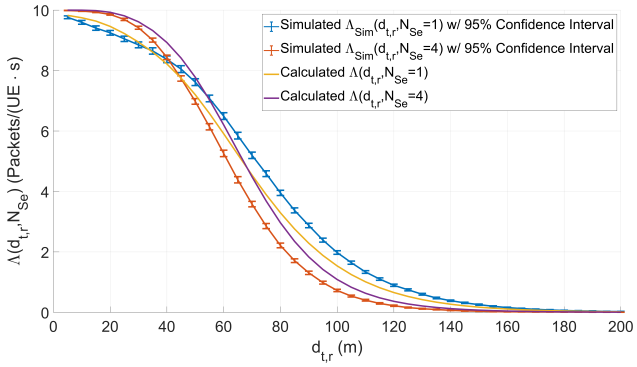
3.4 Model Insights

During C-V2X mode 2 simulation runtime, the channel parameters (N_{sc}) are assumed fixed at a pre-agreed set known to all UEs. Consequently, only the SPS parameters ($T_2/T_1,\ N_{Se}$) are candidates for online optimization. As T_2/T_1 were previously investigated in [28] and found not to significantly impact throughput. We focus instead on the impact of N_{Se} for various UE densities here. For the evaluation, we first computed throughput estimates from

our mathematical model for $\Lambda(d_{t,r}, N_{Se})$ in Matlab and thereafter used the NR sidelink implementation in ns-3 [18] to estimate $\Lambda_{Sim}(d_{t,r}, N_{Se})$ via simulation[9].



(a) $\rho_{UE} = .1$



(b) $\rho_{UE} = .4$

Figure 3.4: Effects of N_{Se} and ρ_{UE} on $\Lambda(d_{t,r}, N_{Se})$ - Calculated and Simulated, with 95% Confidence Intervals

In Figs. 3.4a and 3.4b, we show the effects of varying N_{Se} on $\Lambda(d_{t,r}, N_{Se})$ for $\rho_{UE} = .1$ and .4 respectively. In this case, the parameter under test shows a greater variation of PRR at the different UE densities. In Fig. 3.4a, the lowest density, $N_{Se} = 4$ performs best while in Fig. 3.4b, the highest density, $N_{Se} = 2$ produces the highest throughput averaged over $d_{t,r}$.

[9]A detailed description of simulation procedure follows in Section 3.7.

This corroborates with initial intuition since at higher ρ_{UE}, the resultant increase in N_{SE} causes more congestion, increasing the probability of collision. In contrast, at low ρ_{UE} where congestion is low, using higher N_{Se} introduces diversity gains. This behavior implies that by tuning N_{Se} during operation, throughput gains can be achieved relative to setting fixed N_{Se} at the beginning. However, as can be seen in Fig. 3.4b, which value of N_{Se} is optimal is not always apparent; here $N_{Se} = 4$ results in slightly higher throughput below $d_{t,r} \approx 40$, while $N_{Se} = 1$ results in slightly higher throughput elsewhere.

To determine an optimal N_{Se} for the linear network topology, we define the distance averaged throughput as

$$\overline{\Lambda}_{Sim}(N_{Se}) = \frac{1}{x} \int_{d_{t,r}=0}^{x} \Lambda_{Sim}(d_{t,r}, N_{Se}) d d_{t,r}, \tag{3.25}$$

where x is the domain of the averaging. In practice, the choice of x has no effect as long as $\Lambda_{Sim}(x, S_{Se})$ is very small for all N_{Se}. Consequently, we used $x = 160$ m. Hence the optimal values N^*_{Se} are given by

$$N^*_{Se} = \underset{N_{Se}}{\operatorname{argmax}} \overline{\Lambda}_{Sim}(N_{Se}), \tag{3.26}$$

for a known value of ρ_{UE}. Since foreknowledge of ρ_{UE} is infeasible, and poor setting of N_{Se} can lead to throughput degradation, we propose a standards-compliant algorithm to allow UEs to independently and dynamically select N_{Se} based on estimation of ρ_{UE} during simulation runtime in Section 3.5, called Dynamic Retransmission. Dynamic Retransmission renders throughput robust to the lack of foreknowledge of ρ_{UE} but is performance sub-optimal relative to pre-selecting optimal N^*_{Se} based on full knowledge. We then propose a standards non-compliant algorithm, called Opportunistic Retransmission in Section 3.6, which causes UE to independently and opportunistically select occupied PRBs based on spatial information provided by the Zone ID. This algorithm is also robust to lack of foreknowledge of ρ_{UE} but increases throughput beyond what is possible with the standards-based SPS algorithm.

3.5 Dynamic Retransmissions - A Standards Compliant Robustification Algorithm

Sections 3.5 and 3.6 describe algorithms implemented using ns-3 implementations of C-V2X [18] (available: https://5g-lena.cttc.es/) to explore the effects of the algorithms on average throughput. ns-3 throughput is measured by determining if any of the N_{Se} Physical Sidelink Control Channel (PSCCH) packet copies are correctly decoded by all other UEs through ns-3 tracing. ns-3 implementation details are covered in depth in Section 3.7, and the ns-3 code used to generate and process data is found at http://bit.ly/3SjWuAt. Both algorithms in Sections 3.5 and 3.6 are implemented in ns-3 with the help of ns3-ai [41].

3.5.1 Overview of Dynamic Retransmissions

The Dynamic Retransmission (D-Re) algorithm is designed to enhance the existing 3GPP proposed constant-N_{Se} algorithm by having UEs independently estimate ρ_{UE} during operation and select N_{Se} dynamically based on that estimate. Each UE estimates ρ_{UE} during transmission n, to obtain $\hat{\rho}_{UE}[n]$ as covered in depth in Section 3.5.2. Based on $\hat{\rho}_{UE}[n]$, each UE then determines $N^*_{Se}(\hat{\rho}_{UE}[n])$ from a precomputed table using Eq. (3.26) with other known MAC/PHY parameters (T_1/T_2, RRI, etc.). The precomputed table performs the task of computing $\overline{\Lambda}_{Sim}(N_{Se})$ for $\rho_{UE} = \hat{\rho}_{UE}[n]$ and a set of N_{Se} using Eq. (3.25), then determining N^*_{Se} using Eq. (3.26). Thereafter UE selects PRBs randomly from those remaining as conformal to the SPS algorithm. It is important to note that the D-Re algorithm will perform only slightly worse than a theoretical genie algorithm, where N^*_{Se} is always selected for every ρ_{UE}, without requiring any changes to the SPS standards. To evaluate D-Re performance, we estimate $\overline{\Lambda}_{Sim}$ (Eq. (3.25)) via ns-3 simulation.

3.5.2 Estimation of ρ_{UE}, $\hat{\rho}_{UE}[n]$

During reselection event n, UE_t will exclude $E_x[n]$ PRB from contention based on the SPS scheme. For a single reselection event, ρ_{UE} can be estimated using the probability

of excluding $E_x[n]$ UE. As such, we used an experimentally derived probability of exclusion $P_{EEx}(E_x[n], \gamma_X[n]|N_r)$[10], where $\gamma_X[n]$ is the increase in the SPS threshold during reselection event n.

$\hat{\rho}_{UE}[n]$ can then be computed by passing the single period estimate through a moving average filter with coefficient α. If N_r is known and a PHY model is assumed, then

$$\hat{\rho}_{UE}[n] = (1-\alpha)\hat{\rho}_{UE}[n-1] + \alpha \operatorname*{argmax}_{\rho} P_{EEx}(E_x[n], \gamma_X[n]|N_r). \tag{3.27}$$

Overall the D-Re algorithm is summarized in Alg. 2

Algorithm 2: Dynamic Retransmission Algorithm

Input : Sensing Information from SPS Sensing: $E_x[n]$, and $\gamma_X[n]$, and list of "empty" PRB for random selection

for *Each retransmission event n* **do**

  Estimate $\hat{\rho}_{UE}[n]$ (Eq. (3.27));

  Select $N^*_{Se}(\hat{\rho}_{UE}[n])$ PRB randomly from empty PRB and conclude;

end

3.6 Opportunistic Retransmissions - A Standards Non-Compliant Throughput Enhancing Algorithm

3.6.1 Overview of Opportunistic Retransmissions

Our final algorithm to improve C-V2X mode 2 throughput is Opportunistic Retransmissions (O-Re); it combines ρ_{UE} estimation, dynamic N_{Se} selection, and opportunistically overwriting PRB selections of other UEs in physically distant zones, akin to spatially reusing PRBs. O-Re uses the Zone ID (ZID), which is transmitted as part of the 2^{nd} stage SCI [42], to estimate distance and identify common PRBs occupied by UEs at the edge of UE_t's transmission range with low values of R_c and overwrites them to improve throughput (noticeably effective at high ρ_{UE}). The ZID partitions the network spatially into a set of discrete zones of edge length d_Z; we assume that ZID is derived based on GPS-equipped UE locations.

[10]The formation of $P_{EEx}(E_x[n], \gamma_X[n]|N_r)$ is covered in Section 3.7.1

Like the D-Re algorithm, the O-Re algorithm also estimates ρ_{UE} as $\hat{\rho}_{UE}$[11], and the *edge region* defined below, in terms of parameter d_{edge}[12]. $\hat{\rho}_{UE}$ is then fed into the settings table[13] which is a precomputed table, used to determine the numbers of overwritten PRB $N_{Se,O}$, and corresponding values of the reselection counter of the overwritten PRB N_{R_c}. Once these values are obtained, the UE uses its sensing information via the SPS algorithm to identify if a sufficient number of candidate resources exist. $N_{Se,O}$ is always at least 2, and the UE searches for an equal number of PRB on each side of the edge region. If these PRBs exist, then the UE transmits using the identified resources. Otherwise, the UE uses $\hat{\rho}_{UE}$ to determine N^*_{Se} from a lookup table and transmits N^*_{Se} duplicate packets on randomly selected empty PRBs.

