The Dissertation Committee for Saadallah Kassir
certifies that this is the approved version of the following dissertation:

Modeling, Analysis, and Design of Collaborative Services in Vehicular and Cloud/Edge Networks

Committee:

Gustavo de Veciana, Supervisor

Jeffrey Andrews

Constantine Caramanis

Benjamin Leibowicz

Sanjay Shakkottai

Modeling, Analysis, and Design of Collaborative Services in Vehicular and Cloud/Edge Networks

by

Saadallah Kassir,

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2022

Dedicated to my parents Adel and Rola, siblings Sarah and Samer, and grand-parents Ismail and Ghada.

Acknowledgments

I derived an unexpected amount of enjoyment throughout my doctorate studies, and I owe debt to everyone who taught me, worked with me, and supported me over the last few years. The work delivered and presented in this thesis is the fruit of the support I have received from countless people, yet I shall attempt to recognize a few of the most influential ones.

First, I would like to express my deepest gratitude and appreciation to my Ph.D. advisor, Prof. Gustavo de Veciana, who invested a considerable amount of time and effort in me since my very first day as a graduate student. Aside from being my academic supervisor and (several times) course instructor, he has been a true mentor and role model for me, and an example to follow not only as a successful professional, but also as a bright individual. He taught me through his high work standards to constantly seek to exceed expectations so as to deliver work of the highest quality. He also made me realize, through his passion for teaching, enthusiasm towards our research projects, and our frequent lively and insightful whiteboard discussions, that one is best inspired and prepared to give the best of oneself when work is an enjoyment, and not a burden.

I would also like to thank my dissertation committee members Prof. Jeffrey Andrews, Prof. Constantine Caramanis, Prof. Benjamin Leibowicz,

and Prof. Sanjay Shakkottai for their time and valuable feedback that helped me to strengthen the results in this dissertation. I had the chance to take captivating courses with each of my committee members giving me the necessary tools to progress on my research and to build strong insight in their respective areas of expertise. Furthermore, Prof. Andrews' fundamental work in the field of Wireless Communications/Stochastic Geometry (with Prof. Baccelli), as well as his personal feedback on my research and Wireless Communication course project have been instrumental in deriving the results in the first part of this thesis. In addition, the knowledge that I have acquired through his class and our discussions was crucial in my summer internships at Qualcomm, and I owe him a lot for that. Thanks to Prof. Mark McDermott and Prof. Jonathan Valvano for giving me the opportunity to teach the Embedded Systems Lab as a Teaching Assistant in my first year in graduate school. Learning from your teaching experience and interacting with all these bright students were among the most memorable experiences in my doctorate curriculum. Thanks to Prof. Brian Evans for his guidance during my transition from undergraduate to graduate school, and for his support throughout my doctoral studies. Thanks to our research partners from Fujitsu, Dr. Paparao Palacharla, Dr. Xi Wang and Dr. Nannan Wang, for always providing us with new exciting research directions.

I would not be where I am today without the guidance of Prof. Zaher Dawy at the American University of Beirut, who has mentored me over the last two years of my undergraduate studies. He has trained and nurtured

me through multiple research projects, cultivating in me the rigor, discipline and "academic researcher's mindset" necessary to succeed in graduate school, while instilling in me his passion for wireless networking.

I also want to thank my fellow lab-mates, colleagues and friends I have had the chance to cross paths and interact with during my doctorate studies, including (but definitely not limited to) Dr. Pablo Caballero Garces, Dr. Arjun Anand, Dr. Jiaxiao Zheng, Dr. Pranav Madadi, Michael Stecklein, Jean Abou Rahal, Jianhan Song, Geetha Chandrasekaran, Hasan Beytur, Parikshit Hegde, Heasung Kim, Agrim Bari, Aniruddh Venkatakrishnan, Dr. Ribal Jurdi, Dr. Ahmad AlAmmouri, Nithin Ramesan, Manan Gupta, Isidoros Tziotis, Nihal Sharma, and Advait Parulekar, among plenty of other bright researchers. A few special mentions go to Pablo who has been of incredible help to get started with my research projects in my first year, Agrim who should become the new spiritual leader of our group after I leave, Ahmad for being such a terrible office neighbor by sparking long and pleasant discussions always brightening my day but distracting me from my work, Nithin and Manan for our frequent insightful technical discussions from which I have learnt so much, and Isidoros for being the living proof that one can be simultaneously a successful scholar and a proficient athlete.

I have also had the chance to meet and learn from very talented people in my public-speaking student organization UT Sciences Toastmasters. Aside from helping sharpen my presentation and leadership skills, the people I have met in this club have pushed me to become the best version of myself by

consistently nudging me out of my comfort zone. Among all the members, I would like to specifically thank Dr. Sarah Seraj, Minh Pham, Omid Meh, Dr. Brennan Dubuc, Arushi Guddanti, Adi Ojha, and Shubham Singh for the fun times spent together and for helping me create such a wonderful community.

Thanks to my friends Bassel Abou Ali Modad, Jad Aboul Hosn, Ahmad Chmaisse, Ahmad Ghalayini, Sami Kanafani, Stéphanie Karam, Alaa Kassir, Ghada Seifeddine, Hasan Shami and Tarek Tabaja for always checking up on me, and proving that true friendships are very little affected by long distances.

A lot of the credit goes to the incredible group of friends I have made among the Lebanese squad in Austin, composed of (by seniority) Dr. Naeem Akl, Dr. Hamza Jaffal, Dr. Mohamad Jammoul, Dr. Baker Alawieh, Dr. Mohamad Mahdi Hallal, Hussein Alawieh, Mahmoud Mbarak, and Mohamad Ali Mahaidli (visiting member). These wise Saturday night philosophers, intermittent athletes, and astute board game strategists have played a huge role keeping me sane throughout my doctorate studies, and I am grateful to belong to this fraternal group.

I cannot emphasize enough the significant role played by my roommate and long-time friend Hassan Hmedi. He has been one of the few constants I could rely on throughout the ups and downs of both our doctoral journeys. I will definitely miss our course projects/assignments partnerships, the exquisite food he cooks, and our frequent late-night whiteboard technical discussions (many of them led to breakthroughs in my research). I have always been impressed by his exceptional abilities to assimilate challenging mathematical

concepts very rapidly, and develop strong intuition that he can relate across multiple fields. I have no doubt that these skills will allow him to get very far and achieve great things in his career.

Most importantly, I am blessed to have a very supportive family that assisted me every step of the way since my youngest age. My grand-parents, Ismail and Ghada, were the first people to push me to consider leaving France and shift to the American academic system by joining the American University of Beirut. They have welcomed me to live with them throughout my undergraduate studies, and are beyond any doubt the main reason behind this successful chapter of my life. During that time, my aunts Zeina, and Hoda, as well as my uncle Hussein and his wife Fida have been of tremendous help while navigating through the new academic system and culture, and I could never thank them enough for everything they have done for me. My uncle Mahmoud and his wife Nada have also played a major role advising and supporting me once I moved to Texas, and visiting them regularly felt like deep breaths of fresh air in the midst of the COVID-19 pandemic. Finally, I would like to thank my little brother Samer for always making me smile seeing him following my footsteps, as well as my sister Sarah whose work dedication, natural selflessness and genuine graciousness are truly inspiring. Undoubtedly, none of my achievements would have been possible without the considerable sacrifices made by my parents Adel and Rola since my earliest days, but most particularly since I left home at the age of 17. I wish I could one day live up to the loving parenting standards that they have set raising our family. This achievement is as much theirs as it is mine.

Modeling, Analysis, and Design of Collaborative Services in Vehicular and Cloud/Edge Networks

Publication No. _____

Saadallah Kassir, Ph.D.
The University of Texas at Austin, 2022

Supervisor: Gustavo de Veciana

The new wireless network technologies introduced in the fifth generation of cellular networks (5G) have enabled the development of various classes of mobile applications. This thesis investigates how these emerging mobile use-cases can make the most of the state-of-the-art wireless and computing technologies through effective collaborative network management and operations strategies. We study two general classes of services: (1) collaborative traffic relaying in vehicular ad-hoc networks (VANETs), aiming at providing highly available, fair and reliable connectivity/throughput to the network users; and (2) collaborative real-time services, aimed at providing devices with low-latency and high availability/reliability connectivity.

In the first part of this thesis, we study VANETs and propose a novel vehicle connectivity framework wherein vehicles within communication range of each other form *vehicle clusters*, allowing them to opportunistically route

traffic from/to each other. With the formation of these logical entities, vehicles can be viewed as mobile relay nodes, and have the potential to substantially improve the coverage and per-user throughput of the vehicular network. In this setting, we begin by presenting an analytical framework to study the performance gains enabled by this network architecture on a single road, and we show that vehicle clustering leads to considerable benefits including reduced throughput variability and improved coverage. We then look at larger-scale cellular networks and leverage results from the stochastic geometry literature to show that the proposed opportunistic vehicle clustering and relaying scheme has the potential to improve the throughput for both vehicles and non-vehicle-bound users by more than an order of magnitude through opportunistic relaying and cell load-balancing. Finally, we study wireless resource allocation mechanisms leading to improvements in shared-rate fairness among the network users.

In the second part of this thesis, we study the operation of networks supporting real-time services, with a focus on devising efficient and timely information sharing mechanisms among the interconnected entities. We first examine how joint management of wireless communication and cloud/edge-computing resources can improve the timeliness of the information shared over the network, while reducing network resource provisioning costs. We investigate tradeoffs associated with status-update rate adaptation and service placement in the *Cloud-to-Thing continuum* for devices running real-time applications, and develop associated algorithms aiming at controlling the network

congestion and improving the service availability. We argue that sending more information might be detrimental to its quality, and that various application-specific properties influence the service placement decision in the *Cloud-to-Thing continuum*. We then examine the performance of real-time multi-user services via the specific example of Multiplayer Cloud Gaming (MCG), and exhibit how joint rate adaptation is key to controlling congestion and providing a high quality of service in spite of spatio-temporal variations in the network delays particularly impacting massive multi-user services. Finally, we give particular attention to timely information sharing in collaborative-sensing vehicular networks. We introduce a communication-efficient information-sharing mechanism enabling vehicles to benefit from each other's sensing capability in real-time via a centralized node (e.g., edge compute node, a cellular base station, or a road side unit). Our proposed mechanism opportunistically improves the vehicles' situational awareness when assistance is available, allowing them, for instance, to drive at a faster speed without compromising on safety.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Vision for the Next Generation Wireless Networks

The next generation of wireless networks, spearheaded by the release of new wireless technology standards such as 5G NR and Wi-Fi 6, promises to be a turning point for tomorrow's social interactions. While previous wireless technology generations focused on interconnecting people by providing higher throughput to end-user mobile devices such as mobile phones and tablets, current and future wireless networks are also geared towards interconnecting objects at a massive scale, while offering improved connectivity, throughput and latency performance to the connected devices. New types of services are stemming from this new connectivity framework, benefiting from (1) technological advancements at the physical layer of the network, e.g., massive-MIMO [22], beamforming [127] and improved channel coding techniques [168], (2) more efficient network resource management strategies, e.g., through network slicing [25, 59], Network Functions Virtualization (NFV) [70, 176] and Software Defined Networking (SDN) [176], and (3) additional communication and compute resources deployed in the network, e.g., licensing the millimeter-wave (mmWave) spectrum for cellular communication [132, 131, 10], cellular base-station densification [21, 12] and mobile edge computing infrastructure [111].

In the context of 5G cellular network connectivity for instance, the services to be supported have been broadly classified as (see [7, 122]):

- Enhanced Mobile Broadband (eMBB) providing (possibly highly mobile) devices with high throughput (20 Gbps downlink, 10 Gbps uplink) and high coverage to mobile devices.
- Massive Machine Type Communication (mMTC) serving densely deployed (possibly interconnected) devices ($1,000,000$ devices/km^2), while guaranteeing high service coverage and high energy efficiency at the device side.
- Ultra-Reliable Low Latency Communication (URLLC) guaranteeing low delay (sub-1ms) and highly reliable connectivity (99.999% reliability) to mission-critical devices.

By supporting such services 5G is expected to spark the emergence of a wide variety of networked-applications some of which we shall study in this thesis, including for instance Vehicle-to-Everything (V2X) based services, Extended Reality (XR) headsets, Multiplayer Cloud Gaming (MCG), and Internet-of-Things (IoT) solutions such as smart cities or smart homes.

1.2 Emerging Connectivity and Technological Trends

Driven by the opportunities made possible by 5G, some general trends are influencing the conception and commercialization of new connected devices. In this thesis, we identify, anticipate and embrace some of these trends,

from the perspective of proposing novel network operation strategies to support the associated shifting traffic demands. This section discusses some trends that are investigated in this thesis.

1.2.1 Ride-sharing Platforms

The automotive industry is undergoing several disruptive changes that are likely to have a significant impact on future wireless networks. These include the emergence and ride-sharing services and the *Transportation-as-a-Service* model, as well as the progressive adoption of autonomous driving technologies. Under both of these transportation schemes, passengers, who are no longer required to drive, are free to work/play while commuting. Hence, a considerable shift in cellular traffic patterns is to be expected, as an increasing proportion of cellular network users require connectivity from their vehicles. The resulting demand in infotainment traffic, composed, e.g., of high definition audio/video-streaming, Voice over IP (VoIP), gaming and video-conferencing services, typically requires considerable amounts of wireless resources. Therefore, efficient network operations, infrastructure deployment, and resource allocation mechanisms will need to be devised to support this shift in traffic.

1.2.2 Cloud/Edge Computing and Artificial Intelligence

Another technological trend is the emergence of the *Compute-as-a-Service* business model, allowing mobile devices to cheaply offload heavy computation tasks (such as Machine Learning and Artificial Intelligence models)

to a remote servers, leveraging the increasingly reliable, high throughput, and low-latency communication links [111]. This constitutes a new opportunity for device designers and manufacturers, as they can produce cheaper, smaller, lighter, and more energy-efficient devices by moving the device "intelligence" to a cloud or edge server [106], depending on the application performance requirements. Hence, in next-generation of wireless networks, we expect to see a convergence of the communication and compute network infrastructure and joint resource allocation schemes so as to provide improved end-to-end Quality of Service (QoS) to the network users. Consequently, new classes of performance metrics, resource management strategies and algorithms will need to be developed to satisfy the QoS expectations associated with the emerging use-cases, characterizing the joint performance of the communication and compute networks.

1.2.3 Online Multi-User Services

As connectivity is developing into a ubiquitous commodity worldwide, interactions and collaborations among geographically dispersed users /nodes is becoming increasingly prevalent. The need and ability to simultaneously collaborate on common multi-user projects was highlighted as crucial in the midst of the global COVID-19 pandemic, imposing workers from all around the world to work, interact and collaborate remotely [104]. Examples of such live/real-time multi-user services include video-conferencing, multiplayer cloud gaming, collaborative document editing, source code version control, etc. Many of these

projects/services are highly time-sensitive, and in such settings it is critical to ensure that the information generated by one node in the network is quickly and effectively disseminated to all the participants. Novel performance metrics and network operations are thus being developed to address this type of service.

1.2.4 Autonomous Driving

Autonomous driving is perhaps one of the most substantial paradigm shifts in the history of the automotive industry. Vehicles are expected to become increasingly reliant on advances in artificial intelligence, computer vision algorithms, and hence, computing technologies in general [76]. Similarly, communication technologies will play a major role in the advancement of autonomous driving technologies. Indeed, while the sensors equipped on the vehicles coupled with the on-board computation power need to be sufficient to navigate safely in their environment, additional information shared from other nodes in the network could be used by the vehicles to improve their situational awareness. As such, it is expected that effective information sharing techniques will need to be engineered to improve the safety and efficiency of the future connected vehicular networks.

1.3 Network Design Challenges and Tradeoffs

As the quality requirements of the services relying on wireless networks become increasingly demanding, designing high-performance networks

and deploying the necessary infrastructure can be particularly challenging and expensive.

Three recurring challenges and their associated design tradeoffs are often faced while engineering such networks and services.

(a) **Limited spectrum:** Wireless spectrum can be seen as a scarce natural resource [54], making it a very valuable commodity. Networks and services need therefore to be designed so as to ensure that this resource is used as efficiently as possible. However, maximizing a network's spectral efficiency might lead to undesirable effects such as unfair QoS delivered to different users, while communicating less information to reduce the spectrum utilization may have considerable impact on the provided QoS.

(b) **Information as a time-sensitive resource:** For some classes of application, information sharing in the network needs to be performed in a timely manner, and timeliness constraints can be extremely tight for some kinds of service, e.g., URLLC traffic. Indeed, for many real-time applications, the value of the communicated information quickly wanes over time. As network delays are negatively impacted by network congestion, the quantity of information shared over the network can impact its value/quality. Hence, to ensure high reliability, low latency traffic will typically need to tolerate reduced spectral-efficiency.

(c) **Network temporal dynamics and stochasticity:** Wireless networks are highly dynamic on two different levels: (1) wireless channel conditions vary

on fast time-scales; and (2) user mobility leads to continuous fluctuations in the network parameters, e.g., the number of users in the network, albeit on a slower time-scale. As these dynamics are typically captured by (possibly non-stationary) stochastic processes, it is critical to design robust algorithms that are able to adapt to such variability in the environment. While optimal algorithms/policies are typically desired, they might have poor time-complexity, making them unsuitable for real-time deployment. Optimality is therefore traded-off to allow timely computation and adaptation to time-varying parameters.

This thesis leverages multiple techniques to tackle the identified challenges and balance the corresponding tradeoffs. Among them, five general strategies, often jointly-utilized, are recurring themes in this thesis: (1) **Load balancing**, see Chapters 3 and 4, (2) **Opportunism**, see Chapters 2, 3, 4, and 6 (3) **Rate adaptation**, see Chapters 4, 5, and 6, (4) **Fairness considerations**, see Chapters 2, 3, and 5, (5) **Suboptimal algorithms**, see Chapters 3, 4, 5, and 6.

1.4 Overview of Key Insights

In this thesis, we examine collaborative networks that are subject to different combinations of the design challenges described previously. Our analysis enables us to offer network operation recommendations to network operators and service providers, while providing valuable insights on the resulting network performance. We compile below some of the major pieces of insight that

we shall develop throughout this thesis.

(a) Vehicle-to-Vehicle (V2V) cluster relaying, i.e., the ability for vehicles within communication range of each other to relay each other's traffic to/from the network infrastructure enables substantial benefits to the vehicular network. These benefits include (1) **improved reliability** in the connectivity to the infrastructure as clusters may be *multihomed*, i.e. connected to multiple infrastructure nodes simultaneously, but also since link failure, e.g., due to blockage, can be dealt with by routing traffic via a different vehicle in the cluster; (2) **reduced temporal variability** in the per-user shared-rate, and particularly, reduced fraction of time a typical vehicle is disconnected; (3) **improved mean shared-rate per vehicle**, benefiting from opportunism and load-balancing, with overall gains that can exceed an order of magnitude as compared to a non-cooperative scenario; (4) **improved mean shared-rate for non-vehicle-bound users** who benefit from load-balancing gains; (5) **improved shared-rate fairness** among network users; and (6) **improved resilience to spatial traffic surges** due to the ability to balance vehicular loads across neighboring cells.

(b) **Transmitting additional information is not always beneficial** in networks supporting real-time services, as additional transmitted packets contribute to the network congestion, leading to increased network delays that impact the timeliness/quality of the information being communicated over the network.

(c) The **service placement decision in the *Cloud-to-Thing contiunuum***

is application-specific, and controls a tradeoff between communication and compute resource provisioning costs. Services with tight timeliness constraints might be required to place service close to the devices at the network edge, while applications associated with high-computation and low-communication loads might rather benefit from statistical multiplexing effects by placing their service closer to the cloud.

(d) **The QoS of multi-user real-time services can only be as good as the QoS received by the "worst" individual user**. Therefore, massive multi-user services are particularly sensitive to spatio-temporal fluctuations in network delays, and the most advantaged users are incentivized to "assist" the most deprived ones to improve the overall service QoS.

(e) Neighboring **vehicles can considerably improve their respective situational awareness by opportunistically sharing sensing information** among each other when possible, with minimal communication overheads. This enables them, e.g., to drive at a faster velocity without compromising on the safety of their passengers and their environment.

1.5 Outline

The remaining chapters of this thesis are organized into two parts.

Part I: The first part studies efficient collaborative relaying mechanisms in Vehicular Ad-hoc Networks (VANETs). More particularly, it investigates the benefits associated with joint utilization of Vehicle-to-Infrastructure

(V2I) connectivity and Vehicle-to-Vehicle (V2V) clustering to opportunistically relay traffic from/to the network infrastructure. **Chapter 2** presents a connectivity analysis of VANETs on a single road, equipped with dedicated Road-Side Units (RSUs) to serve the vehicle-bound users. The effect of vehicles not equipped with V2V technology is also considered, as well as the benefits of multi-lane roads and driving patterns along them. **Chapter 3** considers larger-scale cellular networks and investigates how V2V clustering can lead to opportunistic relaying gains along with load-balancing gains, benefiting even non-vehicle-bound users. Wireless resource allocation algorithms are proposed and compared, ensuring shared rate fairness across the network users.

Part II: The second part investigates the notion of timely information sharing in three different categories of collaborative networks with their respective information metrics. **Chapter 4** studies Cloud/Edge/Fog networks supporting possibly heterogeneous types of real-time services. Communication and computation tradeoffs associated with service placement in the continuum between the cloud and the edge are explored, along with associated network resource provisioning and service placement algorithms. **Chapter 5** analyzes the performance and design of real-time multi-user services, such as Multiplayer Cloud Gaming (MCG). Rate-adaptation, network dimensioning and service placement strategies are discussed to ensure a high QoS in stochastic network delay environments. **Chapter 6** examines communication-efficient sensing information sharing schemes in vehicular sensing networks, showing how vehicle cooperation can improve the safety and efficiency of future au-

tonomous vehicle-based transportation systems.

Finally, **Chapter 7** concludes this thesis by compiling the major results and insights obtained through the conducted research, and provides future work suggestions to complement it.

1.6 Publications

Below is a list of conference and journal publications related to the work presented in this thesis that has been published/submitted.

1. S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, **Enhancing Cellular Performance via Vehicle-based Opportunistic Relaying and Load Balancing**. *IEEE INFOCOM 2019*, April 2019.

2. S. Kassir, P. Caballero, G. de Veciana, N. Wang, X. Wang, P. Palacharla, **An Analytical Model and Performance Evaluation of Multi-homed Multilane VANETs**. *IEEE/ACM Transactions on Networking*, February 2021.

3. S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, **Service Placement for Real-Time Applications: Rate-Adaptation and Load-Balancing at the Network Edge**. *IEEE EdgeCom 2020*, August 2020.

4. S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, **Joint Update Rate Adaptation in Multiplayer Cloud-Edge Gaming Ser-**

vices: Spatial Geometry and Performance Tradeoffs. *ACM MobiHoc 2021*, July 2021.

5. S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, **Analysis of Opportunistic Relaying and Load Balancing Gains through V2V Clustering**. *Submitted to IEEE Transactions on Vehicular Technology*, [Under Review].

6. S. Kassir, G. de Veciana, **Opportunistic Collaborative Estimation for Vehicular Networks**. *Submitted to ACM MobiHoc 2022*, [Under Review].

Part I

Collaborative Vehicle Cluster Relaying

Chapter 2

Connectivity Analysis of RSU-based Multihomed Multilane Collaborative VANETs

This chapter[1] explores the benefits of leveraging joint Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) connectivity to improve the vehicles' network connectivity and offload traffic from the traditional cellular infrastructure. In particular we consider a network architecture wherein V2V+V2I capable vehicles form relay network clusters which in turn use V2I links to connect to possibly several Road Side Units (RSUs), leveraging *multihomed* connectivity. The central goal of this chapter is to model and study the connectivity performance and tradeoffs afforded by such Vehicular Ad-Hoc Network (VANET) architectures and their ability to address the potentially substantial traffic demands placed by future intelligent transportation network and future commuters in driverless vehicles.

[1]Publications based on this chapter: [83] S. Kassir, P. Caballero, G. de Veciana, N. Wang, X. Wang, P. Palacharla, An Analytical Model and Performance Evaluation of Multihomed Multilane VANETs. IEEE/ACM Transactions on Networking, February 2021.

2.1 Related Work

There has recently been substantial interest in enabling V2V connectivity driven in part by the desire to improve safety, collaborative sensing and driving [183]. Current Dedicated Short-Range Communications (DSRC) standards for V2V relaying are mature [89, 99, 20], but in general fall short at high vehicle densities or in highly dynamic environments [27, 38, 32]. DSRC also supports V2I connectivity but only to nearby Road Side Units (RSUs) whence their placement is critical [119, 103, 171]. New alternatives based on millimeter-wave (mmWave) and Visible Light Communication (VLC) physical layers that can deliver higher capacity, e.g., 1-10 Gbps, are being currently explored [27, 38, 28]. While these provide substantial improvements in capacity, they typically require Line of Sight (LoS) based connectivity. The network architecture studied in this chapter also provides a partial solution to overcome LoS blockages through the diversity provided by multihomed multilane V2V-based vehicle clusters, making the network more robust to V2V and V2I blocking.

This chapter targets a deeper performance study of a network architecture leveraging RSUs and V2V clustering. In the past, several works have analyzed such networks, characterizing the user association expected delays [4, 136], throughput [13, 31, 84], connectivity [144, 182, 123, 94], re-healing connection time [166, 153] and percolation (full-connectivity) probability [94], among others.

This chapter builds up on models and results presented in these studies,

but includes several novel aspects that were not tackled in the mentioned papers.

2.2 Chapter Contributions and Organization

This chapter examines a model capturing the salient features of a vehicular-based wireless network, and expands previous work along several key directions. Our primary goal is to characterize the ability of such networks to deliver high capacity data rates to vehicles reliably.

First, we consider the role of V2V cluster RSU *multihoming*, i.e., the potential benefits of enabling V2V clusters to connect to multiple RSUs at the same time, in terms of improved connectivity and reliability, as well as reduced variability in users' shared rate.

Second, we provide an analytical framework to evaluate the network performance which not only accounts for the role of multihoming, but also captures the impact of V2V blockages and *market penetration* of V2V and V2I capable vehicles. We evaluate the sensitivity of a typical vehicle performance to market penetration.

Third, our evaluation of such vehicle-based networks suggests that even with a moderate penetration of V2V+V2I capable vehicles one can achieve improved connectivity and stability in per-user shared rate. For instance, *users see a reduced variability in their rate* and users in large stable clusters remain connected for large periods of time. Indeed, at high vehicle densities, one can

expect almost deterministic user perceived performance. Comparisons with simple V2I networks which do not leverage V2V relaying are used to quantify the gains of the cluster-based architecture.

Fourth, we propose a novel framework to study a typical vehicle's performance on multilane highway systems. This analysis provides key insights regarding the generalization of the single-lane results derived in the chapter, as well as the performance of various traffic patterns such as vehicle intensity heterogeneity across the highway lanes, or lanes restricted to V2V+V2I capable vehicles.

Finally, we validate the model's underlying assumptions and analyze the multilane highway performance based on additional system level simulation results of realistic traffic flows on roads, while considering real deployment considerations for the V2X technology. We then revisit the assumptions to understand how idealized control of the vehicle distribution could lead to improved performance. The analysis of this best case scenario naturally leads to the introduction of a throughput-connectivity tradeoff.

Overall, these results show that such a network could provide a reliable means to offload substantial traffic from the cellular infrastructure to vehicles, particularly when the vehicle density is high, i.e., when such assistance is most needed.

In this chapter, we focus on performing system-level modeling, analysis, and simulations. While packet-level considerations including medium-access

and control-plane management may impact the network's performance, this type of analysis has already been performed in related work, e.g., [13, 105, 145], and is deemed out of scope of this chapter.

The remaining of this chapter is organized as follows. Section 2.3 presents a single lane model for a V2V+V2I based wireless network architecture. Section 2.4 develops an analytical characterization of the statistics of typical V2V clusters, e.g., the distributions of the number of vehicles, length and number of connected RSUs as a function of system parameters including the penetration of V2V+V2I capable vehicles. Section 2.5, provides a performance analysis of the coverage probability, shared rate and service redundancy as seen by a typical vehicle, and comparisons with those achieved by a V2I network. Section 2.6 provides an extension to multilane highways to assess the impact of heterogeneity in lane traffic. We evaluate the performance of such fixed-time systems in Section 2.7. We then discuss the results' sensitivity to the vehicle placement assumption in Section 2.8, before we validate this assumption via time traces generated from a micro-mobility traffic simulator in Section 2.9. Finally, we present our conclusions in Section 2.10.

2.3 Network Model

We first consider a model for an infinite straight *single lane* road as in [94]. The model corresponds to a snapshot of a collection of vehicles along the road, whose locations follow a Poisson Point Process (PPP) Φ_v with intensity λ_v (vehicles/meter). The validity of the Poisson model has been discussed

in empirical studies such as [166, 66] showing that such a model remains appropriate in settings under the so-called free-flow traffic conditions. In our analysis, we study the network performance for a snapshot of vehicle configurations, representing the network at a fixed time. We validate the analytical results by showing how the key insight and results also apply in a dynamic setting through simulation results generated via a vehicular micro-mobility simulator, presented in Section 2.9. We model the market penetration of V2V+V2I enabled vehicles on the road as a randomly chosen fraction γ of vehicles. Thus a fraction $(1-\gamma)$ are legacy vehicles without communication capabilities which may block LoS communications among V2V capable vehicles. Furthermore, it follows that the locations of V2V capable vehicles follow a PPP with intensity $\gamma\lambda_v$, and those of legacy vehicles a PPP with rate $(1-\gamma)\lambda_v$. We will use the term *full market penetration* to denote $\gamma = 1$.

Finally, RSUs are equally spaced each λ_r^{-1} meters along the road. RSUs are wired to the Internet infrastructure to provide mapping data, infotainment and cloud computing services and may also relay messages to other clusters/vehicles. A depiction of the geometry of the network is displayed in Figure 2.1.

Connectivity: We model the vehicle connectivity based on the three assumptions listed below.

Assumption 2.3.1. *We assume a unit disk connection model for V2V and V2I links.*

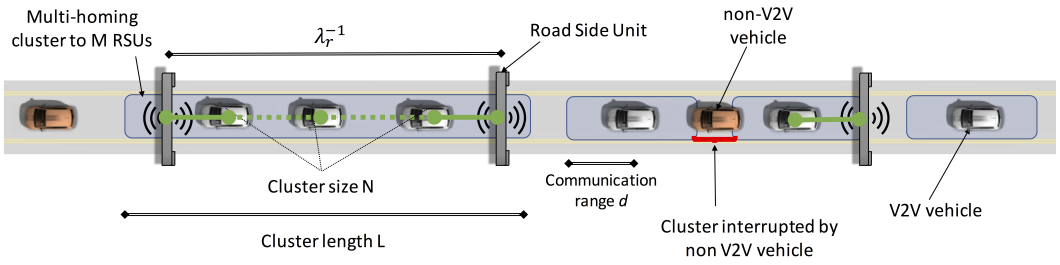


Figure 2.1: Example of the single lane highway modeled.