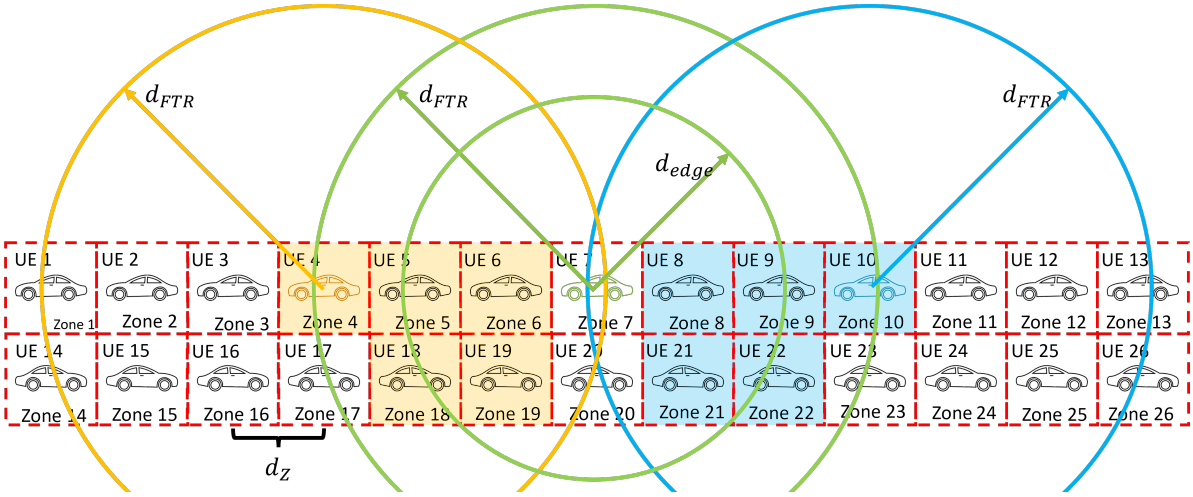


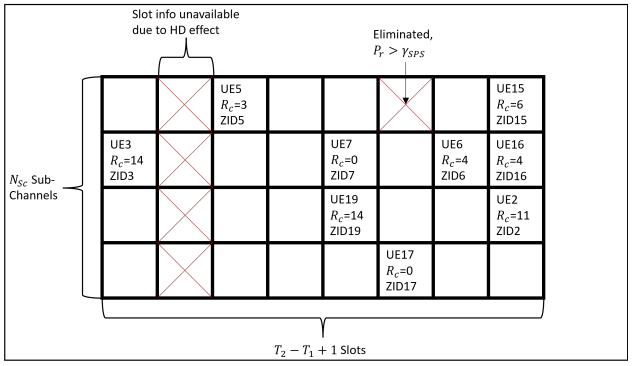
Figure 3.5: Simplified Physical Layout

Illustrative Example: Consider the UEs shown in Fig. 3.5 along with the selection windows in Fig. 3.6. Here we assume that all UEs cannot detect packets from sources further than $d_{FTR} = 3d_Z$ (measured center to center) deterministically. For instance, in Fig. 3.5, UE 7 can detect packets from UE 4 but not UE 3 or 17. Observe UEs 4, 7, and 10, shown in Fig. 3.6 in

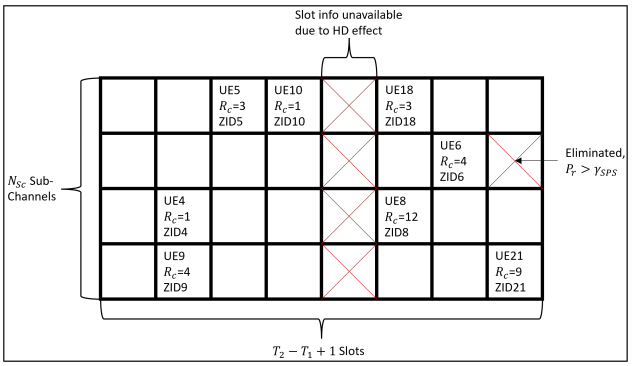
[11]Estimating ρ_{UE} is covered in depth in Section 3.5.2.

[12]Estimating d_{edge} is covered in depth in Section 3.6.2.

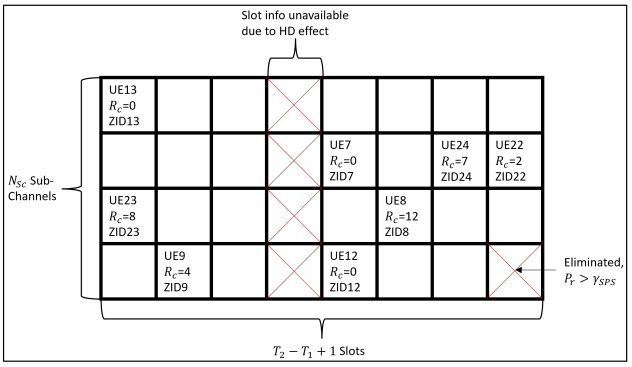
[13]How the settings table is formed is covered in Section 3.6.3.



(a) Sensing Window of UE 4 During Reselection Performed by UE 7



(b) Sensing Window of UE 7 During Reselection Performed by UE 7



(c) Sensing Window of UE 10 During Reselection Performed by UE 7

Figure 3.6: Sensing Windows for Key UEs

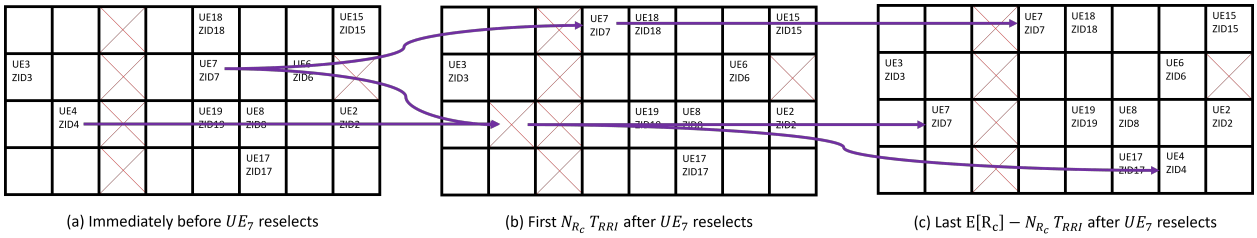


Figure 3.7: Example Effects of O-Re Over Time from the Perspective of UE 5 in Fig. 3.5

orange, green, and blue, respectively, along with respective transmission ranges of the same color. If in Fig. 3.5, UE 7 performs SPS reselection, then its sensing window at that time, along with that of UEs 4 and 10, are shown in Figs. 3.6b, 3.6a, and 3.6c respectively.

After initial SPS sensing step, UE 7 determines that $N_{Se,O} = 2$, $N_{R_c} = 1$, and $d_{edge} = 2d_Z$. After identifying available PRBs, UE 7 randomly selects N_{Se} PRBs from these available PRBs. Under our proposed O-Re selection scheme, UE 7 determines if any UE in zones 4 or 18 or in zones 10 or 22 have R_c at most equal to 1. In the example in Figs 3.5-3.6, UE 7 has decoded packets from UEs 4 (zone 4) and 10 (zone 10), and both have $R_c = 1$.

Assume UE 7 selects those PRBs; it implies that for UE 4 and all other UE in every zone that could have detected both packets (the zones in orange in Fig. 3.5, called "shared zones"), the first packet UE 7 sends in UE 4's PRB is guaranteed to collide with UE 4's packet (similar for every UE in the 'shared zones'), Fig. 3.7b. However, after the first transmission, UE 4 reselects and does not collide with any UE in the shared zones, see Fig. 3.7c[14]. A symmetrical argument holds for UE 10 (and all other UEs) in their shared zones (the zones in blue in Fig. 3.5).

Overwriting packets in PRBs creates a trade-off where the first N_{R_c} UE transmissions have reduced diversity gains compared to the standard. However, it also effectively reduces the probability of interference for some packets for subsequent transmissions while regaining the diversity gains, as shown in Fig. 3.7. In Fig. 3.7b, UE 7's 1st packet (3rd-row 2nd-column)

[14]A collision would require the interferer to be outside the FTR or for reselection to co-occur, neither of which can happen within their shared zones.

is not decoded by UE 5 due to guaranteed collisions. Meanwhile, UE 7's 2nd packet (1st-row, 4th-column) is decodable. After N_{R_c} transmissions, UE 4 reselects, and in Fig. 3.7c, UE 7's 1st packet is decoded.

Although it may seem that only transmitting the 2nd packet works just as well for UE 7, the 2nd packet has a higher likelihood of collision in Fig. 3.7c; during the transition from Fig. 3.7a to 3.7b that PRB had no occupant, the PRB used by UE 7's 1st packet did. If any UEs in the orange zones of Fig. 3.5 had performed reselection at the same time as UE 7, they could have selected the PRB occupied by UE 7's 2nd packet but not the PRB used by its 1st packet. We will show that, on average, this provides throughput gains over the entire R_c packets sent using the same PRB over just performing dynamic N_{Se} selection. The above principle holds for higher values of $N_{Se,O}$. In the case of even (odd) $N_{Se,O}$, the UE searches for $N_{Se,O}/2$ ($\lfloor N_{Se,O}/2 \rfloor$) PRB on each side of the edge region and selects one PRB randomly from the set that the SPS scheme has determined is empty.

3.6.2 Estimation of $d_{edge}[n]$

Algorithm 3: Estimation of $d_{edge}[n]$

Input : $N_{rx}[ZID]$, β, γ

Initialize: $d_{edge}[0] = 100$, $d = 0$, $N = 0$

for *Each* ZID_i *from closest to furthest* **do**

  d = dist between ZID_i and own ZID;

  $N = N + N_{rx}[ZID_i]$;

  if $N / \sum_i (N_{rx}[ZID_i]) \geq \gamma$ **then**

    d = dist between ZID_{i-1} and own ZID;

    exit;

  end

end

$d_{edge}[n] = (1 - \beta) d_{edge}[n-1] + \beta d$

During reselection event n, reference UE decodes $N_{rx}[ZID]$ unique packets from other UEs in each ZID, since the prior reselection event $n - 1$; this can be used to order the ZIDs from closest to farthest for the reference UE, producing an initial estimate for $d_{edge}[n]$

and cumulatively summing until it exceeds the threshold γ. Similar to $\hat{\rho}_{UE}[n]$, the initial estimate is passed through a moving average filter with coefficient β to compute final $d_{edge}[n]$ as summarized in Alg. 3.

3.6.3 Construction of the Settings Table

We compute the average per-UE throughput, $\Lambda_{O-Re}(d_{t,r}, N_{Se,O}, N_{R_c})$, when overwriting $N_{Se,O}$ PRB with R_c value N_{R_c} in Eq. (3.28). Because the right-hand side of Eq. (3.28) assumes that PRBs are overwritten every reselection event, it constitutes an upper bound for the performance of the O-Re scheme.