More specifically, a link is established if the destination vehicle is within a communication range of radius d meters of the transmitter (vehicle or RSU), as in [136, 110], and the LoS between their antennas is not obstructed, e.g., by another vehicle. We assume that the LoS between RSUs and cluster-head vehicles is never obstructed, e.g., by having RSUs above the road as illustrated in Figure 2.1.

Assumption 2.3.2. *We assume that* $d < \lambda_r^{-1}/2$.

Indeed, the communication range d would typically be on the order of 10-200 meters, while RSUs might be deployed at a distance λ_r^{-1} on the order of a few kilometers apart.

Assumption 2.3.3. *We assume V2V links to have very high capacity, exceeding the maximum RSU capacity* ρ^{RSU}.

Those links can be for instance based on mmWave or VLC technologies [159, 27, 38], while the V2I links have maximum capacity of ρ^{RSU}. Thus, for simplicity, V2V links are not a bottleneck in this system. One possible scenario

that can be envisaged is using VLC technology, known for its considerable bandwidth [28] for V2V links, while cellular (potentially mmWave based) links are used for V2I. Another scenario would be that both V2V and V2I links run on the same technology, but on orthogonal channels, and where the V2V channels can have larger bandwidth than the V2I ones. Some other multi-RAT network design considerations to avoid throughput bottlenecks are presented in [90].

The above assumptions capture the salient features of V2V+V2I networks, allowing us to explore their fundamental characteristics of possible deployments. In this chapter, we focus on analyzing the downlink performance of this network architecture.

Sharing / Scheduling: V2V capable vehicles within communication range can form V2V relaying clusters. In this chapter, we assume *RSU multihoming*, i.e., a cluster can connect to multiple RSUs in its range, as illustrated in Figure 2.1. This enables the vehicles to see (i) improved performance, i.e., connectivity and reduced variability, by sharing the capacity of multiple RSUs and (ii) improved reliability through infrastructure redundancy in the case of link failures. In this chapter, for simplicity and given Assumption 2.3.3, we will use a max-min fair resource allocation among the vehicles and clusters; where a resource allocation is said to be *max-min fair* if it is only possible to increase the resources assigned to a vehicle by decreasing the rates of vehicles which have lower rates [18]. We study the downlink shared rate seen by vehicles assuming the V2V-V2I capable vehicles are always active, i.e., full buffer traffic.

Moreover, we assume that the network does not allocate any bandwidth resources to a disconnected vehicle, i.e., that cannot reach an RSU either directly or through its cluster.

Benchmark system: We compare the described *V2V+V2I* multihoming architecture with the same *V2I* network but **without** V2V relaying, i.e., where vehicles **do not** relay data to form *V2V* relaying clusters and are only be connected to the infrastructure if they are within range of an RSU.

2.4 V2V Cluster Characterization

Definition 2.4.1 (Vehicle Relay cluster)**.** *A V2V relay cluster is a group of vehicles that can inter-communicate without the network infrastructure, i.e., each vehicle has a direct connectivity link with at least one other vehicle in its cluster. A vehicle that does not have any other vehicle within communication range is considered a cluster of size* 1*.*

A typical cluster is characterized by a (N, L, M) triplet, where N and L are random variables denoting the size (number of vehicles) and length of the cluster, respectively; and M denotes the number of RSUs that the cluster is connected to, see Figure 2.1. The performance analysis will be based on characterizing the distributions of N, L and M. The following lemmas, proved in Appendices A.2, A.3, A.4, summarize cluster statistics results.

Lemma 2.4.2 (Cluster Size Distribution)**.** *The number of vehicles N in a typical cluster follows a geometric distribution with parameter $\varphi = 1 - \gamma(1 -$*

$e^{-\lambda_v d}$), *i.e.*,

$$p_N(n) = \varphi\,(1-\varphi)^{n-1}\,, \tag{2.1}$$

and $\mathbb{E}[N] = 1/\varphi$. *Consequently, under full market penetration,* N *follows a geometric distribution with parameter* $e^{-\lambda_v d}$.

Lemma 2.4.3 (Cluster Length Distribution)**.** *The typical cluster's length* L *distribution can be obtained by the inverse Laplace transform* $\mathcal{L}^{-1}(\cdot)$ *as follows:*

$$f_L(l) = \mathcal{L}^{-1}\left(\frac{e^{-2sd}\varphi}{1 - M_T(-s) + \varphi M_T(-s)}\right)(l), \tag{2.2}$$

where

$$M_T(s) = \frac{\lambda_v e^{d(s-\lambda_v)} - \lambda_v}{(s-\lambda_v)(1-e^{-\lambda_v d})}, \tag{2.3}$$

and the conditional distribution of L *given* $N = n$ *is given by*

$$f_{L|N}(l \mid N = n) = \mathcal{L}^{-1}\left(e^{-2sd}\left[M_T(-s)\right]^{n-1}\right)(l). \tag{2.4}$$

For the case of full market penetration, the cluster length L *distribution is:*

$$f_L(l) = \mathcal{L}^{-1}\left(\frac{e^{-d(2s+\lambda_v)}\,(s+\lambda_v)}{s + \lambda_v e^{d(s-\lambda_v)}}\right)(l). \tag{2.5}$$

Lemma 2.4.4 (Number of connected RSUs)**.** *The conditional c.d.f. of the number of RSUs* M *serving a cluster of length* L *is given by*

$$F_{M|L}(m \mid L = l) = \begin{cases} 1 & \text{if } m\lambda_r^{-1} < l, \\ 1 - \frac{l}{m\cdot\lambda_r^{-1}} & \text{if } (m-1)\lambda_r^{-1} < l \le m\lambda_r^{-1}, \\ 0 & \text{otherwise}, \end{cases} \tag{2.6}$$

and the conditional CDF of the number of RSUs that serve a cluster with $N = n$ *vehicles is:*

$$F_{M|N}(m \mid N = n) = F^{c}_{L|N}((m-1)\lambda_r^{-1} \mid N = n) - \int_{(m-1)\cdot\lambda_r^{-1}}^{m\cdot\lambda_r^{-1}} \frac{l \cdot f_{L|N}(l \mid N = n)}{m \cdot \lambda_r^{-1}} dl. \tag{2.7}$$

Finally, the CDF of M *is given by:*

$$F_M(m) = \sum_{n=1}^{\infty} p_N(n) F_{M|N}(m \mid N = n), \qquad \text{for } m \in \mathbb{N}. \tag{2.8}$$

2.5 Single Lane VANET Performance Analysis

In this section, we analyze the performance of V2V+V2I multihoming networks and compare it to the V2I only network architecture.

Notation 2.5.1. *We distinguish performance metrics corresponding to the V2I networks via variables with an asterisk superscript, i.e.,* R_v *and* R_v^* *will denote the rate of a typical vehicle in the V2V+V2I and the V2I only networks respectively. Also, we will evaluate the performance seen by a typical vehicle, indicating the related metrics by a subscript* v*.*

We validate our theoretical analysis by running MATLAB simulations of the studied static single-lane vehicular network. Section 2.9 provides further simulation results of a dynamic network using traces generated via a vehicle micro-mobility simulator.

2.5.1 Typical Vehicle Coverage Probability

Definition 2.5.2 (Coverage Probability)**.** *We define the coverage probability as the probability that a typical vehicle is connected to one or more RSUs, either directly or through V2V relaying.*

One clear benefit of the V2V+V2I architecture is that it allows vehicles to relay messages from/to RSUs, increasing the coverage probability. We let π_v denote the probability that a typical vehicle is connected (possibly through relaying) to the infrastructure. Specifically, note that the typical vehicle coverage probability for the benchmark V2I network is independent of the traffic intensity. By contrast, in the V2V+V2I network, higher traffic intensities lead to longer and bigger clusters, increasing the typical vehicle coverage probability. The following result proved in Appendix A.5 addresses the coverage probability for both networks assuming $2d \leq \lambda_r^{-1}$.

Lemma 2.5.3 (Coverage probability)**.** *The coverage probability of a* ***typical vehicle*** *in the V2V+V2I network is given by:*

$$\pi_v = \varphi^2 \cdot \sum_{n=1}^{\infty} n \cdot (1-\varphi)^{n-1} \cdot F^c_{M|N}\left(0 \mid N = n\right), \tag{2.9}$$

where $F^c_{M|N}\left(0 \mid N = n\right)$ is the probability that a cluster is connected to at least one RSU given $N = n$; see Lemma 2.4.2.

The coverage probability of a typical vehicle in a V2I network is independent of λ_v and given by:

$$\pi_v^* = \frac{2d}{\lambda_r^{-1}}, \quad \textit{for} \quad 2d \leq \lambda_r^{-1}. \tag{2.10}$$

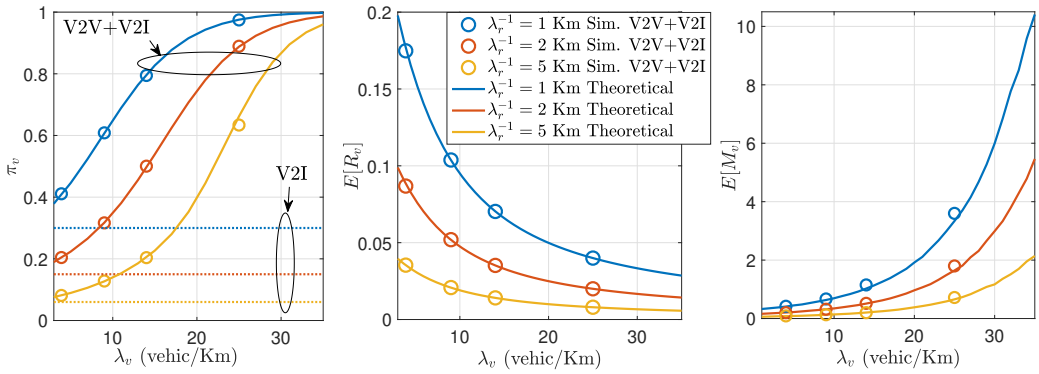


Figure 2.2: Left: Vehicle coverage probability for V2V and no V2V cases. Center: Expected RSU network throughput. Right: Expected number of RSUs connected per typical vehicle. In all cases $d = 150$ m, $\gamma = 1$. Legend applies to all plots.

Numerical evaluations of Equations 2.9 and 2.10 are displayed in Figure 2.2 (left). As expected, the coverage probability is always greater for V2V+V2I and increases rapidly to 1 with the traffic load intensity λ_v on the road. Figure 2.3 exhibits the coverage probability for V2V+V2I as a function of the penetration γ; it shows that the sensitivity of the coverage to the traffic intensity is higher at higher γ, e.g., for $\gamma = 0.9$ where the coverage probability attains a maximum for $\lambda_v \approx 25$ vehicles/km and varies notably with λ_v. Indeed, increasing λ_v increases the effect of the blocking vehicles, reaching a regime where long clusters are not possible and where π_v is independent of λ_v, consistently with Equation 2.10. Therefore, if $\gamma < 1$, π_v eventually decreases and converges back to the value presented in Equation 2.10.

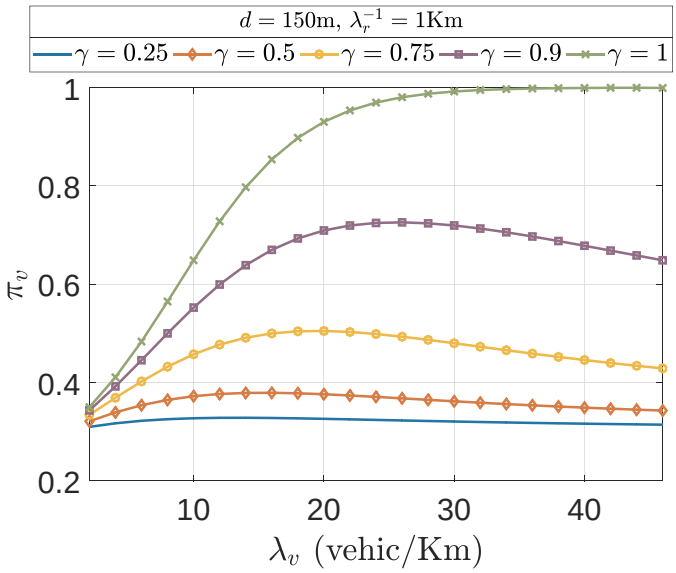


Figure 2.3: Impact of the load in the coverage probability for different market penetrations γ.

2.5.2 Typical Vehicle Shared Rate

The shared rate seen by a typical vehicle is defined as its allocations of the multihomed RSU capacity of its cluster under max-min fair sharing and denoted by the random variable R_v. The shared rate, for both networks, i.e., V2V+V2I and V2I, thus depends on $\lambda_v, \gamma, d, \rho^{\text{RSU}}$ and λ_r^{-1}, as proved in Appendix A.6.

Theorem 2.5.4 (Expected shared rate)**.** *The mean shared rate of a typical vehicle in the V2V+V2I and the V2I networks are equal, i.e.,* $\mathbb{E}[R_v] = \mathbb{E}[R_v^*]$ *and given by:*

$$\mathbb{E}[R_v] = \frac{\rho^{RSU}}{\gamma \lambda_v \lambda_r^{-1}} \left(1 - e^{-2\gamma\lambda_v d}\right) \leq \rho^{RSU} \frac{\mathbb{E}[M]}{\mathbb{E}[N]}, \tag{2.11}$$

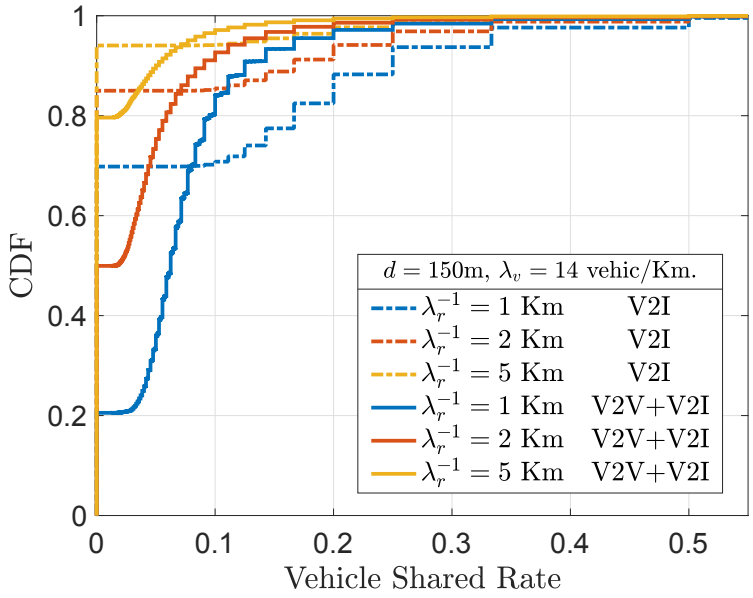


Figure 2.4: Empirical CDF of the typical shared rate for V2I vs V2V+V2I and different inter-RSU distances, for $\gamma = 1, \rho^{RSU} = 1$.

where $\mathbb{E}[N]$ and $\mathbb{E}[M]$ can be computed using Lemmas 2.4.2 and 2.4.4.