Eq. (3.28) represents the averaging (over $1/E[R_c]$ total transmissions) of the two different PRRs which occur over the R_c packet transmissions before reselection times the transmission rate $1/T_{RRI}$. The first term inside the brackets represents the PRR for the first N_{R_c} transmissions during which UE_t and the overwritten UE are guaranteed to collide (Fig. 3.7b). In this case, only half of the $N_{Se,O}$ packets sent can be decoded, the $(1 - P_{PRR}(d_{t,r}, N_{Se,O}))^{\lceil N_{Se,O}/2 \rceil}$ term. The second term then represents the remaining $E[R_c] - N_{R_c}$ transmissions after the overwritten UE gives up its resource (Fig. 3.7c). In this case, all packets sent by UE_t can be decoded, so the PRR becomes $(1-P_{PRR}(d_{t,r}, \left\lceil \frac{N_{Se,O}}{2} \right\rceil))^{\lfloor N_{Se,O}/2 \rfloor}(1-P_{PRR}(d_{t,r}, N_{Se,O}))^{\lceil N_{Se,O}/2 \rceil}$.

In general $\Lambda_{O-Re}(d_{t,r}, N_{Se,O}, N_{R_c})$ satisfies

$$\begin{aligned}\Lambda_{O-Re}(d_{t,r}, N_{Se,O}, N_{R_c}) \leq \frac{1}{T_{RRI}E[R_c]} \cdot \\ \left[N_{R_c} \left(1 - (1 - P_{PRR}(d_{t,r}, N_{Se,O}))^{\lceil N_{Se,O}/2 \rceil} \right) + (E[R_c] - N_{R_c}) \left(1 - \right. \right. \\ \left. \left. (1 - P_{PRR}(d_{t,r}, \left\lceil \frac{N_{Se,O}}{2} \right\rceil))^{\lfloor N_{Se,O}/2 \rfloor} (1 - P_{PRR}(d_{t,r}, N_{Se,O}))^{\lceil N_{Se,O}/2 \rceil} \right) \right].\end{aligned} \tag{3.28}$$

When the UE does not find $N_{Se,O}$ UEs in zones farther than d_{edge} with $R_c = N_{R_c}$, it defaults to the 3GPP reselection scheme with $N_{Se} = N_{Se}^*$ as calculated by Eq. (3.26), which implies a lower bound given by

$$\Lambda_{O-Re}(d_{t,r}, N_{Se,O}, N_{R_c}) \geq \Lambda(d_{t,r}, N^*_{Se}). \tag{3.29}$$

The upper bound of $\Lambda_{O-Re}(d_{t,r}, N_{Se,O}, N_{R_c})$ computed using Eq. (3.28) then averaged like in Eq. (3.25) to form $\overline{\Lambda}_{O-Re}(N_{Se,O}, N_{R_c})$. To form the settings table for a given value of ρ_{UE} all values of $N_{Se,O}$ and N_{R_c} which cause $\overline{\Lambda}_{O-Re}(N_{Se,O}, N_{R_c}) > \overline{\Lambda}_{Sim}(N^*_{Se})$ are ranked in descending order according to how much greater $\overline{\Lambda}_{O-Re}(N_{Se,O}, N_{R_c})$ is until random selection with $N_{Se} = N^*_{Se}$ performs best[15], terminating the table.

This results in our O-Re settings table for $\rho_{UE} = .4$ as seen in Tab. 3.1.

Parameter Combination #	$N_{Se,O}$	N_{R_c}	N_{Se}
1	4	0	-
2	2	0	-
3	4	1	-
4	2	1	-
5	3	0	-
6	4	2	-
7	2	2	-
8	3	1	-
9	-	-	2

Table 3.1: O-Re Settings Table for $\rho_{UE} = 0.4$

Overall the O-Re algorithm is summarized in Alg. 4

3.7 Simulation Methods to Cross Validate Mathematical Modeling and ns-3 Implementation of C-V2X Protocol

3.7.1 Model Validation

To explore the proposed D-Re and O-Re algorithms, we use the ns-3 NR C-V2X simulator from [18]. To ensure the simulator performed as expected, we began with tests on baseline

[15]Because our model of Λ is not perfect, we limit the settings table to include only values which are at least .1 packets/(UE $\cdot$ s) better than $\overline{\Lambda}_{Sim}(N^*_{Se})$.

Algorithm 4: Opportunistic Retransmission Algorithm

Input : Sensing Information from SPS Sensing: $E_x[n]$, $N_{rx}[ZID]$, R_c and ZID of decoded packets, and list of "empty" PRB for random selection

for *Each retransmission event n* **do**
- Estimate $\hat{\rho}_{UE}[n]$ (Eq. (3.27));
- Estimate $d_{edge}[n]$ (Alg. 3);
- **for** *Each choice i in the settings table corresponding to* $\hat{\rho}_{UE}[n]$ **do**
  - **if** $\lfloor N_{Se,O}[i]/2 \rfloor$ *PRB with* $R_c = N_{R_c}[i]$ *on each side of the edge region are decoded in the sensing window* **then**
    - **if** $N_{Se,O}[i]$ *even* **then**
      - Select identified PRB for transmission and conclude;
    - **else**
      - Select identified PRB + 1 random PRB from empty PRB for transmission and conclude;
    - **end**
  - **else if** *i is the last setting in the settings table* **then**
    - Select N^*_{Se} PRB randomly from empty PRB and conclude;
  - **end**

end

examples that validate performance in Sec. V[16]. Two enhancements to the ns-3 NR C-V2X code in [18] were implemented to conform to our PHY model assumptions: a) SNR threshold-based packet detection, as discussed in Sec. III.B and b) enabling UE in ns-3 to use the observed RSRP for PRBs during SPS reselection as per Sec. III (by editing the object containing the sensing results in ns-3 called *sensingData*). These enhancements are shown as part of a flowchart describing the simulation in Fig. 3.8 in red.

The ns-3 simulation topology mimics Fig. 3.2 by placing 150 UEs each in $\ell = 2$ lanes separated by distance $d_L = 4$ m. All parameters used for simulation can be found in Table 3.2.

For the values in Table 3.2, we found that on average, 40.36 PRBs were excluded, and no increase in γ_{SPS} was required for $\rho_{UE} = .1$ and $N_{Se} = 1$, the lowest tested packet density. Whereas for the highest packet density tested $\rho_{UE} = .4$ and $N_{Se} = 4$, an average of 93.60

[16] All code required to generate and analyze the data for both model validation and D-Re/O-Re testing can be found at http://bit.ly/3SjWuAt.

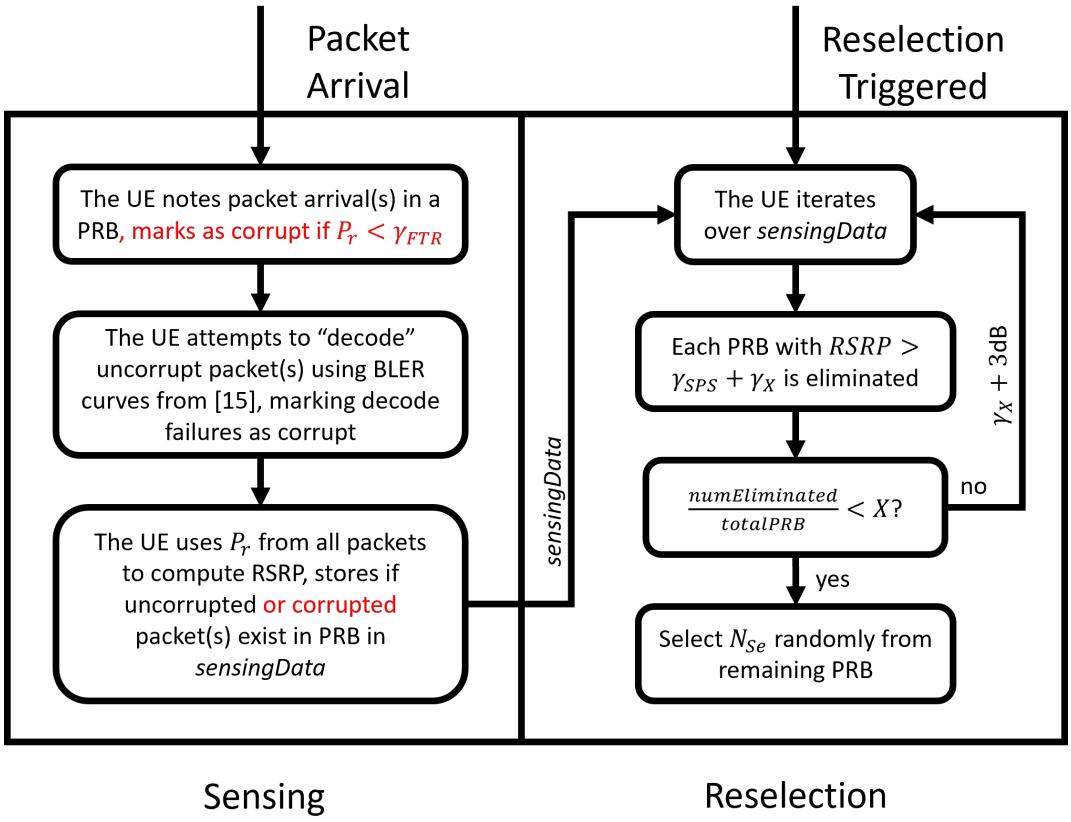


Figure 3.8: ns-3 NR C-V2X Simulator Operation with Enhancements in Red

PRBs were excluded after a required increase in γ_{SPS} of 14.41 dBm.

The ns-3 simulations produce database files that contain information on PSCCH and PSSCH transmission and reception, which are processed in Matlab to produce traces of PRR. Two sets of data need to be excised to avoid initialization and edge effects. First, even though our initialization procedure theoretically provides uniform distribution in time, the UEs do not perform any sensing, so they are more likely to collide early in the simulation. Hence, we remove the first 40 seconds (out of a total of 90 seconds) of simulation time to allow the system to achieve a steady state. Through evaluation of the traces, we found this to be more than sufficient time for the UEs to distribute themselves evenly in time and frequency. Secondly, because the UEs at the lane edge have half as many UEs to receive

Parameter	Value	Parameter	Value
T_0	.1 s	P_{Keep}	0
T_1	2 slots	P_t	15 dBm
T_2	33 slots	ℓ	2
N_{sc}	4 subchannels	d_L	4 m
μ	0	RRI	.1 s
γ_{FTR}	-110 dBm	$\gamma_{baseSPS}$	-110 dBm

Table 3.2: Constant Parameters for Simulation

packets compared to central UEs, their receptions are removed from the data set. In this case, we determined the edge to be 150 meters based on the transmit power and channel model resulting in a high probability of FTR error beyond 150 meters. To determine P_{EEx}, we created separate traces of $\gamma_X[n]$ and $E_x[n]$ during each reselection for each UE. We expunged the same subsets of data as with our PRR analysis.

Simulation completion time is primarily influenced by the number of packets sent and received and three key factors; a) the number of UEs, b) N_{Se}, and c) UE density. While the first two increase the number of packets in the system, lower densities mean greater distance between UEs and more FTR errors. Because packets that experience FTR errors are marked corrupt earlier in the simulation, they take less time to process later, as some functions can be skipped. Simulations take on the order of hours to complete, with the shortest ($d_V = 20$, $N_{se} = 1$) taking approximately 4 hours while the longest ($d_V = 5$, $N_{se} = 4$) took approximately 12 hours.

3.7.2 NS-3 Enhancements Enabling D-Re and O-Re Algorithms

Because modifying 5G NR protocol stack parameters during runtime is challenging to implement in ns-3, we used the ns3-ai application module [41]. This is an extension to the ns-3

stack that allows networking researchers to avail of open-source AI/ML libraries, typically in Python, via suitable message passing. Ns3-ai facilitates this information transfer through the use of shared memory. Hence ns-3 simulation outputs are processed via ns-3-ai, and updated parameters (in the case of D-Re and O-Re) or PRB selection (in the case of O-Re) are passed back to ns-3, and the simulation continues. The overall flow is shown in Fig. 3.9, with additions shown in red.

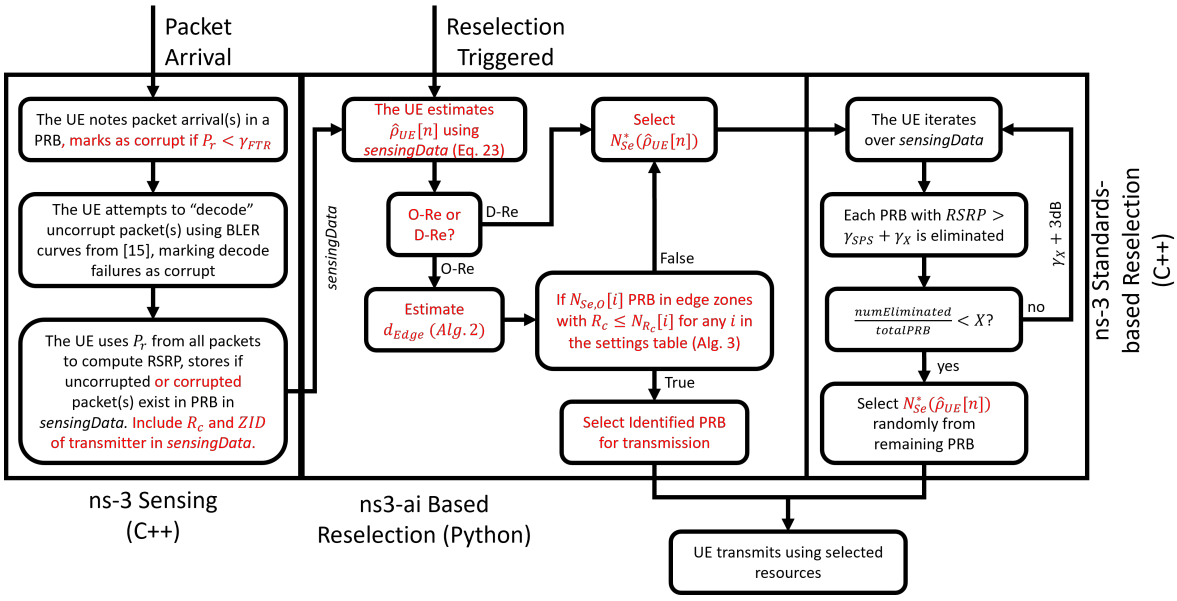


Figure 3.9: ns-3 NR C-V2X Simulator Operation with D-Re/O-Re Enhancements in Red

In addition to the ns-3 5G LENA enhancements in Sec. VIII, other necessary modifications include: 1) Added the ability for ns-3 nodes to record the R_c and ZID value of a packet in *sensingData*. 2) Made changes to the reselection code, which allowed both the changing of N_{Se} during runtime and, 3) For O-Re specifically, the ability to control the PRB directly used for transmission.

As with the standards-based algorithm, when analyzing the D-Re and O-Re data, the first 40 (of 90) seconds of data are removed to ensure a steady state, and the UEs within 150 meters from the edge are removed to remove the edge effects. Additionally, because these

UEs cannot estimate ρ_{UE} correctly, they are instead given the correct ρ_{UE} value rather than estimating. The data output format by ns-3 is identical to the data produced for model validation and is processed similarly. In addition to packet traces, we also created traces of $\hat{\rho}_{UE}[n]$ and $d_{edge}[n]$ to measure the effects of the tuning parameters α, β, and γ[17].

Because storing setting tables for every ρ_{UE} is very memory intensive, two changes were made to reduce complexity. First, rather than storing tables for discrete values of ρ_{UE}, they were stored for ranges of ρ_{UE}, as minor changes to ρ_{UE} did not result in significant changes. Secondly, some parameter sets are impossible and thus were purged. For instance, in Table 3.1, if the UE could not find two PRBs for parameter combination 2, it also cannot for parameter combination 5; thus, choices 5 and 8 were eliminated. In general, if a parameter combination in a settings table has $N_{Se,O}$ greater than a previous parameter combination but the same N_{R_c}, then it were eliminated as it is impossible.

The use of ns3-ai adds to the time needed to complete the simulation. Like with the model validation, the primary driver is the number of packets sent, which in the case of D-Re and O-Re is driven by ρ_{UE}. Both algorithms recommend using more duplicate packets for transmission at low ρ_{UE}, causing simulations to take longer. D-Re took approximately 5.5 hours to complete in the shortest case ($\rho_{UE} = .1$) to 7.5 hours in the longest case ($\rho_{UE} = .4$). Because of the greater complexity of the O-Re algorithm and the larger amount of data passed by ns3-ai, simulations took approximately 6.5 hours to complete for $\rho_{UE} = .1$ and 8 hours in the longest case ($\rho_{UE} = .4$).

[17] α controls the moving average filtering of $\hat{\rho}_{UE}$, β controls the limits of the edge region, and γ controls the moving average filtering of d_{edge}

3.8 Results and Analysis

3.8.1 Effects of Algorithm Parameters α, β, and γ

Learning Rate for $\hat{\rho}_{UE}$, α

To evaluate the effects of α on $\hat{\rho}_{UE}$, we examined $\hat{\rho}_{UE,AVG}$, the average of $\hat{\rho}_{UE}$ across all UEs in the data set as a function of time. Fig. 3.10 shows $\hat{\rho}_{UE,AVG}$ for various values of α when $\rho_{UE} = .4$. As can be expected with a 1st-order moving average, smaller values take longer to converge, while larger values may lead to oscillatory convergence. We ultimately chose to use $\alpha = .5$ as a good balance between final accuracy and convergence time.

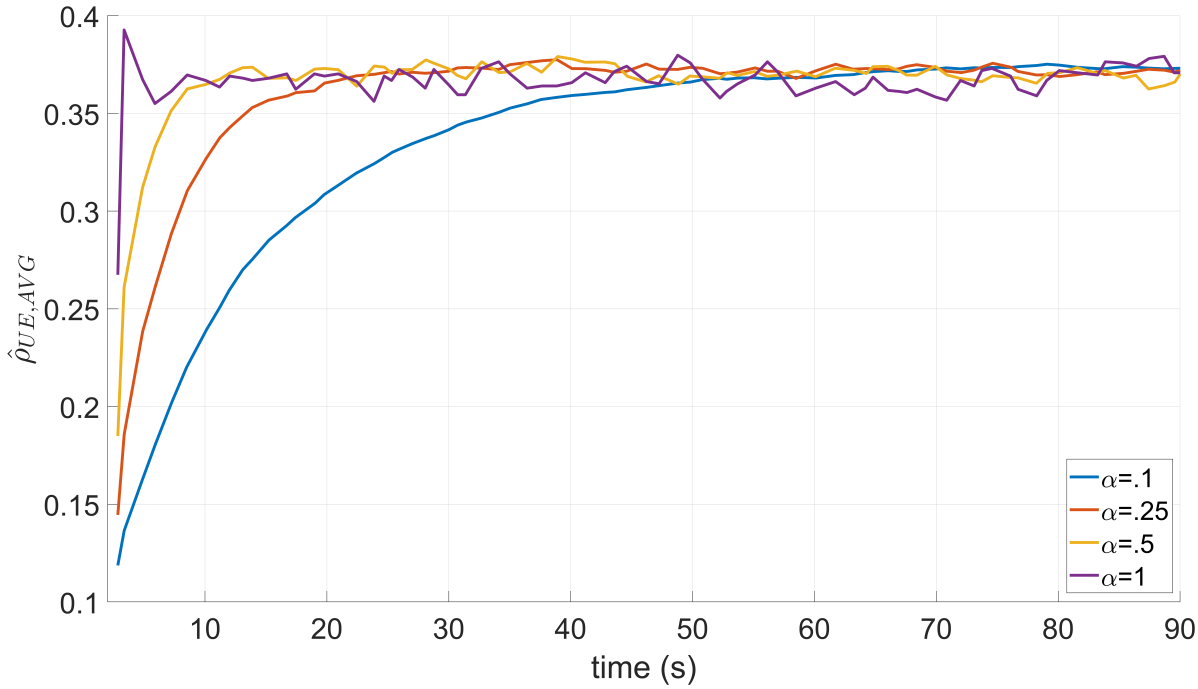


Figure 3.10: Effect of α on $\hat{\rho}_{UE,AVG}$

Learning Rate for d_{edge}, β and Threshold for d_{edge}, γ

To evaluate the effects of β and γ, we use the metric $d_{edge,AVG}$ -the average of d_{edge} across all UEs in the data set as a function of time. Fig. 3.11 show the effects of γ on $d_{edge,AVG}$ when $\rho_{UE} = .4$. Note that (averaged) d_{edge} at convergence is controlled by γ used; smaller values of γ resulting in UEs assuming the edge is closer and hence lower $d_{edge,AVG}$. β serves as a learning rate for d_{edge} as can be seen in Fig 3.12 with lower values resulting in slower convergence, while higher values produce quicker convergence but prone to oscillations as already observed.