Note that the mean rate for both architectures are equal because the number of busy RSUs is the same, independently of the underlying V2V connectivity. Assuming all vehicles are infinitely backlogged the overall downlink rate is the same and thus so is the mean rate per vehicle.

Although V2V relaying collaboration does not alter the *mean* shared rate seen by vehicles, see Figure 2.2 (center); it significantly impacts the coverage probability and the shared rate *distribution*, as specified in the following theorem, proved in Appendix A.7.

Theorem 2.5.5 (Shared rate distribution)**.** *The c.d.f. of the shared rate* R_v

in a V2V+V2I network satisfies:

$$F_{R_v}(r) \geq 1 - \varphi^2 \sum_{n=1}^{\infty} n\ (1-\varphi)^{n-1} F^{c}_{M|N}\left(\left\lceil \frac{rn}{\rho^{RSU}} \right\rceil \mid n\right), \tag{2.12}$$

and $P(R_v = 0) = 1 - \pi_v$ *while that in the V2I network is given by*

$$F_{R^*_v}(r) = 1 - \frac{2d}{\lambda_r^{-1}} \cdot Q\left(\frac{\rho^{RSU}}{r} - 1,\quad 2\gamma\lambda_v d\right), \tag{2.13}$$

where Q *is the regularized gamma function and* $P(R^*_v = 0) = 1 - \pi^*_v$*. Furthermore,* $R^*_v \geq^{icx} R_v$*, where icx dominance*[2] *implies:*

$$Var(R^*_v) \geq Var(R_v). \tag{2.14}$$

Numerical evaluations of Equations 2.12 and 2.13 are shown in Figure 2.4 and the resulting variability in Figure 2.5. These demonstrate the superiority of the V2V+V2I network architecture in terms of providing, not only improved connectivity, but also a substantial decrease in the shared rate variability of a typical user. Note that in Figure 2.5 we have plotted the dispersion of the per-user shared rate, defined as σ/μ, i.e., the standard-deviation over the mean of the per user shared rate. In addition, we have displayed the lower bound on the dispersion for the non-V2V scenario, given by the dispersion as $\lambda_v \to \infty$. It can be observed that the rate dispersion converges to 0 for the V2V+V2I network. By contrast, in the V2I network the dispersion of the shared rate is bounded below. These results show that the V2V+V2I network at reasonably high vehicle density will provide them with an increasingly stable and almost deterministic shared rate to vehicles.

[2]The definition for icx dominance is provided in Appendix A.1

2.5.3 Multihoming Redundancy

RSU multihoming, i.e., the ability for a cluster to connect to multiple RSUs simultaneously via different vehicles, provides connection redundancy to a cluster. This redundancy in principle improves the reliability of vehicle connectivity in presence of unreliable/obstructed V2I links. The following result follows immediately from Equation 2.11 in Theorem 2.5.4.

Corollary 2.5.6 (Multihoming)**.** *The expected number of RSUs* $\mathbb{E}[M]$ *per cluster is bounded by:*

$$\mathbb{E}[M] \geq \frac{1 - e^{-2\gamma\lambda_v d}}{\gamma\lambda_v\lambda_r^{-1}(1 - \gamma + \gamma \cdot e^{-\gamma\lambda_v d})}, \tag{2.15}$$

which for full market penetration corresponds to

$$\mathbb{E}[M] \geq \frac{e^{\lambda_v d} - e^{-\lambda_v d}}{\lambda_v\lambda_r^{-1}} = \frac{2\sinh(\lambda_v d)}{\lambda_v\lambda_r^{-1}}. \tag{2.16}$$

As can be observed from this equation, $\mathbb{E}[M]$ i.e., the expected number of RSUs that the cluster of a typical vehicle is connected to grows rapidly with the traffic intensity λ_v and the vehicle communication range d. A similar trend is observed in Figure 2.2 (right) where we have plotted $\mathbb{E}[M_v]$, the mean number of RSUs a typical vehicle would see its cluster connected to. We see a rapid increase in the expected number of RSUs as λ_v increases. These results confirm an exponential growth of redundancy suggesting possibly substantial improvements in reliability of multihomed systems.

The effect of redundancy is also reflected in Figure 2.6 exhibiting the probability that a typical vehicle benefits from multihoming as the vehicle intensity increases. This probability reaches values very close to 1 under heavy

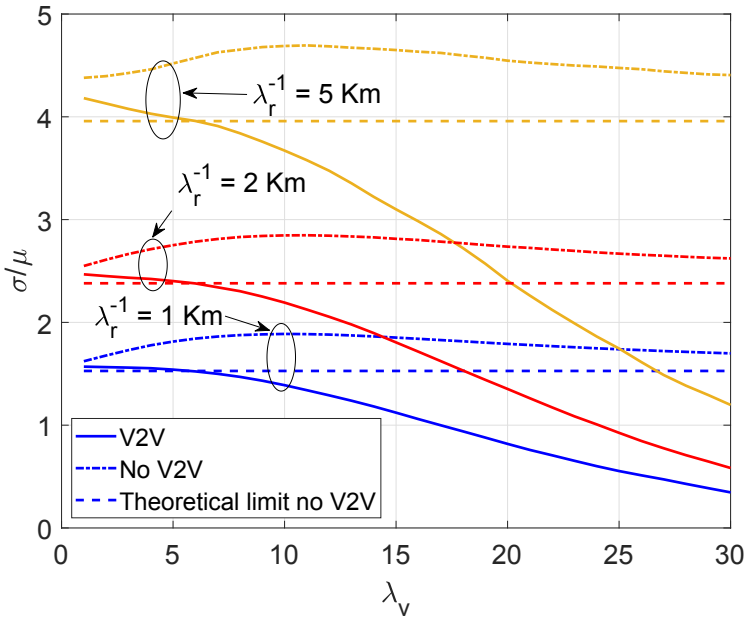


Figure 2.5: Dispersion (standard deviation over the mean) of the vehicle shared rates under V2V+V2I and V2I only scenarios, and different inter-RSU distances, for $d = 150$m, $\gamma = 1$.

traffic conditions, for the given values of λ_r^{-1}, providing evidence of the potential for higher reliability through multihoming.

2.6 Extension to Multilane Highways

The system described in Section 2.3 and analyzed in Section 2.5 considers a single lane highway. In this section we consider multilane highways. Because an exact analysis is somewhat intricate we shall explore how one can relate the performance of multilane highways to the single lane setting.

Definition 2.6.1 (Multilane highway)**.** *We define a multilane highway as a triplet:* $\left(\eta, \boldsymbol{\lambda}^{V2V}, \boldsymbol{\lambda}^{b}\right)$*, where* η *is the number of lanes, which are indexed se-*

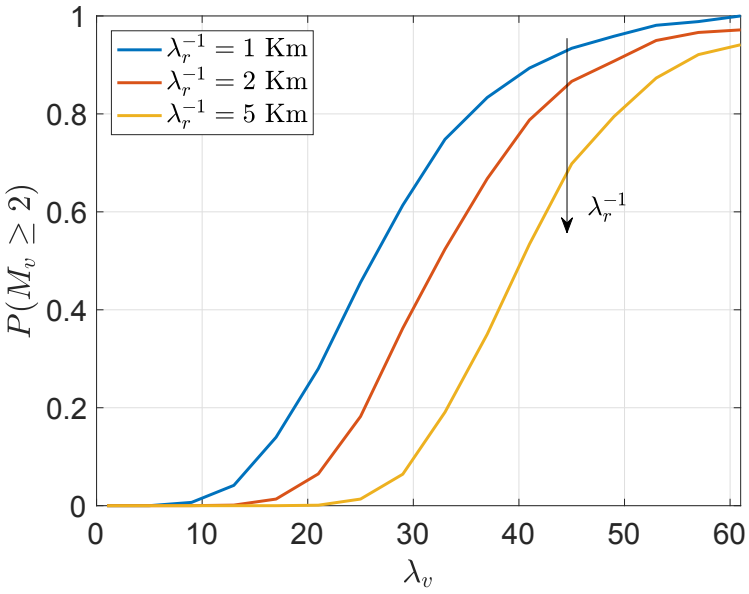


Figure 2.6: Redundancy: Probability for a typical vehicle cluster to be connected to 2 or more RSUs.

quentially $1, 2, \ldots, \eta$,

$$\boldsymbol{\lambda}^{V2V} \triangleq (\lambda_k^{V2V} : k = 1, 2, \ldots, \eta), \qquad \lambda^{V2V} \triangleq \sum_{k=1}^{\eta} \lambda_k^{V2V} \tag{2.17}$$

and

$$\boldsymbol{\lambda}^{b} \triangleq (\lambda_k^{b} : k = 1, 2, \ldots, \eta), \qquad \lambda^{b} \triangleq \sum_{k=1}^{\eta} \lambda_k^{b} \tag{2.18}$$

correspond to the intensities of V2V capable and blocking legacy vehicles in each lane, respectively. We assume each lane has independent PPPs of vehicles, and distances among lanes are negligible as compared to the communication range d.

Definition 2.6.2 (Multilane blocking model)**.** *In our multilane highway, LoS blocking is modeled as follows. Consider a triplet* $(\mathrm{k}^{-}, \mathrm{k}^{b}, \mathrm{k}^{+})$ *as the lane in-*

dex of the transmitter, of a potential blocker and the receiver, respectively. A blocker may obstruct the LoS link from k^- *to* k^+ *if and only if it is located in a lane between the transmitter and receiver, i.e.,*

$$\mathrm{k}^- = \mathrm{k}^b = \mathrm{k}^+ \quad or \quad \mathrm{k}^- < \mathrm{k}^b < \mathrm{k}^+ \quad or \quad \mathrm{k}^- > \mathrm{k}^b > \mathrm{k}^+.$$

From this definition, it follows that the worst case number of lanes where vehicles might be located and ***might*** *block a LoS link is* $k^* = \max(1, \eta - 2)$.

For a typical vehicle in a multilane highway $\mathcal{M}$, we define the number of vehicles, length and number of multihomed RSUs to its cluster as $(N_v^{\mathcal{M}}, L_v^{\mathcal{M}}, M_v^{\mathcal{M}})$ for the multi-lane highway and $(N_v^{\mathcal{S}}, L_v^{\mathcal{S}}, M_v^{\mathcal{S}})$ for a single lane road $\mathcal{S}$. We will also define $(\pi_v^{\mathcal{M}}, R_v^{\mathcal{M}})$ and $(\pi_v^{\mathcal{S}}, R_v^{\mathcal{S}})$ as the coverage probability and shared rate of a typical vehicle in multi and single lane highways. The following result, proved in Appendix A.8, compares the connectivity performance of single-lane and multilane highways.

Theorem 2.6.3. *For a given multilane highway* $\mathcal{M} = (\eta, \boldsymbol{\lambda}^{V2V}, \boldsymbol{\lambda^b})$ *let* $\mathcal{S} = (1, \gamma\lambda, \lambda_{\mathit{eff}}^b)$ *be an associated single lane highway system where:*

$$\lambda = \lambda^{V2V} + \lambda^b; \quad \gamma = \frac{\lambda^{V2V}}{\lambda} \quad and \quad \lambda_{\mathit{eff}}^b = \max(\lambda_0^b, \lambda_k^b, \sum_{i=2}^{\eta-1} \lambda_i^b).$$

Then, it follows that [3]*:*

$$N_v^{\mathcal{M}} \geq^{st} N_v^{\mathcal{S}}, \quad L_v^{\mathcal{M}} \geq^{st} L_v^{\mathcal{S}} \quad and \quad M_v^{\mathcal{M}} \geq^{st} M_v^{\mathcal{S}}$$

[3]The definitions of $\leq^{st}$ and $\leq^{icx}$ dominance can be found in Appendix A.1

and

$$\pi_v^{\mathcal{M}} \geq \pi_v^{\mathcal{S}}, \quad R_v^{\mathcal{M}} \leq^{icx} R_v^{\mathcal{S}}.$$

In other words, the multilane highway has larger cluster statistics, better coverage and decreased variability relative to the associated single lane highway.

Intuitively, this theorem indicates that taking any configuration of vehicles on a multilane highway system, and comparing it with an associated highway where all the vehicles are collapsed onto a single lane, a typical vehicle's cluster size, cluster length, number of reachable RSUs will always be larger (stochastically dominate) in the multilane configuration. Moreover, the connectivity of a typical vehicle will also be higher in this setting, while the rate variability it experiences is reduced (this follows from *icx* dominance). A high level illustration of our approach is depicted in Figure 2.7 and a sketch of the proof is provided in the appendix. The single lane performance can in turn be obtained by using the result in the previous sections.

2.7 Multilane Performance Evaluation

In this section, we further assess the performance of the proposed V2V+V2I network architecture via simulations. This will enable us to infer useful design and deployment strategies for the proposed collaborative technology in future vehicles and highways. The simulator used in this section is an extension of the one used in Section 2.5 to a multi-lane vehicular network. The communication range d is set to be 150 meters and the inter-RSU distance

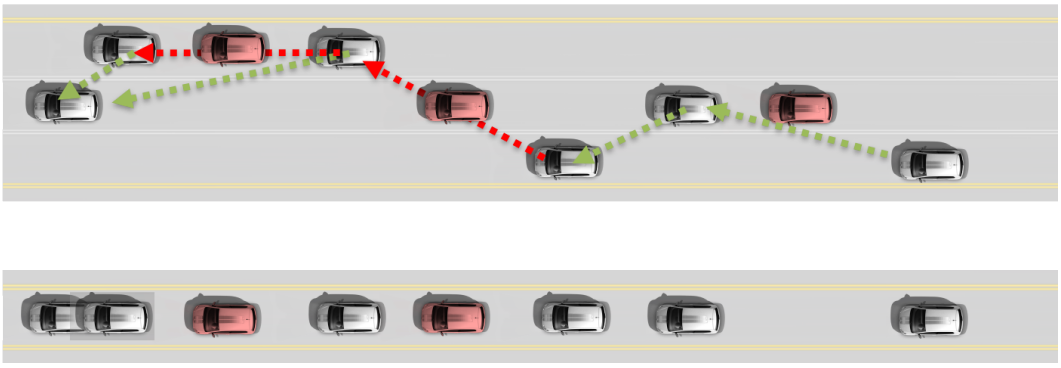


Figure 2.7: Example of the multi-lane highway approximation construction. The bottom system is the construction proposed based on the rules in Definition 2.6.2.

λ_r^{-1} is fixed at 1 km, unless otherwise specified. Table 2.1 shows typical values for different parameters that were used in the simulations. For our figures, we have obtained 95% confidence intervals achieving relative errors below 2% (not displayed). In order to capture the effect of the blocking vehicles in the multilane system, we modeled vehicles as having a length of 5 meters allowing overlapping of vehicles resulting from the Poisson assumption on their location distribution.

d		λ_v	
mmWave	VLC	Free-flow	Congestion
$75 - 200$m	≈ 100m	≤ 25 veh./km	≥ 60 veh./km

Table 2.1: Typical communication ranges and traffic densities, see [166, 118, 125]

2.7.1 Homogeneous Multilane Highways

Figure 2.8 illustrates the variation in the coverage probability π_v as η increases, but the overall traffic intensity on the highway ($\lambda_v = 20$ vehicles/Km) remains unaltered. This can be interpreted as the effect of increasing the vehicles' "degrees of freedom" to overcome blocking by legacy vehicles.