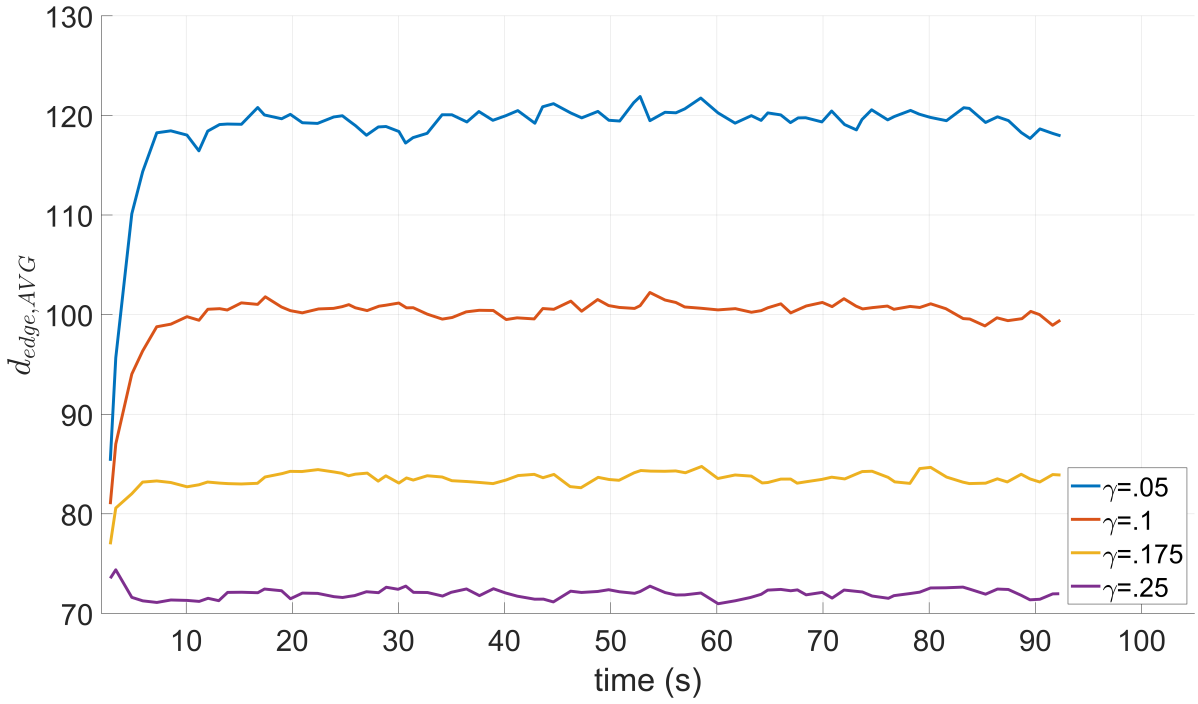


Figure 3.11: Effect of γ on $d_{edge,AVG}$, $\beta = .5$

Since a good choice of d_{edge} is very scenario dependant, it reflects on the selection of γ and β. Higher values of γ result in lower d_{edge} estimates, which allows more PRBs to be considered as potential overwrites. However, if the UEs that are reusing these PRBs are not sufficiently distant, it could cause performance degradation. Similarly, higher values of β mean that the

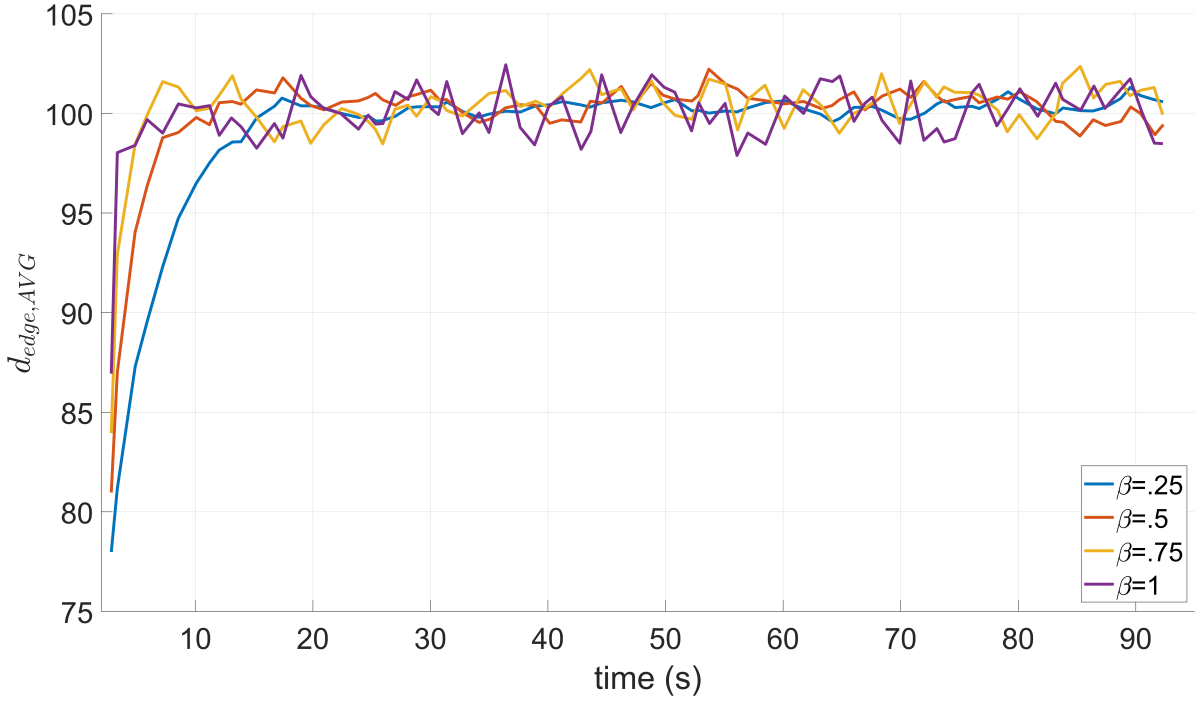


Figure 3.12: Effect of β on $d_{edge,AVG}$, $\gamma = .1$

current estimate is given more weight; thus, the set of "distant" UEs tend to be larger as the current estimate is based on the distances of the current receptions. Alternatively, selecting too small values for either parameter results in no PRB ever being overwritten; thus, O-Re effectively acts like D-Re and shows no performance gains. In summary, to determine the values of γ and β which best balance these considerations, we measured $\mu_{\Lambda,O-Re}$ for a variety of γ and β values as shown in Table 3.3.

We see that the best combination of γ and β is $\gamma = .5,\ \beta = .1$, which we used for the remainder of our experiments. We also see that throughput is not overly sensitive to the choice of γ and β with many combinations of parameters performing better than the standard (SPS) compliant algorithm, which resulted in $\overline{\Lambda}_{Sim}(N^*_{Se} = 2) = 4.74585$ packets/(UE $\cdot$ s) as a baseline.

		γ			
		.05	.1	.175	.25
β	.25	4.786	4.814	4.824	4.719
	.5	4.80	4.838	4.828	4.752
	.75	4.80	4.825	4.797	4.752
	1	4.783	4.828	4.785	4.620

Table 3.3: $\mu_{\Lambda_{O-Re}}$ for various γ and β with $d_V = 5$

3.8.2 Algorithm Performance

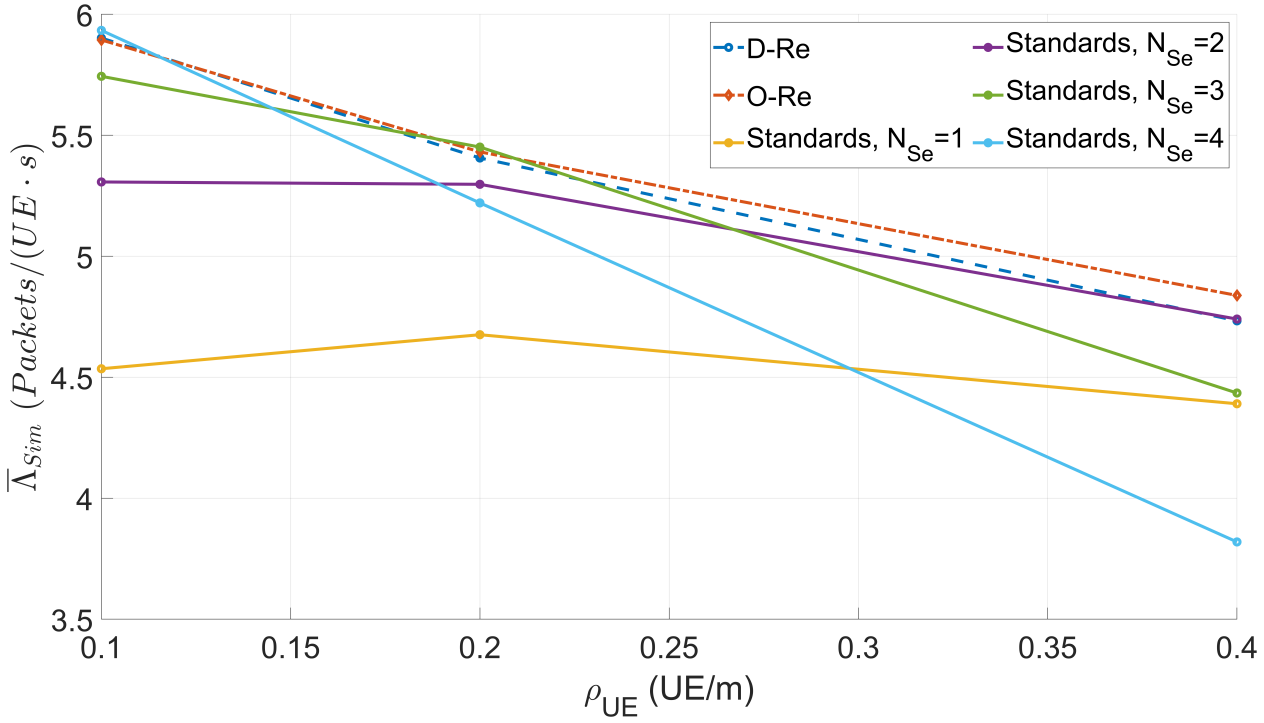


Figure 3.13: Simulated $\overline{\Lambda}_{Sim}$, $\mu_{\Lambda,D-Re}$, and $\mu_{\Lambda,O-Re}$ as a Function of ρ_{UE}

D-Re performance

As seen in Fig. 3.13, the D-Re algorithm's goal of robustification is well met. In all tested densities, the D-Re algorithm performs second best in terms of distance-averaged throughput. As expected, there exist some losses because $\hat{\rho}_{UE}$ is not a perfect estimator of ρ, causing some UE to occasionally select a sub-optimal value for N_{Se}.

Because in the standards-based algorithm, N_{Se} has to be pre-selected due to lack of foreknowledge of ρ_{UE} (or changes to ρ_{UE} over time), it can result in throughput degradation due to mismatch. For $\rho_{UE} = .1,\ .2$ or $.4$ losses can be up to 23.6%, 14.2% or 19.4% respectively. The D-Re algorithm only loses 0.53%, 0.83% or 0.16% for $\rho_{UE} = .1,\ .2$ or $.4$ respectively below the best N_{Se} and can result in improvement gain of up to 30.14%, 15.61% or 23.92% for $\rho_{UE} = .1,\ .2$ or $.4$ respectively below the worst N_{Se}.

O-Re Performance

The O-Re algorithm met its goal of robustification and throughput improvements at high ρ_{UE}. Because true network operators must select on N_{Se} and use it at all densities, O-Re is compared to an individual curve rather than a hypothetical "Genie," which selects the best N_{Se}.

If a network provider selects $N_{Se} = 4$, believing the UE density will be low, O-Re performs .67% worse at $\rho_{UE} = .1$, a small degradation in performance. However, should the provider be incorrect and the UE density increases, we see that the O-Re algorithm shows great improvements, 4.038% and 26.67% at $\rho_{UE} = .2$ and $.4$ respectively. The O-Re algorithm also shows modest improvements over the D-Re algorithm at high ρ_{UE}; for $\rho_{UE} = .4$, the O-Re algorithm performs 2.22% better. The throughput improvements increase as a function of ρ_{UE} because the O-Re algorithm attempts to exploit PRB reuse. At higher ρ_{UE}, there are more transmitters leading to a greater diversity of R_c values even though each UE transmits fewer packets optimally, so the O-Re algorithm has more opportunities to overwrite PRBs and improve performance. In contrast, at lower ρ_{UE} values such as $\rho_{UE} = .1$ and $.2$, fewer

UEs lead to a lower diversity of R_c values, and hence few PRBs are overwritten. This causes the algorithm to perform about the same as the D-Re algorithm at low ρ_{UE}.

3.9 Summary

This chapter presented a full mathematical model of NR C-V2X throughput in mode 2. We validated this model using ns-3 simulations and further explored the effects of select system parameters on the throughput performance. Resulting insights were used to develop two distributed adaptive algorithms - the standards-compliant D-Re algorithm and the standards-non-compliant O-Re algorithm. Through ns-3 simulations, we demonstrated that the D-Re algorithm that estimates UE density in simulation achieves throughput comparable to that of a 3GPP algorithm that assumes perfect knowledge of the UE density. Further, the O-Re algorithm improves the performance relative to the 3GPP SPS algorithm by making use of the zone ID (currently part of SCI but unused) and value of the reselection counter (a proposed addition to the SCI) to improve throughput for high UE densities while maintaining the robustness of the D-Re algorithm at all other UE densities.

3.10 Concluding Remarks

The key contributions of this chapter were 1) to create a standards-compliant algorithm (D-Re) that makes C-V2X communications robust to lack of foreknowledge of UE density and 2) to create a standards non-compliant algorithm (O-Re) which can exceed the throughput of the standards SPS algorithm at high UE densities. These are important because SPS algorithm parameters when set incorrectly can result in performance degradation of up to 30% and at high UE densities the the service requirements shown in Table 1.1 are the hardest to meet.

Beyond The proposed algorithms shown here, there are other techniques that we did not take advantage of and could be combined with D-Re or O-Re to further improve performance. In addition to non-orthogonal multiple access (or successive interference cancellation) and a variety of recombining methods which we did not consider in this thesis another SPS

parameter that might be jointly tuned alongside N_{Se} is the resource reservation interval T_{RRI}. T_{RRI} variation has been explored by itself in [] but could in principle be combined with blind retransmissions to boost performance beyond what either can provide separately.

It is important to note that C-V2X is a more rich environment for improvements than ProSe. This is because the sensing present in C-V2X (which is absent from ProSe discovery) makes estimating the underlying UE density much easier. The base algorithms, SPS vs discovery, also provide a wider variety of ways to tune the process and thus more opportunities for improvement. The additional inclusion of explicit positional information (in the form of zone ID) is also helpful.

Chapter 4

CONCLUSION

In this thesis, we considered two cellular protocols designed to enable communications outside the coverage of an eNb, 4G LTE ProSe, and 5G NR C-V2X. The following are the main takeaways for public safety device manufacturers and standardization bodies. Sidelink, originally introduced for ProSe, provides an invaluable tool to achieve the coverage extension needs of public safety users in off-network scenarios. Issues, however, arise because the original designs (Modes 1 and 2 in Chapter 2) had extremely high latency issues. Additionally, economic issues have made it impossible to secure the chipsets required to manufacture ProSe-enabled devices. This has pushed device manufacturers to use similar C-V2X standards (which also make use of the Sidelink) to enable coverage extensions for public safety users. ProSe features not present in C-V2X are still important for public safety, especially relaying, and should be adapted into C-V2X to support public safety use cases. Techniques needed to modulate the discovery parameters during runtime exist, given that changes in the underlying physical parameters (e.g., UE density) can have large effects on the discovery completion time. The algorithm presented in Chapter 2 of this thesis provides a good example of such an algorithm. In C-V2X, Throughput, the focus of this thesis, can potentially be met, but techniques are needed to adapt to real-world conditions[1]. The optimal MAC parameters are highly affected by the underlying physical layer, and algorithms like those presented in Chapter 3 (D-Re and O-Re) are key to meeting throughput requirements. Some additional changes ought to be made to the standards beyond those to enable O-Re; the sensing assumption that only decoded packets can be excluded during SPS reselection is damaging to throughput for no tangible benefit. The most difficult Metric to meet otherwise

[1]See Table 1.1 for C-V2X metric requirements in a variety of use cases.

is the required communications range, which is the largest range at which PRR is equal to the reliability metric. Throughput improvements benefit this at the margins as improving the average throughput has the added effect of improving the required communication range, see Fig. 3.4. Although not covered in this thesis, other potential enhancements to meet these requirements are successive interference cancellation and similar technologies, [43], or simply enabling the concept of high-powered UEs from the public safety domain.

The original contributions of this thesis include 1) evaluated the performance of 4G LTE ProSe discovery and introduced a mathematical model of discovery completion time (the time to discover all other nearby UEs); 2) demonstrated the effects of modifying UEs to FDD rather than the standard assumption of HD and demonstrated that doing so provided significant gains to discovery completion time without the need for algorithmic interventions; 3) introduced an algorithm to improve discovery by learning the underlying density of UE and tuning discovery parameters to adapt to an underlying hidden parameter; 4) evaluated the performance of 5G NR C-V2X packet transmission and developed a mathematical model for throughput and PRR; 5) created an algorithm to improve throughput in NR C-V2X beyond the standards SPS algorithm for periodic transmissions by opportunistically reselecting resources (akin to spatial reuse) through the estimation of underlying hidden parameters (UE density) and the use of currently unused system information (R_c and zone ID).

Moving forward, there are several fruitful paths for future work. First, the O-Re algorithm presented in Chapter 3 can likely be improved. One potential vector of improvement would be to explore the effects of modifying the choice of the reselection counter (how many transmissions the resource is used for). Under the semi-persistent scheduling scheme, R_c is chosen randomly because the UE does not know how good the chosen resource truly is, but in the O-Re scheme, you have some knowledge that the resource chosen is a good choice, thus should naturally want to use it longer. Another possibility would be to push the work in a more vehicular direction. Until now, although Chapter 3 uses the C-V2X standards, mobility (UE movement during the simulation) is ignored due to the more low-speed nature of pedestrian public safety UEs. If this direction were followed, mobility would be intro-

duced, the effects studied, and algorithms developed, which improved performance under those circumstances. Finally, if we continued our focus on public safety users, I believe the most relevant direction would be to explore the effects of coexistence. Throughout the thesis, we have assumed that the only users using the bandwidth devoted to sidelink are other cellular UEs. With the advent of the unlicensed band NR (NR-U), there will be a need to explore how to operate off-network C-V2X in an environment where packets do not adhere to the periodic assumptions placed on the min in this thesis. Adjacent to this thrust are considerations for dynamic transmissions (non-periodic NR C-V2X transmissions which use the sidelink) and cellular communication in contested environments (jamming, spoofing, interception, etc.), both of which share parallels to an NR-U environment.

BIBLIOGRAPHY

[1] 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*, pp. 1–1, 2021.

[2] FirstNet, "First Responder Network Authority," www.firstnet.gov.

[3] Ericson, Dan, "Improving ProSe Off-Network Coverage." Session at National institute of Standards and Technology (NIST) Public Safety Communications Research (PSCR) Division 2019 Stakeholders Meeting, July 2019.

[4] A. Kumbhar, F. Koohifar, I. Güvenç, and B. Mueller, "A survey on legacy and emerging technologies for public safety communications," *IEEE Commun. Surveys Tuts.*, vol. 19, pp. 97–124, 1st Qtr. 2017.

[5] D. Griffith and F. Lyons, "Optimizing the UE Transmission Probability for D2D Direct Discovery," in *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Dec. 2016.

[6] D. Griffith, A. B. Mosbah, and R. Rouil, "Group Discovery Time in Device-to-Device (D2D) Proximity Services (ProSe) Networks," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pp. 1–9, May 2017.

[7] M. Zorzi and R. R. Rao, "Capture and Retransmission Control in Mobile Radio," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 8, pp. 1289–1298, 1994.

[8] L. Nasraoui and S. Ikki, "Robust Neighbor Discovery Through SideLink Demodulation Reference Signal for LTE ProSe Network," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–5, 2019.

[9] Chang, Wenson and Kang, Hua, "Cyclic Block-Spreading Direct Discovery for D2D Networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7610–7620, 2020.