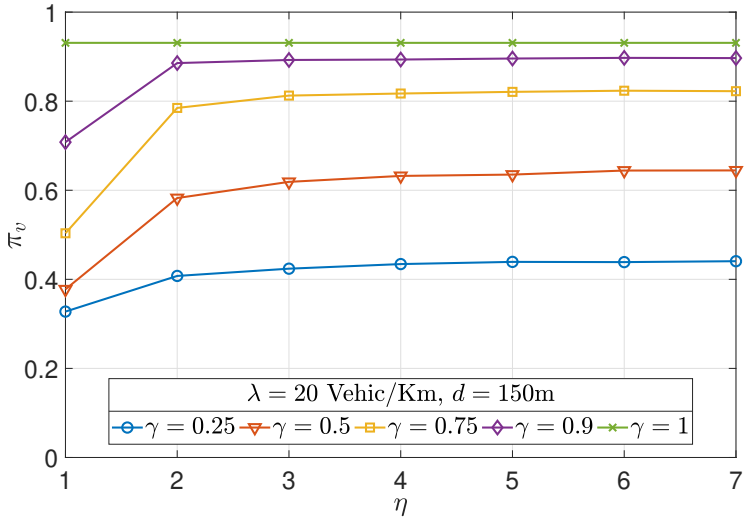


Figure 2.8: Typical vehicle's coverage probability analysis as the driving "degrees of freedom", i.e. η increases, for different γ.

A first observation is that the marginal gain in performance is most considerable when increasing the number of lanes from 1 to 2, while further increments in the number of lanes result in smaller relative gains. An explanation of this effect is that vehicles in the V2V+V2I network will see on average twice fewer blockers when passing from $\eta = 1$ to 2; while the relative decrease in the average number of blockers is smaller for higher values of η. Note that increasing the "degrees of freedom" does not affect the performance of the

system under full-market penetration as the same clusters will be formed for any value of η. From this result, one can infer that, as long as it is greater or equal than 2, the number of lanes of a highway, will not substantially affect the connectivity probability.

2.7.2 Heterogeneous Multilane Highways

Next, we further explore the impact of heterogeneous traffic intensity across lanes on the coverage probability π_v. Note that such heterogeneity is typical in highways nowadays in a free-flow regime, since for instance a greater density of slower vehicles is seen in the right hand lanes. Figure 2.9(a) exhibits the effect of the vehicle distribution on a three-lane highway. In this figure, each coordinate represents the proportion of vehicles driving on each lane, therefore all possible configurations lie on the simplex. We observe that the homogeneous configuration has the best performance as it offers the best balance between minimizing the effect of blockers on the same and across lanes. The results show that performance deteriorates slowly when moving away from the homogeneous configuration, only experiencing notable decreases when moving to extreme distributions, e.g., all users are concentrated on one lane. In order to extrapolate these results to greater values of η we define five different types of heterogeneous lane intensity distributions:

- Homogeneous: all lanes have equal vehicle intensities, e.g. for $\eta = 5$, $\boldsymbol{\lambda} = \lambda_v \eta \cdot [\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}]$.

- V: traffic is symmetrically and gradually concentrated around the left-

most and rightmost lanes of the highway, such that the intensity is minimized in the middle and maximized in the first and last lanes, e.g. for $\eta = 5$, $\boldsymbol{\lambda} = \lambda_v \eta \cdot [\frac{1}{3}, \frac{2}{15}, \frac{1}{15}, \frac{2}{15}, \frac{1}{3}]$.

- C: traffic is restricted to two lanes with identical intensities while $\eta - 2$ lanes are empty, e.g. for $\eta = 5$, $\boldsymbol{\lambda} = \lambda_v \eta \cdot [\frac{1}{2}, 0, 0, 0, \frac{1}{2}]$.

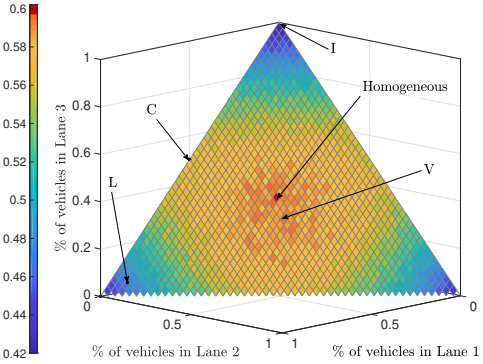
- I: traffic is restricted to one lane with $\eta - 1$ lanes empty, e.g. for $\eta = 5$, $\boldsymbol{\lambda} = \lambda_v \eta \cdot [1, 0, 0, 0, 0]$.

- L: 90% of traffic is in the first lane while the other 10% is evenly distributed across the $\eta - 1$ remaining lanes, e.g. for $\eta = 5$, $\boldsymbol{\lambda} = \lambda_v \eta \cdot [\frac{9}{10}, \frac{1}{40}, \frac{1}{40}, \frac{1}{40}, \frac{1}{40}]$.

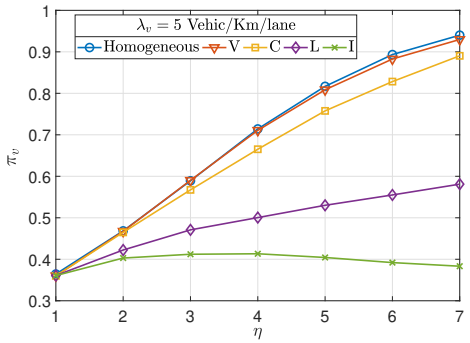
Figure 2.9(b) confirms the trends exhibited in Figure 2.9(a) as the number of lanes of the highway increases. The homogeneous distribution remains best as compared to the V, C, L and I configurations.

We note that unlike in Figure 2.8, the total number of vehicles increases with η in the highway system. An interesting insight which can be inferred from these results is the idea that congested highways (large λ_v) may have a better connectivity performance than free-flowing systems, as the intensity distribution is typically uniform across all the lanes in such cases.

Another interesting observation can be made for extreme configurations such as configuration I. We observe that as η increases, along with the density of vehicles, π_v also increases for small η as more vehicles join clusters. However,



(a) Coverage probability for $\eta = 3$ lanes.



(b) Coverage probability for different configurations.

Figure 2.9: Multilane configuration coverage probability analysis for $\gamma = 0.8, d = 150$m, $\lambda_r^{-1} = 1$ km.

for larger values of η, more non-V2V vehicles prevent the formation of large clusters, leading to a decrease in π_v.

2.7.3 V2V Segregation Impact

While manufacturers progressively release new vehicle models equipped with the V2V+V2I technology, we envision a transition period during which the roads will be shared among the new V2V-enabled and older legacy vehicles. In order to accelerate the integration and the spread of new automotive technologies, policies restricting specific lanes to driverless and V2V-enabled vehicles only might be adopted, akin to the concept of high-occupancy vehicle lane. We analyze the effect on the coverage probability of reserving the first lane for V2V-enabled vehicles and we will define α as the percentage of V2V-enabled vehicles driving on this lane, i.e. the first lane has a vehicle intensity of $\alpha\gamma\lambda_v$ with only V2V-enabled vehicles while the others are mixed

and uniformly distributed. Figure 2.10 shows the effect of α on the network performance.

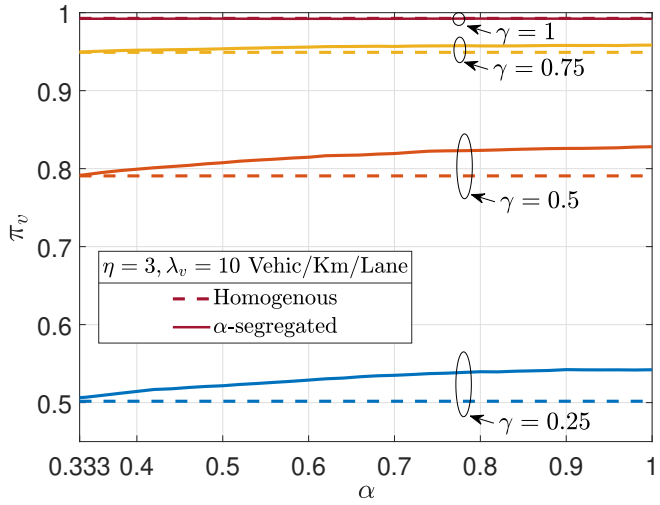


Figure 2.10: Connectivity of α-segregated scenario for different γ.

We observe that for α large enough, segregation does improve coverage, particularly for low market penetration, implying that such a policy would lead to improved connectivity in the early stages of the V2V capable vehicles deployment.

2.8 Revisiting the Poisson Assumption

In this section, we revisit one of the main assumptions of our network model, namely the Poisson distribution for cars on the highway. We study its validity through realistic multilane highway system simulations, before discussing the impact of different configurations on the network performance.

Recall that as discussed in Section 2.3, this assumption was validated in part for the free-flow setting in [166, 66].

2.8.1 Validity of the Poisson Assumption

We first explore the degree to which the PPP assumption might hold via a detailed simulation of vehicles on the road. We use traces generated from the open-source Simulation of Urban MObility (SUMO) micro-mobility simulator [64], capturing realistic traffic features, e.g., a car-following model, vehicles passing each other, vehicle dimension, and velocity control, among others.

Figure 2.11 shows the distribution for the inter-vehicle distances obtained in the simulator, on a three lanes straight highway. In light traffic, i.e., in the free-flow regime, the simulated traffic leads indeed configurations of vehicles where the inter-vehicle spacing is exponentially distributed, one of the features of a PPP. When the vehicle density increases, the spacing slightly deviates from the exponential distribution, becoming more deterministic due to congestion. Note that the results shown in Figure 2.11 correspond to inter-vehicle spacing for the projection of cars in the three lanes onto the given axis, hence although vehicles cannot be closer than their dimension permits on a given lane in the simulator, the projection of the vehicles' centers on the three lanes can be arbitrarily close.

Therefore, as long as the vehicles are operating in the free-flow regime, we expect that the observations and conclusions drawn from Figures 2.2-2.6

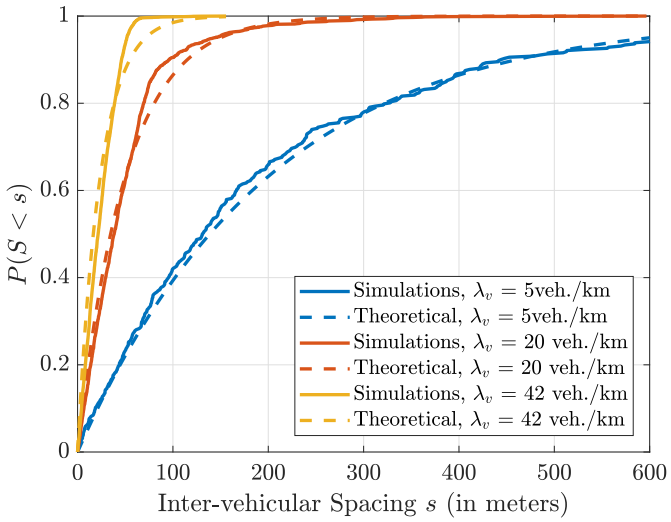


Figure 2.11: Comparison of the simulated inter-vehicle spacing c.d.f. with the corresponding exponential random variables, on a collapsed 3 lanes highway system, $\eta = 3$.

in the single lane scenario to apply in the multilane configuration as well. Moreover, our analysis in Sections 2.6 and 2.7 predicts improved performances compared to the single lane case. For instance, we expect a higher probability of connectivity, better redundancy, or improved per-user shared rate for instance, due to the fact that clusters can be larger in size and that blocking vehicles have a less severe impact on the others.

2.8.2 Study of Non-Poisson Traffic Scenarios

As shown in Figure 2.11, the Poisson assumption may not hold for all traffic patterns, e.g., when vehicles are not moving in the free-flow regime. We now consider two specific scenarios where the Poisson assumption may not

be applicable: (1) a high vehicle density regime, and (2) a road with traffic lights deployed every 1 km with random phases, turning from red to green and green to red every 60 seconds. In both settings, vehicles are not free to move at their desired speed and vehicles' locations may be correlated, either due to congestion or to the traffic lights. We are interested in evaluating how the connectivity probability is impacted by this correlation in vehicle locations, using SUMO traces. Figure 2.12 shows how the connectivity π_v of a typical vehicle varies as a function of γ in these two scenarios. For comparison purposes, we also exhibit the network performance for the case where vehicles are placed according to a PPP on the road. As a fair comparison, the three scenarios were simulated with the same vehicle density of $\lambda_v = 42$ vehicles/km.

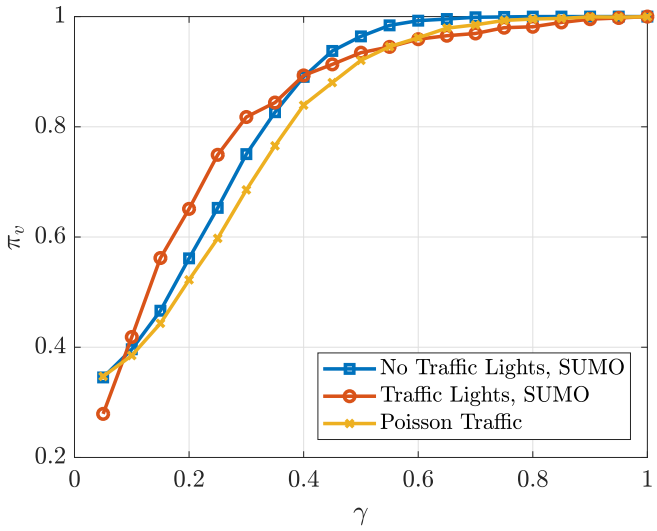


Figure 2.12: Connectivity probability as a function of γ for three different traffic patterns, $\eta = 3$, $\lambda_r^{-1} = 1$km, $\lambda_v = 42$ vehicles/km.

As shown on Figure 2.12, the Poisson assumption underestimates the

connectivity probability of a typical vehicle as compared to a realistic high-density traffic pattern scenario. This can be explained by the fact that faster vehicles may be stuck behind slower ones when the highway gets congested, leading to larger clusters of vehicles. In addition, in the traffic light scenario, the typical vehicle has a better connectivity than in the Poisson traffic regime when γ is small, as a typical vehicle is likely going to be in a large cluster formed at the traffic lights. However, for larger values of γ, the connectivity in the traffic light scenario becomes slightly lower than under Poisson traffic, as vehicles that are not in those traffic light generated clusters have less opportunity to cooperate with other vehicles than if the traffic were Poisson. Moreover, when γ is close to 0, a V2V+V2I capable vehicle, may be disadvantaged by being in the clusters formed at the traffic lights as the high density of legacy vehicles considerably reduces its field of view, and hence its ability to reach other V2V+V2I capable vehicles. For this reason, the Poisson traffic scenario leads to slightly better performance than the traffic light one in this regime.

In summary, a typical vehicle's connectivity analysis shows that the Poisson traffic assumption underestimates the network's performance compared to realistic non-Poisson scenarios for small values of γ, and may be a reasonable assumption when γ grows larger.

2.8.3 Insight on Alternative Distributions

Although the PPP assumption will be a good fit in certain regimes, it will still fail for others that may arise in the future, e.g., where cars may intentionally form platoons to increase highway throughput. To better understand how such patterns might affect connectivity, in this section we ask the question "What is the best possible configuration of cars, i.e., resulting in the best connectivity metrics?". We shall focus on two performance metrics: coverage π_v and mean rate per user. Two regimes can be distinguished. The first one corresponds to situations where $\lambda_v \geq 1/d$, i.e. where the vehicle density is large enough so that vehicles can be separated by $1/d$ meters. In such a scenario, vehicles would form a single infinite cluster leading to $\pi_v = 1$ and maximum mean rate per user since all the RSUs are in use. The other regime of interest is where $\lambda_v < 1/d$. Consider first a configuration where all the clusters in the network are of same size. Then spacing the vehicles by d within the cluster would ensure maximal cluster length, and hence maximal π_v and $\mathbb{E}[R_v]$ as this would maximize the "space covered" by clusters and thus the RSU busy time. Similarly, spacing vehicles in adjacent clusters by $2d$ would also maximize $\mathbb{E}[R_v]$, without affecting the coverage. Following these two rules, we derive expressions for π_v and u, the average RSU utilization capturing the same information as $\mathbb{E}[R_v]$. For a fixed cluster size n:

$$\pi_v(n) = \min[(n+1) \cdot d \cdot \lambda_r, 1] \tag{2.19}$$

$$u(n) = \min[\frac{n+1}{n} \cdot d \cdot \lambda_v, 1] \tag{2.20}$$

Clearly, as n increases, $\pi_v(n)$ increases while $u(n)$ decreases. We exhibit that trend through a tradeoff curve between coverage and throughput as a function of n in Figure 2.13:

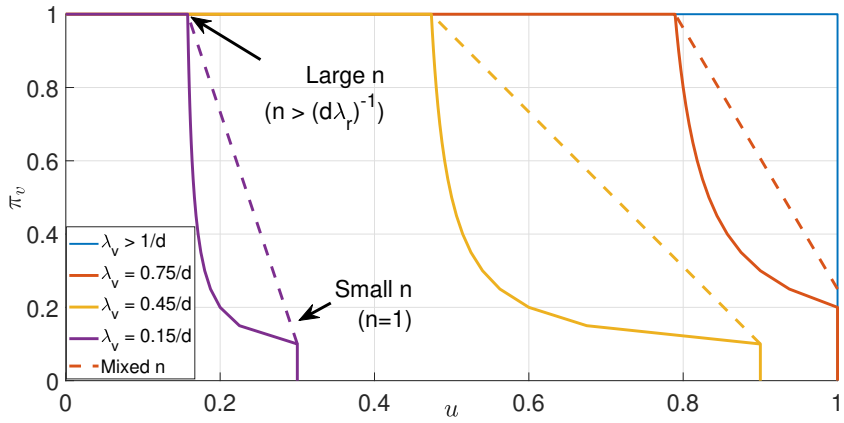


Figure 2.13: Tradeoff curve between connectivity π_v and RSU utilization u, for different λ_v (in vehicles/km), and the achievable performance by mixing cluster sizes.

Figure 2.13 exhibits the tradeoff between connectivity and throughput. In a low density regime, vehicles form longer clusters but cover less area as the cluster size n increases, improving the connectivity but reducing the average RSU utilization, and hence the mean rate per user. We note that when the vehicle density λ_v is large enough, the tradeoff does not occur as vehicles can get full connectivity and maximum mean rate per user. In scenarios where cluster size mixing is allowed, cluster can see an even better performance, represented by a straight line between any two points on the tradeoff curves. We note that the best mixing possible is combinations of clusters of size 1, i.e. isolated vehicles, and clusters of size $n = \left\lfloor \frac{1}{d\lambda_r} + 1 \right\rfloor$. The tradeoff curves asso-

ciated with such mixings are drawn as dashed lines on Figure 2.13. Intuitively, clusters of size 1 help to maximize the total area covered by the clusters, while the largest clusters increase the connectivity probability of a typical vehicle. Different combinations of those two cluster sizes can be constituted to reach any specific connectivity or throughput target.

2.9 Performance Evaluation of Multilane Dynamic Networks

So far, our analysis and validations have focused on a snapshot of a highway system at a given time. While the Poisson model for vehicle locations on the road holds in free-flow traffic as discussed in the previous section, the evolution of the vehicle location over time may impact the overall network performance. In this section, we perform additional time-domain validations based on SUMO traces. We describe below the SUMO simulation setup.

Scenario: We consider a three-lane straight highway of length 10,000 meters, where vehicles that reach the end of the road are regenerated at its beginning. The vehicle density is mentioned alongside each experiment.

Traffic: The highway speed limit is set to be 35 meters/sec., however, each vehicles sets a random target velocity by picking a velocity multiplier from a normal distribution of mean 1.0 and standard deviation 0.3. The multiplier values are then capped between 0.2 and 2.0. In addition, we adopt the Krauss car-following model in our experiments.

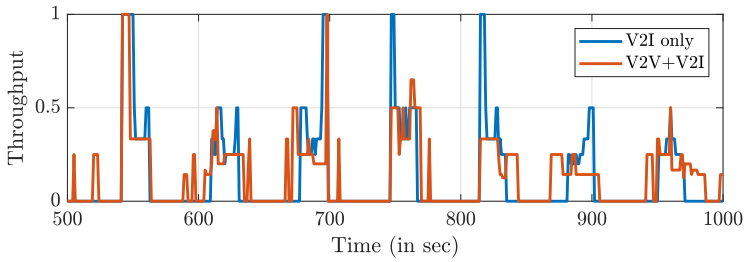
Simulation Duration: The simulation duration is set to $5,000$ sec., but

only samples starting from $2,500$ sec. are considered to ensure enough mixing in the vehicles' positions.

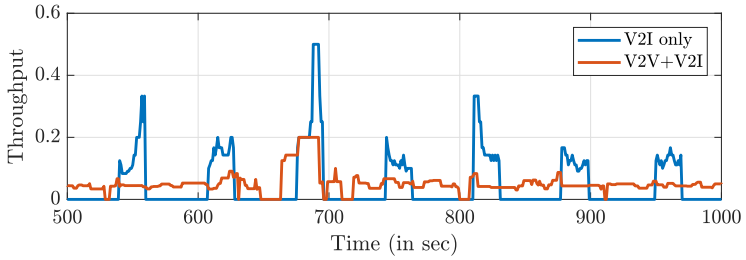
We now present the results of our three SUMO experiments.

2.9.1 Reduction in Throughput Temporal Variability

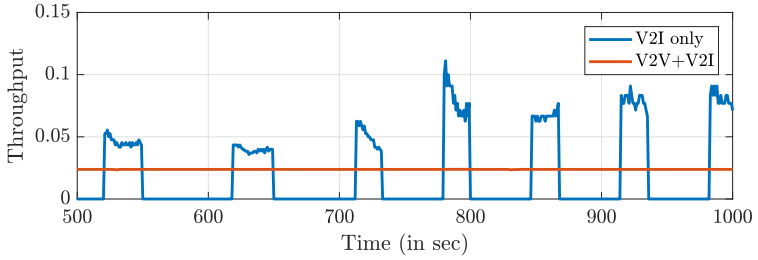
As discussed throughout this chapter, one key advantage of a V2V+V2I topology over V2I only connectivity is that a typical vehicle sees a considerable reduction in throughput variability, while remaining connected to the network for longer periods of time. Figure 2.14 shows how the throughput seen by a typical vehicle varies over time, for different vehicle densities and penetration rates.



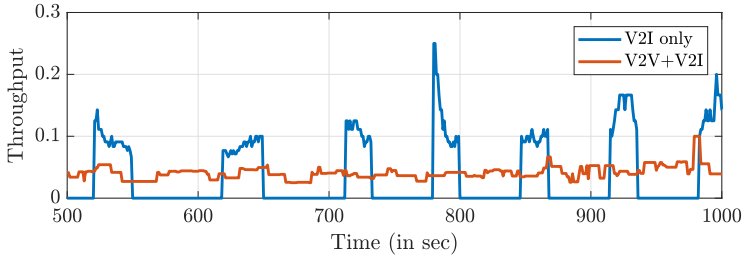
(a) Throughput Time Trace, $\lambda_v = 5$ veh./km, $\gamma = 1$.
$\mathbb{E}[R_v] = \mathbb{E}[R_v^*] = 0.167, \mathrm{Var}(R_v) = 0.0776, \mathrm{Var}(R_v^*) = 0.0640$.



(b) Throughput Time Trace, $\lambda_v = 20$ veh./km, $\gamma = 1$.
$\mathbb{E}[R_v] = \mathbb{E}[R_v^*] = 0.056, \mathrm{Var}(R_v) = 0.0123, \mathrm{Var}(R_v^*) = 0.0042$.



(c) Throughput Time Trace, $\lambda_v = 42$ veh./km, $\gamma = 1$.
$\mathbb{E}[R_v] = \mathbb{E}[R_v^*] = 0.0219, \mathrm{Var}(R_v) = 1.03e-3, \mathrm{Var}(R_v^*) = 1.74e-8$.



(d) Throughput Time Trace, $\lambda_v = 42$ veh./km, $\gamma = 0.5$.
$\mathbb{E}[R_v] = \mathbb{E}[R_v^*] = 0.0411, \mathrm{Var}(R_v) = 4.1e-3, \mathrm{Var}(R_v^*) = 2.84e-4$.

Figure 2.14: Time-Domain Dynamic Simulation of a typical vehicle's throughput, $d = 150$m, $\lambda_r^{-1} = 1$ km, $\eta = 3$, $\rho^{RSU} = 1$.

One can observe that in the non-cooperative scenario, a typical vehicle sees alternating on/off periods during which vehicles see high rates, before being disconnected. The frequency of the on-periods depends on λ_r^{-1} and the vehicle's velocity, while the on/off durations depend on the communication range d_v, as well as the vehicles' velocity. As can be seen in the V2V+V2I scenarios, the throughput time trace is steadier in general, which is consistent with the results in Theorem 2.5.5. While the shared rate improvement may not be very clear in very low traffic regimes, e.g., for $\lambda_v = 5$ vehicles/km, reduction in variability becomes much more visible for larger vehicle densities, where vehicles have more opportunities to cooperate and form clusters. The variance in throughput effectively vanishes when the vehicle density is large enough, e.g., $\lambda_v = 42$ vehicles/km, as the max-min fair scheduler is able to allocate resources perfectly given the large number of of V2V links (large λ_v, large γ).

Moreover, as predicted by our analysis, smaller penetration rates negatively impact the network's performance. However, allowing vehicle cooperation on a multilane highway has been shown to address this issue, see Theorem 2.6.3. The last time trace in Figure 2.14 shows that even with low/medium penetration rates, cooperation considerably reduces the throughput variability seen by a typical vehicle in the network.

2.9.2 Study of the Rate at which Clusters Change

One challenge that needs to be considered in real deployment of such cooperative ad-hoc networks is cluster management. As vehicles are moving on the highway at different velocities, clusters of vehicles will inevitably split and merge with others over time. If the rate of such "cluster change" events is too high, then the network may not be able to dynamically allocate resources to vehicles, hence impacting the network performance. We study the viability of our proposed V2V+V2I scheme, by studying the rate of "cluster change" events seen by a typical vehicle over time. Figure 2.15 shows how this metric changes as a function of the penetration rate γ, for different values of vehicle intensity λ_v.

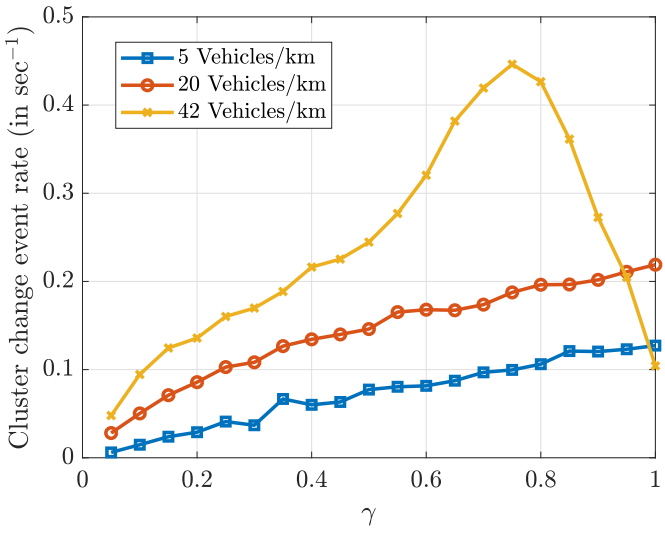


Figure 2.15: Mean Rate of cluster change event vs. the penetration rate γ in a V2V+V2I scenario, $\eta = 3$.

We observe on this figure that the rate of cluster change event increases

with γ. As more vehicles are equipped with V2V technology, clusters get larger, hence more vehicles might be susceptible to leaving it, while more vehicles may join it as well. However, when the V2V+V2I capable vehicle density becomes large enough, the rate of change events decreases, as vehicles are likely to form larger clusters that are harder to split. In the limit, one can expect that all the vehicles form a single large cluster, further reducing the change rate.

One important takeaway is the fact that even at its peak, the rate of cluster change remains reasonably small (on the order of one change every two seconds) as compared to the reactivity of the control plane of today's networks, making the V2V+V2I topology a viable solution.

2.9.3 Sensitivity of Coverage Probability to RSU placement

Finally, we evaluate how the coverage probability changes under random perturbations in the RSU locations. This is a crucial study, as in real-deployments, network operators may not have full control on the exact RSU placement. We simulate a network where the RSU locations are independently perturbed by a random amount between 0 and d_p meters from either side of their initial location (regularly placed $\lambda_r^{-1} = 1$ km apart), where we vary d_p from 0 to 300 meters. For brevity, we present our results without an associated figure. The random perturbations do not affect the mean connectivity time of a typical vehicle, for any vehicle intensity λ_v and penetration rate γ, as the RSU coverage regions never overlap if $d_p \leq \lambda_r^{-1}/2 - d$. Perturbations do however affect the variance in the time to access the network, i.e., a vehicle

requesting connectivity service at a random instant will see more variability in the time until it can connect to the network. Still, V2V+V2I cooperation helps to improve the connectivity probability, hence reducing the mean time to access the network, along with its variance.

2.10 Chapter Conclusion

In this chapter we have analyzed the performance of a multi-homed V2V+V2I architecture. Our main conclusion is that V2V relay clusters along with RSU multi-homing improves significantly the typical vehicle coverage probability and reliability, while reducing the variability of the shared rate per user when compared to a traditional V2I architecture. These properties position this architecture as a critical enabler for Internet connectivity services in future vehicular networks. We also conclude that the V2V technology penetration level is critical in the system performance given that many legacy vehicles will obstruct the LoS and prevent some vehicles to communicate. These difficulties may be mitigated if dedicated lanes are used by new vehicles that are V2V+V2I capable, particularly at low penetration levels. Moreover, we proposed a new mechanism to bound the performance of multi-lane highways by equivalent single lane highways, and our simulation results highlight a robustness of performance to heterogeneous vehicle distributions across lanes. We then described how the results presented throughout the chapter would change if one could control the relative positions of the vehicles on the road, e.g., when autonomous vehicles form platoons, and how the

connectivity-throughput tradeoff can be formally characterized in such scenarios. Finally, the theoretical results (e.g., reduction of variability of the rate seen by a typical vehicle), key assumptions of our model (e.g., PPP assumption, regular RSU placement), and practical deployment considerations (e.g., rate of cluster change impacting cluster management) have been validated via a micro-mobility traffic simulator.

Chapter 3

Throughput Analysis of Collaborative VANETs in Cellular Networks

The previous chapter proposed a solution to support the connectivity requirements associated with vehicle-bound users demanding reliable and high-throughput connectivity. We leveraged Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) links (commonly called V2X) possibly operating in the millimeter-wave (mmWave) frequency bands, while deploying Road Side Units (RSUs) close to the vehicles. The "ubiquitous" availability of V2X connectivity offers the prospect of enabling new vehicle-based services, e.g., data relaying and caching, that could also reduce the traffic loads on traditional cellular networks. In addition, RSU deployment might come at substantial costs for network operators, and solutions leveraging the existing cellular infrastructure may therefore be preferable.