[10] A. B. Mosbah, D. Griffith, and R. Rouil, "Enhanced Transmission Algorithm for Dynamic Device-to-Device Direct Discovery," in *2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pp. 1–8, Jan. 2018.

[11] Gu, Xin and Peng, Jun and Cai, Lin and Cheng, Yijun and Zhang, Xiaoyong and Liu, Weirong and Huang, Zhiwu, "Performance Analysis and Optimization for Semi-Persistent Scheduling in C-V2X," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 4628–4642, 2023.

[12] M. Gonzalez-Martín, M. Sepulcre, R. Molina-Masegosa, and J. Gozalvez, "Analytical Models of the Performance of C-V2X Mode 4 Vehicular Communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1155–1166, 2019.

[13] G. P. Wijesiri N.B.A., J. Haapola, and T. Samarasinghe, "A Discrete-Time Markov Chain Based Comparison of the MAC Layer Performance of C-V2X Mode 4 and IEEE 802.11p," *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2505–2517, 2021.

[14] C. Campolo, A. Molinaro, F. Romeo, A. Bazzi, and A. O. Berthet, "5G NR V2X: On the Impact of a Flexible Numerology on the Autonomous Sidelink Mode," in *2019 IEEE 2nd 5G World Forum (5GWF)*, pp. 102–107, 2019.

[15] V. Todisco, S. Bartoletti, C. Campolo, A. Molinaro, A. O. Berthet, and A. Bazzi, "Performance Analysis of Sidelink 5G-V2X Mode 2 Through an Open-Source Simulator," *IEEE Access*, vol. 9, pp. 145648–145661, 2021.

[16] Z. Ali, S. Lagén, and L. Giupponi, "On the impact of numerology in NR V2X Mode 2 with sensing and random resource selection," in *2021 IEEE Vehicular Networking Conference (VNC)*, pp. 151–157, 2021.

[17] B. McCarthy and A. O'Driscoll, "OpenCV2X Mode 4: A Simulation Extension for Cellular Vehicular Communication Networks," in *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pp. 1–6, 2019.

[18] Z. Ali, S. Lagén, L. Giupponi, and R. Rouil, "3GPP NR V2X Mode 2: Overview, Models and System-Level Evaluation," *IEEE Access*, vol. 9, pp. 89554–89579, 2021.

[19] S. Kim, B.-J. Kim, and B. B. Park, "Environment-Adaptive Multiple Access for Distributed V2X Network: A Reinforcement Learning Framework," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, pp. 1–7, 2021.

[20] Bhadauria, Shubhangi and Vasan, Sneha and Roshdi, Moustafa and Roth-Mandutz, Elke and Fischer, Georg, "A Deep Reinforcement Learning: Location-based Resource Allocation for Congested C-V2X Scenario," in *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 0940–0946, 2021.

[21] A. Dayal, V. K. Shah, B. Choudhury, V. Marojevic, C. Dietrich, and J. H. Reed, "Adaptive Semi-Persistent Scheduling for Enhanced On-road Safety in Decentralized V2X Networks," in *2021 IFIP Networking Conference (IFIP Networking)*, pp. 1–9, 2021.

[22] W. Xing, H. He, Y. Lu, J. Yang, Y. Hu, and L. Chen, "A Novel Method on Resource Allocation for V2X," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 18–23, 2020.

[23] B. McCarthy and A. O'Driscoll, "Adapting the Resource Reservation Interval for Improved Congestion Control in NR-V2X," in *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 261–270, 2022.

[24] B. Gu, W. Chen, M. Alazab, X. Tan, and M. Guizani, "Multiagent reinforcement learning-based semi-persistent scheduling scheme in c-v2x mode 4," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 12044–12056, 2022.

[25] 3GPP, "Study on Evaluation Methodology of New Vehicle-to-Everything (V2X) Use Cases for LTE and NR," Technical Report (TR) 37.885, 3rd Generation Partnership Project (3GPP), June 2019. Version 15.3.0.

[26] C. Brady and S. Roy, "Effects of Resource Pool Size on Performance of Mode-2 LTE Device-to-Device Discovery," in *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Feb. 2020.

[27] Brady, Collin and Roy, Sumit, "A Robust Algorithm for LTE D2D (Sidelink) Discovery for Public Safety Communications," *IEEE Systems Journal*, vol. 16, no. 3, pp. 4175–4186, 2022.

[28] Brady, Collin and Cao, Liu and Roy, Sumit, "Modeling of NR C-V2X Mode 2 Throughput," in *2022 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, pp. 19–24, 2022.

[29] Brady, Collin and Roy, Sumit, "Opportunistic Retransmissions: A Novel Distributed Adaptive Algorithm for NR C-V2X Mode 2," in *IEEE Systems Journal*, 2023.

[30] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA): Radio Resource Control (RRC); Protocol Specification," Technical Specification (TS) 36.331, 3rd Generation Partnership Project (3GPP), Jan. 2019. Version 15.4.0.

[31] 3GPP TS 36.213 v16.0.0, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release 16),", 2020.

[32] S. Gamboa, F. J. Cintron, D. Griffith, and R. Rouil, "Adaptive Synchronization Reference Selection for Out-of-Coverage Proximity Services," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–7, Oct. 2017.

[33] 3GPP, "Study on LTE Device to Device Proximity Services: Radio Aspects," Technical Report (TR) 36.843, 3rd Generation Partnership Project (3GPP), Mar. 2014. Version 12.0.1.

[34] C. M. Grinstead and J. L. Snell, *Introduction to Probability*. AMS, 2003.

[35] NSNAM, "ns-3 Home Page," nsnam.org.

[36] R. Rouil, F. J. Cintrón, A. Ben Mosbah, and S. Gamboa, "Implementation and Validation of an LTE D2D Model for Ns-3," in *Proceedings of the Workshop on Ns-3*, WNS3 '17, p. 55–62, Association for Computing Machinery, 2017.

[37] CTTC, "LENA Home Page," http://networks.cttc.es/mobile-networks/software-tools/lena/.

[38] 3GPP, "NR; Multiplexing and Channel Coding," Technical Specification (TS) 38.212, 3rd Generation Partnership Project (3GPP), Jan 2022. Version 17.0.0.

[39] 3GPP, "NR; Physical Layer Procedures for Data," Technical Specification (TS) 38.214, 3rd Generation Partnership Project (3GPP), Jan 2022. Version 17.0.0.

[40] S. Lagen, K. Wanuga, H. Elkotby, S. Goyal, N. Patriciello, and L. Giupponi, "New Radio Physical Layer Abstraction for System-Level Simulations of 5G Networks," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2020.

[41] H. Yin, P. Liu, K. Liu, L. Cao, L. Zhang, Y. Gao, and X. Hei, "Ns3-Ai: Fostering Artificial Intelligence Algorithms for Networking Research," in *Proceedings of the 2020 Workshop on Ns-3*, WNS3 2020, (New York, NY, USA), p. 57–64, Association for Computing Machinery, 2020.

[42] 3GPP MCC Support, "Final Report of 3GPP TSG RAN WG1 #100-e v2.0.0," 3gpp tsg ran wg1 meeting #100bis-e, April 2020.

[43] A. Bazzi, C. Campolo, V. Todisco, S. Bartoletti, N. Decarli, A. Molinaro, A. O. Berthet, and R. A. Stirling-Gallacher, "Toward 6g vehicle-to-everything sidelink: Nonorthogonal multiple access in the autonomous mode," *IEEE Vehicular Technology Magazine*, vol. 18, no. 2, pp. 50–59, 2023.

[44] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA): Medium Access Control (MAC) Protocol Specification," Technical Specification (TS) 36.321, 3rd Generation Partnership Project (3GPP), Jan. 2019. Version 15.4.0.

[45] Hashima, Sherief and Hatano, Kohei and Takimoto, Eiji and Mahmoud Mohamed, Ehab, "Neighbor Discovery and Selection in Millimeter Wave D2D Networks Using Stochastic MAB," *IEEE Communications Letters*, vol. 24, no. 8, pp. 1840–1844, 2020.

[46] Kaleem, Zeeshan and Qadri, Nadia N. and Duong, Trung Q. and Karagiannidis, George K., "Energy-Efficient Device Discovery in D2D Cellular Networks for Public Safety Scenario," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2716–2719, 2019.

[47] A. Nabil, K. Kaur, C. Dietrich, and V. Marojevic, "Performance Analysis of Sensing-Based Semi-Persistent Scheduling in C-V2X Networks," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, 2018.

[48] X. He, J. Lv, J. Zhao, X. Hou, and T. Luo, "Design and Analysis of a Short-Term Sensing-Based Resource Selection Scheme for C-V2X Networks," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11209–11222, 2020.

[49] Y. Jeon, S. Kuk, and H. Kim, "Reducing Message Collisions in Sensing-Based Semi-Persistent Scheduling (SPS) by Using Reselection Lookaheads in Cellular V2X," *Sensors*, vol. 18, p. 4388, Dec. 2018.

[50] N. Bonjorn, F. Foukalas, and P. Pop, "Enhanced 5G V2X Services Using Sidelink Device-to-Device Communications," in *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pp. 1–7, 2018.

[51] T. Park and K. G. Shin, "SNB: Reduction of Consecutive Message Reception Failures in C-V2X Communications," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, pp. 1–6, 2020.

[52] L. F. Abanto-Leon, A. Koppelaar, and S. H. de Groot, "Enhanced C-V2X Mode-4 Subchannel Selection," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, 2018.

[53] F. Eckermann, M. Kahlert, and C. Wietfeld, "Performance Analysis of C-V2X Mode 4 Communication Introducing an Open-Source C-V2X Simulator," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pp. 1–5, 2019.

[54] 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, 2018.

[55] J. Zang and M. Shikh-Bahaei, "Full Duplex-Based Scheduling Protocol for Latency Enhancement in 5G eV2X VANETs," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2021.

[56] Y. Jeon and H. Kim, "An Explicit Reservation-Augmented Resource Allocation Scheme for C-V2X Sidelink Mode 4," *IEEE Access*, vol. 8, pp. 147241–147255, 2020.

[57] S. Sabeeh, P. Sroka, and K. Wesołowski, "Estimation and Reservation for Autonomous Resource Selection in C-V2X Mode 4," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, 2019.

[58] 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, 2021.

[59] S. Sabeeh and K. Wesołowski, "C-V2X Mode 4 Resource Allocation in High Mobility Vehicle Communication," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–6, 2020.