In this chapter[1], we focus on leveraging vehicle clustering using the V2X technology to provide improved cellular connectivity for infotainment content

[1]Publications based on this chapter: [84] S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, Enhancing Cellular Performance via Vehicular-based Opportunistic Relaying and Load Balancing. IEEE INFOCOM 2019, April 2019; and S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, Analysis of Opportunistic Relaying and Load Balancing Gains through V2V Clustering. Submitted to IEEE Transactions on Vehicular Technology, [Under Review].

delivery to vehicle passengers (as opposed to delay-sensitive safety data). The central challenge is to develop an understanding of the performance and trade-offs of vehicular-based wireless architectures, when taking into account the roles of the vehicle clusters on the roads and the cellular network geometry, while adopting a more realistic wireless link model than in Chapter 2. As in that chapter, we consider a setting wherein clusters of well connected vehicles share possibly multihomed connectivity to the cellular infrastructure, i.e., one or more Base Stations (BSs) can transmit data to a cluster of vehicles which can in turn relay data to the appropriate vehicle. This leads to two types of benefits which we discuss next.

3.1 Overview of Benefits Associated with V2V-Clustering

3.1.1 Opportunistic Throughput Gains

The first benefit stems from the significant throughput gains achievable through opportunistic relaying to vehicle clusters. For example, as shown in Figure 3.1, rather than sending directly to a vehicle v_4 at the cell edge, a BS b_1 can send data to a nearby vehicle v_1 and the cluster can then use high capacity V2V connectivity to relay data to v_4. Given the order of magnitude differences in the peak capacity of nearby users compared to edge users in a typical cell, as long as V2V capacity is plentiful the potential of such cluster-based cooperative relaying is extremely high. When v_4 and v_1 lie in the same cell we refer to this as *intra-cell opportunism*, and if b_1 uses relay v_1 to forward data to a vehicle in another cell (say v_5) we call this *inter-cell opportunism*. This approach might

be particularly relevant in mmWave based infrastructures, whose short range and susceptibility to obstructions make efficient deployment challenging. By leveraging cluster-based relaying one can exploit spatial diversity to find Line of Sight (LoS) channels to BSs, providing improved coverage, throughput and reliability, see Chapter 2.

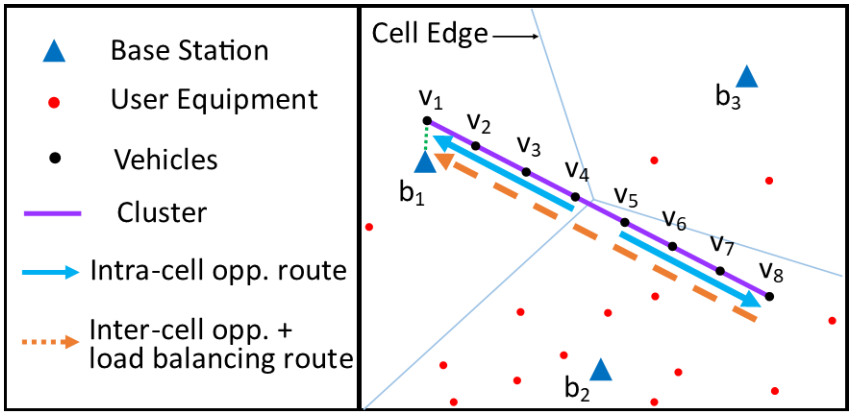


Figure 3.1: Example of an eight vehicle cluster traversing two cells shared by other User Equipments.

3.1.2 Load Balancing Gains

The second benefit comes from enabling load balancing across neighboring cells. For example, as shown in Figure 3.1, the traffic destined to a cluster of vehicles spanning multiple BS cells, e.g., b_1 and b_2, can be delivered through either one or both of the BSs, depending on their current loads. For instance, b_1 and b_2 currently have 5 and 15 users/vehicles in their respective cells but by using inter-cell cooperative relaying the loads could be shifted so that they each serve 9 and 11 respectively. Such an approach can help reduce

the variability across cell loads, hence diminishing the spatial and temporal variability in users' perceived shared rate. This is especially important in the context of 5G where small-cells cover limited regions and thus might see higher relative load variability. Moreover, although the link between the cluster and the lightly-loaded BS b_1 may be weaker than the link to b_2, the former will be able to allocate more wireless resources to the cluster, benefiting not only the vehicles, but also other devices associated to b_2 as fewer users are contending for channel access.

3.2 Related Work

Extensive research efforts have recently been devoted to investigating the benefits of opportunistic relaying in cellular networks and cell association load-balancing. We present next an overview of relevant related work in both directions.

Many researchers have explored gains that can be achieved through opportunistic relaying in cellular networks [73, 36, 177, 15, 175, 98, 179, 80], commonly exhibiting gains in throughput, rate fairness, and/or outage probability. For instance, [15] proposes a promising software framework that leverages opportunism to improve by 2× the total throughput delivered by a WiFi-based WLAN network. Our work shows that much greater gains can possibly be achieved in large-scale cellular networks by leveraging V2V relaying to serve vehicle-bound users. Studies such as [175] analyze the opportunistic gain in the context of VANETs, and also show that opportunism improves the down-

link throughput. The focus is, however, on comparing the performance of different routing strategies, and proposing efficient relaying protocols rather than analyzing the gains associated with opportunism and load balancing. Although [175] studies RSU-based networks, it provides some valuable insight regarding exploiting opportunism, that can be applicable in cellular-network settings. The authors in [80] show that up to 5.7× data rate gains an be achieved by the cell-edge users and 4.1× gains for the median users via Device-to-Device (D2D) opportunistic relaying. In our work, we show that additional gains can be expected by considering the geometry of vehicle clusters on the roads in cellular networks as well as the role of load-balancing.

Another line of work explored the benefits of load balancing in wireless networks. Some proposed solutions include channel borrowing [42], cell breathing [139, 16], BS association biasing [174, 146], centralized dynamic inter-cell and intra-cell handovers [150], distributed user association policies under heterogeneous traffic [91], and combinations thereof. Our work proposes a novel load-balancing solution leveraging V2V connectivity among vehicles driving on a road network served by the cellular infrastructure.

Other works have also investigated the benefits of D2D-based load balancing. In [45], spectrum savings enabled by D2D-based load balancing across cells were investigated. In our work, we characterize instead the relaying benefits in terms of user shared-rate and fairness gains. In [180], the authors introduce an optimization framework to find the optimal load-balancing and routing strategies in D2D-relay-based networks, considering a sum-rate maximization

objective, but do not study the potential fairness gains associated with D2D-relaying. In addition, in contrast to both of these works, we leverage tools from the stochastic geometry literature [14] to understand the role that the V2V-cluster relay geometry plays on their load-balancing ability, and hence, on the large-scale cellular network performance. Other works have established the critical role that load balancing plays in improving mean user rate or improving a notion of fairness, see, e.g., [97]. This work has been extended to vehicular network settings where V2I and V2V links are used to offload traffic from one cell to another [160, 167, 149]. While these works exhibit the benefits of load balancing, they focus on defining routing strategies, rather than evaluating the resource allocation and the potential per-user rate gains that a load balancing scheme might generate. The traditional approach to balance mobile users' loads across cells is via the formulation of an optimization problem, see e.g., [45, 180, 137] which in turn suggests appropriate scheduling algorithms, e.g., [43, 180, 102]. Other researchers propose learning-based solutions to determine effective association policies, see e.g., [100], but perhaps lack the development of underlying insights useful towards the design of vehicular network-based relaying strategies. Finally, some papers focused on studying multihomed load balancing schemes. For instance, [30] presents algorithms and experimental results showing how load balancing can improve the performance of multihop multihomed VANETs, but no network modeling and analysis was performed, and the study was mainly focused on uplink access, while we focus on multihomed downlink connectivity in this work.

This work is, to the best of our knowledge, the first one evaluating jointly opportunistic and load balancing gains by leveraging V2V cluster-based relaying to enhance the cellular infrastructure.

3.3 Chapter Contributions and Organization

The main objective of this chapter lies in modeling and analyzing the potential benefits of vehicle cluster-based opportunistic relaying in terms of shared rate gains and improved fairness. More specifically, this chapter makes five major contributions.

First, we present a model to study the performance of cellular networks leveraging V2V-clustering. The model captures the essential features and tradeoffs associated with this technique, while leveraging tools and results already established in the field of stochastic geometry.

Second, we study analytically the sources of intra-cell opportunistic gains, providing additional insight on the benefit of V2V-clustering, and providing tools to assess the performance gains.

Third, we formulate a network-level (centralized) fairness oriented resource allocation and load balancing optimization problem, allowing us to capture the full gains associated with intra-cell and inter-cell opportunism, as well as load balancing through BS multihoming.

Fourth, we propose a cluster-level (distributed) and computationally efficient load balancing algorithm that greedily and locally re-associates vehi-

cles to BSs. We assess its performance by comparing it to the network-level fairness optimal algorithm, and policy which only leverages intra-cell opportunism. We then argue that our cluster-level resource management algorithm is suitable for real-time dynamic allocation compared to a centralized network-level solution, and may be preferable despite its sub-optimality.

Finally, we discuss technical challenges associated with V2V cluster-based relaying, such as incentive mechanisms, the impact on packet delays, as well as real-time cluster management challenges.

The remaining of chapter is organized as follows. In Section 3.4 we propose our stochastic geometric network model and we present in Section 3.5 the associated analysis geared at understanding the roots of intra-cell opportunistic gains. In Section 3.6, we introduce a centralized network-level and a distributed cluster-level resource allocation and user association algorithms leveraging intra-cell opportunism, inter-cell opportunism, and load balancing. Section 3.7 presents simulation results for a variety of scenarios suggesting $10\times$-$20\times$ shared rate gains along with significant improvements in shared rate fairness. In Section 3.8, we present a critical outlook on V2V-clustering by highlighting major technical challenges associated with the proposed network architecture. Finally Section 3.9 concludes the chapter.

3.4 System Model

In this section, we propose a system model enabling us to study gains associated with opportunism and load balancing in cellular networks enhanced

by V2V cluster relaying.

3.4.1 Network Model

We consider a network where BSs are randomly placed on the plane according to a homogeneous Poisson Point Process (PPP) Φ_{BS} with intensity λ_{BS}, see e.g., [11]. Another independent homogeneous PPP Φ_{M} of intensity λ_{M} models the locations of the mobile User Equipment (UE). In this network, the road infrastructure is modeled as an arbitrary stationary line process Φ_{R} of line intensity λ_{R} meters of road per m^2, and independent of Φ_{BS}. Conditioned on a realization ϕ_{R} of this road infrastructure, vehicles with a fixed Line-of-Sight V2V-communication range of d_{R} meters are dropped on the roads and form vehicle clusters.

Definition 3.4.1 (Vehicle Cluster)**.** *Given an arbitrary configuration of vehicles on a road, a vehicle cluster is a sequence of vehicles on the same road such that any two consecutive vehicles are within communication range d_R of each other.*

It follows from this definition that a vehicle can only belong to a single cluster. Vehicles (hence clusters) are modeled as randomly distributed on the roads according to the following bursty-traffic model:

- Vehicle clusters consist of an independent random number of vehicles where the typical cluster size Z follows an arbitrary (but known) discrete distribution.

- Vehicles are equispaced[2] within a cluster, and the inter-vehicular distance is fixed to be $d_{\mathrm{V}} \leq d_{\mathrm{R}}$ meters, consistent with Definition 3.4.1, i.e., clusters can be seen as chains of successive vehicles within communication range of each other. Thus, the random size Z of a typical cluster induces a random length L in meters from the first to the last vehicle, such that $L = (Z-1)d_{\mathrm{V}}$ meters. We use the convention that a cluster with one vehicle has length 0, a cluster with two vehicles has length d_{V}, etc.

- Clusters are dropped on roads in ϕ_{R}, such that the distance between two consecutive clusters on the same road is random. Specifically, the typical inter-cluster distance T between the last vehicle of a typical cluster and the first vehicle of the preceding one follows an arbitrary (known) distribution satisfying $T > d_{\mathrm{R}}$ meters almost surely, consistent with Definition 3.4.1. This captures the requirement that two vehicles in different clusters cannot be within communication range of each other, otherwise they would be part of the same cluster.

For any given d_{V} and known $\mathbb{E}[Z]$ and $\mathbb{E}[T]$, one can characterize the vehicle density λ_{V} on a road in vehicles/meter as follows:

$$\lambda_V = \frac{\mathbb{E}[Z]}{(\mathbb{E}[Z]-1)\cdot d_{\mathrm{V}} + \mathbb{E}[T]} = \frac{\mathbb{E}[Z]}{\mathbb{E}[L] + \mathbb{E}[T]} \tag{3.1}$$

[2]While the equispaced model for vehicle placement within a cluster is idealized, our extensive experiments showed that for a given cluster length distribution the vehicle configuration within the cluster has an almost negligible impact on the network performance. Hence, we shall adopt the equispaced model to keep the subsequent analysis tractable.

The above clustered vehicle model induces a spatial process $\Phi_{\mathrm{V}} = \{V_i : i \in \mathbb{N}\}$ denoting the locations of the vehicles in the network, which is not a PPP. Each vehicle is assumed to also correspond to an active user, i.e., with full-buffer traffic. In the sequel we let $\phi_{\mathrm{BS}} = \{b_i : i \in \mathbb{N}\}$ denote a realization of the PPP Φ_{BS} and refer to BSs directly through and their locations, e.g., b_i. This convention is adopted for all point processes. As shown in Figure 3.2, the BSs in ϕ_{BS} induce a Voronoi tessellation $\mathcal{T}(\phi_{\mathrm{BS}}) = \{\mathcal{T}^b(\phi_{\mathrm{BS}}) \mid b \in \phi_{\mathrm{BS}}\}$, where each BS b has an associated cell: $\mathcal{T}^b(\phi_{\mathrm{BS}}) = \{x \in \mathbb{R}^2 \mid \|x - b\|_2 \leq \|x - b'\|_2, \forall b' \in \phi_{\mathrm{BS}}\}$. Based on this tessellation of BSs' cells we define the following additional notation. The set of vehicles ϕ_{V} is partitioned such that $\phi_{\mathrm{V},c}$ denotes vehicles belonging to cluster c while ϕ_{V}^b denotes the set of vehicles in BS b's cell. Similarly, the set of mobile UEs ϕ_{M} in the network is partitioned such that ϕ_{M}^b denotes the set of UEs in BS b cell. Finally, we let ϕ_{C}^b denote the set of clusters that include *at least one* vehicle in b's cell, while $\phi_{\mathrm{BS},c}$ denotes the set of BSs containing at least one of cluster c's vehicles in its cell.