[60] A. Bazzi, G. Cecchini, A. Zanella, and B. M. Masini, "Study of the Impact of PHY and MAC Parameters in 3GPP C-V2V Mode 4," *IEEE Access*, vol. 6, pp. 71685–71698, 2018.

[61] G. Cecchini, A. Bazzi, B. M. Masini, and A. Zanella, "LTEV2Vsim: An LTE-V2V Simulator for the Investigation of Resource Allocation for Cooperative Awareness," in *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pp. 80–85, 2017.

[62] B. Toghi, M. Saifuddin, H. N. Mahjoub, M. O. Mughal, Y. P. Fallah, J. Rao, and S. Das, "Multiple Access in Cellular V2X: Performance Analysis in Highly Congested Vehicular Networks," in *2018 IEEE Vehicular Networking Conference (VNC)*, pp. 1–8, 2018.

[63] L. Lusvarghi and M. L. Merani, "On the Coexistence of Aperiodic and Periodic Traffic in Cellular Vehicle-to-Everything," *IEEE Access*, vol. 8, pp. 207076–207088, 2020.

[64] A. Dayal, V. K. Shah, B. Choudhury, V. Marojevic, C. Dietrich, and J. H. Reed, "Adaptive semi-persistent scheduling for enhanced on-road safety in decentralized v2x networks," in *2021 IFIP Networking Conference (IFIP Networking)*, pp. 1–9, 2021.

[65] 3GPP, "NR; Physical Channels and Modulation," Technical Specification (TS) 38.211, 3rd Generation Partnership Project (3GPP), Jan 2019. Version 17.0.0.

[66] 3GPP, "NR; Physical layer procedures for control," Technical Specification (TS) 38.212, 3rd Generation Partnership Project (3GPP), Jan 2022. Version 16.8.0.0.

[67] 3GPP, "NR; Radio Resource Control (RRC); Protocol specification," Technical Specification (TS) 38.331, 3rd Generation Partnership Project (3GPP), Dec 2021. Version 16.8.0.0.

[68] 3GPP MCC Support, "Final Report of 3GPP TSG RAN WG1 #101-e," 3gpp tsg ran wg1 meeting #102-e, August 2020.

[69] 3GPP MCC Support, ""Final Report of 3GPP TSG RAN WG1 #98bis v2.0.0," 3gpp tsg ran wg1 meeting #98bis, October 2019.

[70] F. Shakeriaski and M. Ghodrat, "Challenges and Limitation of Wearable Sensors used in Firefighters' Protective Clothing," *Journal of Fire Sciences*, 2022.

[71] M. M. Saad, M. T. R. Khan, S. H. A. Shah, and D. Kim, "Advancements in vehicular communication technologies: C-v2x and nr-v2x comparison," *IEEE Communications Magazine*, vol. 59, no. 8, pp. 107–113, 2021.

[72] M. Muhammad Saad, M. Ashar Tariq, M. Mahmudul Islam, M. Toaha Raza Khan, J. Seo, and D. Kim, "Enhanced semi-persistent scheduling (e-sps) for aperiodic traffic in nr-v2x," in *2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 171–175, 2022.

[73] M. Mikami, Y. Ishida, K. Serizawa, H. Nishiyori, K. Moto, and H. Yoshino, "Field experimental trial of dynamic mode switching for 5g nr-v2x sidelink communications towards application to truck platooning," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pp. 1–5, 2020.

[74] Y. Yoon, J. Kang, and H. Kim, "Persistent scheduling with broadcast feedback for cellular v2x communication," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, pp. 1–5, 2021.

[75] L. Cao, H. Yin, R. Wei, and L. Zhang, "Optimize Semi-Persistent Scheduling in NR-V2X: An Age-of-Information Perspective," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2053–2058, 2022.

[76] G. Shah, M. Saifuddin, Y. P. Fallah, and S. D. Gupta, "Rve-cv2x: A scalable emulation framework for real-time evaluation of cv2x-based connected vehicle applications," in *2020 IEEE Vehicular Networking Conference (VNC)*, pp. 1–8, 2020.

[77] Y. Yoon and H. Kim, "A stochastic reservation scheme for aperiodic traffic in nr v2x communication," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2021.

[78] F. Romeo, C. Campolo, A. Molinaro, and A. O. Berthet, "Asynchronous traffic on the sidelink of 5g v2x," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2020.

[79] Asano, Masashi and Fujii, Masahiro, "A packet collision reduction based on reselection for lte v2x mode 4," *Applied Sciences*, vol. 12, no. 17, 2022.

[80] Brady, Collin and Roy, Sumit, "Analysis of Mission Critical Push-to-Talk (MCPTT) Services Over Public Safety Networks," *IEEE Wireless Communications Letters*, vol. 9, no. 9, pp. 1462–1466, 2020.

[81] A. Sanchoyerto, R. Solozabal, B. Blanco, and F. Liberal, "Analysis of the Impact of the Evolution Toward 5G Architectures on Mission Critical Push-to-Talk Services," *IEEE Access*, vol. 7, pp. 115052–115061, 2019.

[82] J. Pieper et al., "Mission Critical Voice Quality of Experience Access Time Measurement Methods," Tech. Rep. 8275, NIST, 10 2019.

[83] 3GPP TS 22.179 v16.5.0, "Mission Critical Push to Talk (MCPTT); Stage 1 (Release 16),".

[84] Seattle Fire Department, Seattle, WA, *Seattle Fire Department Policies and Operating Guidelines (POG)*, 2014.

[85] M. Stojkovic, "Public safety networks towards mission critical mobile broadband networks," Master's thesis, NTNU, Trondheim, Norway, 2016.

[86] A. Merwaday and I. Guvenc, "UAV assisted heterogeneous networks for public safety communications," in *2015 IEEE Wireless Communications and Networking Conference Workshop*, pp. 329–334, March 2015.

[87] K. Gomez, T. Rasheed, L. Reynaud, and I. Bucaille, "Realistic deployments of LTE-based hybrid aerial-terrestrial networks for public safety," in *2013 IEEE 18th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pp. 233–237, Sep. 2013.

[88] J. Haigh, *Probability Models*, ch. 8, p. 192. Springer, 3 ed., 2002.

[89] A. Wald, "On cumulative sums of random variables," *Ann. Math. Statist.*, vol. 15, pp. 283–296, 09 1944.

[90] 3GPP TS 26.441 v15.0.0, "Codec for Enhanced Voice Services (EVS); General overview (Release 15),".

[91] "P25 Radio systems," Training Guide, Daniels Electronics, 2004.

[92] "Guide to Digital Radio Standards for Utilities," White Paper, Tait Communication, 2012.

[93] "TP9600 Specifications," Product Specifications, Tait Communication, 2019.

[94] M. S. Wen-Bin Yang, "LTE Physical Layer Performance Analysis," Tech. Rep. 7986, NIST, 5 2014.

[95] J. D. C. Little, "A proof for the queuing formula: L = λw," *Oper. Res.*, vol. 9, p. 383–387, June 1961.

[96] James Kurose, Keith Ross, *Computer Networking: A Top-Down Approach*, ch. 1, p. 41. Pearson, 7 ed., 2017.

[97] "800MHz Public Safety Radio System Replacement Services Project #13074C Attachment A - Appendices,", Allegan County, Michigan, 2013.

[98] H. Holma, and A. Toskala, *WCDMA for UMTS: HSPA Evolution and LTE*, ch. 12, p. 383. Wiley, 5 ed., 2010.

[99] S. Wang, W. Guo and T. O'Farrell, "Low energy indoor network: deployment optimisation," *J Wireless Com Network*, 2012.

[100] L. Kleinrock, *Queuing Systems Volume 1: Theory.* Wiley, 1 ed., 1975.

[101] S. Ross, *Introduction to Probability Models.* Academic Press, 11 ed., 2014.

[102] B. Vujičić, N. Cackov, S. Vujičić, and L. Trajković, "Modeling and characterization of traffic in public safety wireless networks," in *In Proc. of SPECTS*, pp. 214–223, 2005.

[103] S. W. Choi, Y. Song, W. Shin, and J. Kim, "A feasibility study on mission-critical push-to-talk: Standards and implementation perspectives," *IEEE Commun. Mag.*, vol. 57, no. 2, pp. 81–87, 2019.

VITA

Journal Papers:

- **C. Brady** and S. Roy, "Opportunistic Retransmissions: A Novel Distributed Adaptive Algorithm for NR C-V2X Mode 2," accepted by IEEE Systems Journal Mar 2023

- **C. Brady** and S. Roy, "A Robust Algorithm for LTE D2D (Sidelink) Discovery for Public Safety Communications," in IEEE Systems Journal, vol. 16, no. 3, pp. 4175-4186, Sept. 2022

- **C. Brady** and S. Roy, "Analysis of Mission Critical Push-to-Talk (MCPTT) Services Over Public Safety Networks," in IEEE Wireless Communications Letters, vol. 9, no. 9, pp. 1462-1466, Sept. 2020

Conference Papers:

- **C. Brady**, L. Cao and S. Roy, "Modeling of NR C-V2X Mode 2 Throughput," 2022 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR), Arlington, VA, USA, 2022

- **C. Brady** and S. Roy, "Effects of Resource Pool Size on Performance of Mode-2 LTE Device-to-Device Discovery," 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 2019

Presentations:

- **C. Brady**, L. Cao and S. Roy, "NR C-V2X Tutorial," WNS3 2023

- **C. Brady**, L. Cao and S. Roy, "Modeling of NR C-V2X Mode 2 Throughput," 2022 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR), Arlington, VA, USA, 2022

- **C. Brady**, Po-Han Huang, "Implementation of a Location/CSI Assisted ML Handover Algorithm in ns-3," WNS3, 2022

- T. Henderson, **C. Brady**, "Modeling, Simulation, & Performance Evaluation," Public Safety Broadband Stakeholder Meeting, 2018

- **C. Brady**, "Modeling and Improving Device-to-Device Direct Discovery," Public Safety Broadband Stakeholder Meeting, 2019

Industry Experience:

- Research Intern, MIT Lincoln Laboratory, Summer 2018 and 2019

Teaching Experience:

- EE 505, Probability and Random Processes, one quarter

- EE 271, Digital Circuits and Systems, one quarter

- EE 233, Circuit Theory, three quarters

- EE 215, Intro to Electrical Engineering, three quarters