Simulated Network Model. While our analytical results will be based on the above general network model, we shall adopt a more specific model for our simulation results, by relating the parameters d_{V}, λ_{V}, the distributions of the random variables Z and T, and the road process Φ_{R} in a consistent manner. Note that while Φ_{R} can be selected independently, the other parameters need to be carefully jointly chosen so as to satisfy all the model constraints, i.e., Equation 3.1, and the communication range constraints. To that end, we shall generate the road infrastructure as a Poisson Line Process (PLP) Φ_{R} of line

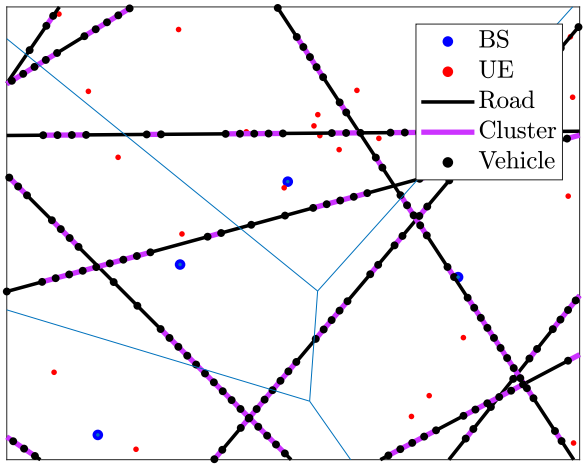


Figure 3.2: Illustration of a random network realization, i.e., realizations of ϕ_{BS} and ϕ_{M} modeling the BS and UE locations, along with the induced $\mathcal{T}(\phi_{\mathrm{BS}})$, and an arbitrary road infrastructure supporting randomly placed vehicle clusters.

intensity λ_{R} meters of road per m^2, see [37, 34]. As for the cluster-generation parameters, we shall fix λ_{V}, and select the three other ones accordingly as follows:

- We model $Z \sim \text{Geometric}(e^{-\lambda_{\mathrm{V}} d_{\mathrm{R}}})$, which corresponds to the cluster size distribution as if the vehicles were distributed as a PPP on the roads and grouped if they were within communication range d_{R}, as in Chapter 2.

- We let $d_{\mathrm{V}} = \lambda_{\mathrm{V}}^{-1}(1 - \frac{\lambda_{\mathrm{V}} d_{\mathrm{R}} e^{-\lambda_{\mathrm{V}} d_{\mathrm{R}}}}{1-e^{-\lambda_{\mathrm{V}} d_{\mathrm{R}}}})$ corresponding to the *mean* inter-vehicular distance within a cluster if the vehicles were distributed as a PPP on the roads, i.e., for $X \sim \text{Exp}(\lambda_{\mathrm{V}}), d_{\mathrm{V}} = \mathbb{E}[X|X \leq d_{\mathrm{R}}]$, see, e.g., Chapter 2.

- We model $T \sim \text{Exp}(\mu)$ such that $T > d_{\text{R}}$ almost surely, consistent with the corresponding distribution if the vehicles were distributed as a PPP on the roads, and corroborated by empirical observations in bursty-traffic settings [66]. The parameter μ is selected to be consistent with Equation 3.1, i.e., $\mu = \lambda_{\text{V}}$.

3.4.2 Link Capacity Model

In our analysis, we will consider downlink transmissions. For the *traditional cellular* network, i.e., without V2V cluster relaying, we model the downlink capacity from BS $b \in \phi_{\text{BS}}$ to user $u \in \phi_{\text{V}}^b \cup \phi_{\text{M}}^b$ (i.e., to vehicle a mobile UE) for a given network realization as depending on the Signal-to-Interference-and-Noise-Ratio (SINR) given by:

$$\text{SINR}_u^b = \frac{p_{\text{BS}} \cdot H_u \cdot P_u^b}{I_u^b + \sigma^2}, \tag{3.2}$$

where p_{BS} is the BS transmission power, H_u models the independent Rayleigh fast-fading gain such that $H_u^b \sim \text{Exp}(1)$, P_u^b is a random variable modeling the path-gain of the link between b and u, I_u^b is the interference power seen by user u associated to BS b, σ^2 models the total thermal noise power over the allocated bandwidth. Letting d_b^u denote the distance between user u and BS b, we model LoS blocking for all the wireless links through the dual-slope distance-dependent binary random variable P_u^b purposed in the 3GPP standard [53]:

$$P_u^b = \begin{cases} k_{\text{LoS}} \cdot (d_u^b)^{-\alpha_{\text{LoS}}} & \text{w.p.} \quad p_{\text{LoS}}(d_u^b), \\ \min \begin{bmatrix} k_{\text{LoS}} \cdot (d_u^b)^{-\alpha_{\text{LoS}}} \\ k_{\text{NLoS}} \cdot (d_u^b)^{-\alpha_{\text{NLoS}}} \end{bmatrix} & \text{w.p. } 1 - p_{\text{LoS}}(d_u^b), \end{cases} \tag{3.3}$$

where k_{LoS} and k_{NLoS} capture the signal attenuation at a reference distance of 1 meter for LoS and NLoS links respectively, while α_{LoS} and α_{NLoS} represent the respective path-loss exponents such that $\alpha_{\text{LoS}} \leq \alpha_{\text{NLoS}}$. Moreover, $p_{\text{LoS}}(\cdot)$ is a non-increasing non-negative function of the distance d_u^b, satisfying $p_{\text{LoS}}(0) \leq 1$. Finally I_u^b is the interference power seen by user u, originating from all the BSs except b, i.e.,:

$$I_u^b = \sum_{b' \in \phi_{\text{BS}} \setminus \{b\}} p_{\text{BS}} \cdot H_u^{b'} \cdot P_u^{b'}. \tag{3.4}$$

We assume for simplicity that a user u associated with BS b always observes interfering signals from other BSs through NLoS links, i.e., for all b' in $\phi_{\text{BS}} \setminus \{b\}$, we have $p_{\text{LoS}}(d_u^{b'}) = 0$.

Finally, the average transmission rate r_u^b from BS b to user u for a link of bandwidth w for a given network realization is modeled for simplicity by the Shannon ergodic rate:

$$r_u^b = w \cdot \mathbb{E}_{\{H_u^b\}_{b \in \phi_{\text{BS}}}} \left[\log_2 \left(1 + \frac{\text{SINR}_u^b}{\Gamma} \right) \right], \tag{3.5}$$

where Γ models the gap between the actual transmission rate and the Shannon capacity, modeling the joint effect of quantized modulation schemes, finite-length codes, channel estimation error due to vehicle mobility, etc., see [79], and the expectation is taken over all the fading terms. Note that we do not average the transmission rate over large-scale SINR variations such as blocking and link distance/vehicle mobility as we leverage opportunism with respect to these fluctuations.

In the *cluster-based opportunistic relaying* scenario, UEs see the same capacity as in the traditional cellular network setting. By contrast, the average downlink transmission rate from BS $b \in \phi_{\mathrm{BS}}$ to any vehicle belonging to cluster $c \in \phi_{\mathrm{C}}^{b}$ is modeled by

$$r_{c}^{b,*} = \max_{v \in \phi_{\mathrm{V},c} \cap \phi_{\mathrm{V}}^{b}} r_{v}^{b}, \tag{3.6}$$

i.e., the transmission rate from b to a (relay) vehicle in cluster c and in BS b's cell. In the sequel, we shall refer to this relay vehicle as the *cluster-head*, and a cluster may have multiple cluster-heads if it is multihomed. Note that we make cluster-head decisions based on the user average rates r_{v}^{b} as we envision the cluster-head selection decisions/handoffs to realistically occur on slower time-scales (on the order of hundreds of milliseconds to seconds) than the short channel coherence time associated with fading experienced by vehicles that may be moving at high velocity (on the order of milliseconds). We shall further make the following assumption which is in line with a setting where vehicles use high capacity V2V line of sight links, e.g., mmWave, to connect to the vehicles directly ahead and/or behind them in the same cluster.

Assumption 3.4.2 (V2V link model)**.** *We assume intra-cluster V2V links have sufficiently high capacity (e.g., mmWave bands) so as to ensure they are not the bottleneck in relaying traffic to vehicles within clusters, and do not interfere with infrastructure transmissions (e.g., sub-6GHz bands).*

We envision the allocation of wireless resources for the V2V links to be in line with the 5G NR V2X Sidelink resource allocation schemes described in

the 3GPP Release 16, i.e., either coordinated by the relevant BSs (mode 1) or uncoordinated, where vehicles access wireless resources from a resource pool preallocated by the BS (mode 2), see [63].

3.5 Intra-cell Opportunism Performance Analysis

As introduced earlier in this chapter, we are ultimately interested in analyzing both opportunism and load balancing gains. While the opportunism gain analysis can be tractable, the study of load-balancing gains is more complex as it is intrinsically related to the resource allocation policies used in the network. In this section, we focus on understanding the cause and effect of intra-cell opportunism solely, in a scenario where no load balancing is performed. In subsequent sections we will get a full picture of the intra-cell opportunism, inter-cell opportunism and load balancing gains. Recall that intra-cell opportunism considers data relaying only among vehicles that are in the same cluster and in the same BS cell. In other words, clusters are assumed to be artificially broken at the cell boundaries, creating logically independent sub-clusters. While this mechanism clearly reduces the benefits of V2V cooperation, a formal analysis of this setting allows for a better understanding of the origin of the gains associated with V2V cluster relaying.

The gains associated with intra-cell opportunism can be summarized in terms of two phenomena: (1) clustering allows vehicles to route traffic through the closest vehicle in their cluster to the BS, leveraging a higher SINR link; and (2), routing traffic through vehicles closer to the BS improves robustness to

blocking, by increasing the probability of benefiting from a LoS wireless link. Although these two effects are closely related, we shall study them separately.

3.5.1 Clustering Reduces the Effective Distance

The most obvious benefit of intra-cell opportunism is the flexibility to route traffic through the vehicle that sees the best BS link in the cluster. A well known result in stochastic geometry characterizes the random distance between a typical user in a cellular network and its closest BS as following a Rayleigh distribution [14], under the assumption that the BSs are deployed according to a PPP. This result still applies to our framework despite the vehicles not following a PPP, as the roads and cluster generation processes are all independent of Φ_{BS}. Hence, we let the random variable D denote the distance between a typical vehicle in the network and its closest BS in the *traditional network* scenario, such that:

$$f_D(d) = 2\pi\lambda_{\mathrm{BS}} d e^{-\lambda_{\mathrm{BS}}\pi d^2}, \forall d \in \mathbb{R}_+. \tag{3.7}$$

To quantify analytically the potential gains associated with intra-cell opportunism, we derive the distribution of the *effective distance* D^* between the typical vehicle and its closest BS in the *intra-cell cooperative network* scenario, i.e., the distance between the BS $b \in \phi_{\mathrm{BS}}$ and the cluster-head associated with a typical vehicle's cluster $c \in \phi_{\mathrm{C}}$. We have:

$$D^* = \max_{v' \in \phi_{\mathrm{V},c} \cap \phi^b_{\mathrm{V}}} D^b_{v'}. \tag{3.8}$$

We note that while Equation 3.8 resembles Equation 3.6, the closest vehicle to the BS in the cluster may not be the one relaying the data, as it may not provide the best link in the case where it is not in LoS link to the BS. Nevertheless, studying D^* allows us to quantify the potential of effective distance reduction gains though V2V relaying. Using the network geometry, one can develop an expression for the conditional c.d.f. of D^* given $D = d$.

Theorem 3.5.1 (Effective Distance Conditional c.d.f.)**.** *Given the distance $D = d$ between a typical vehicle and its closest BS, and the distribution for the typical cluster size Z, the c.d.f. of the distance between the closest vehicle in the typical vehicle's cluster to the serving BS is given by:*

$$\mathbb{P}(D^* \leq x | D = d) = \tag{3.9}$$
$$\begin{cases} 1, & \text{for } 0 \leq d \leq x, \\ \dfrac{2}{\pi \mathbb{E}[Z]} \displaystyle\int_0^{\theta_0(d,x)} e_0(d,x,\theta) \cdot \sum_{i=1+\lceil \frac{l_0(d,x,\theta)}{d_V} \rceil}^{\infty} \mathbb{P}(Z \geq i) d\theta, & \text{for } 0 \leq x \leq d, \end{cases}$$

and the associated variables and functions are given in Table 3.1.

Table 3.1: Theorem 3.5.1 Intermediary Variables; for $x \leq d \in \mathbb{R}, \theta \in [0, \frac{\pi}{2}]$

$\theta_0(d,x) = \sin^{-1}(x/d)$
$l_0(d,x,\theta) = d\cos(\theta) - \sqrt{x^2 - (d\sin(\theta))^2}$
$d_0(d,x,\theta) = d_{\mathrm{V}} \cdot \left\lceil \frac{l_0(d,x,\theta)}{d_{\mathrm{V}}} \right\rceil$
$r_0(d,x,\theta) = \sqrt{d^2 + d_0(d,x,\theta)^2 - 2d \cdot d_0(d,x,\theta)\cos(\theta)}$
$a_0(d,x,\theta) = \pi r_0^2 - \left[r_0^2 \cos^{-1}\left(\frac{d_0^2+r_0^2-d^2}{2d_0 r_0}\right) + d^2 \cos^{-1}\left(\frac{d_0^2+d^2-r_0^2}{2d_0 d}\right) - \frac{\sqrt{(r_0+d-d_0)(d_0+r_0-d)(d_0+d-r_0)(d_0+r_0+d)}}{2} \right]$
$e_0(d,x,\theta) = e^{-\lambda_{\mathrm{BS}} a_0(d,x,\theta)} \cdot \mathbb{1}\left\{ \left\lfloor \frac{l_0(d,x,\theta)}{d_{\mathrm{V}}} \right\rfloor \neq \left\lfloor \frac{l_0(d,x,\theta)+2\sqrt{x^2-(d\sin(\theta))^2}}{d_{\mathrm{V}}} \right\rfloor \right\}$

A formal proof of this theorem is provided in Appendix B.1. One can then directly derive from Theorem 3.5.1 and Equation 3.7 the (unconditional) distribution of D^*. Figure 3.3 exhibits the reduction in the effective distance that the typical vehicle experiences in a traditional and cooperative relaying network by comparing the c.d.f.'s of D and D^*. As seen on the figure, a typical

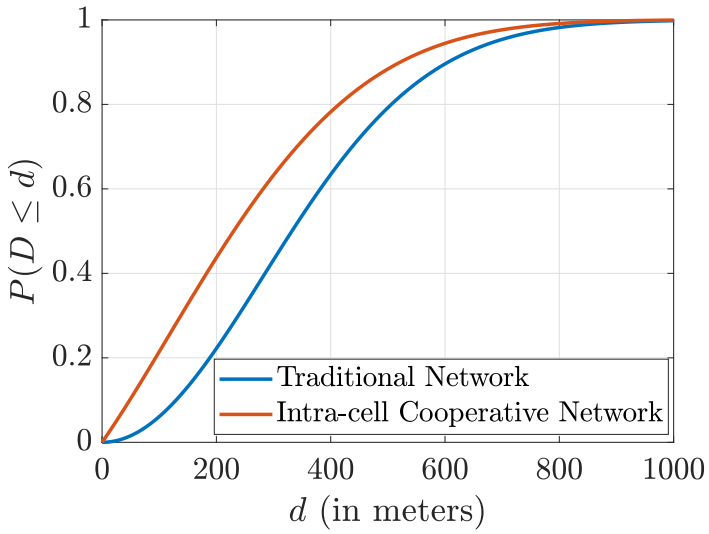


Figure 3.3: Comparison of the c.d.f.'s of the effective distance between the typical vehicle and its attached BS, under the traditional and cooperative network scenarios, for $\lambda_{\text{BS}} = 2$ BSs/km^2, $\lambda_{\text{V}} = 30$ vehicles/km and $d_R = 100$m.

vehicle is expected to benefit from considerable gains associated with reduced effective distance to the tagged BS. For instance, 43% of the vehicles will be effectively within 200m from their attached BS thanks to intra-cell cooperation, while only 23% would be within this range in a traditional network. Besides these considerable direct gains, this effective distance reduction between a typical vehicle and its associated BS induces another benefit that we study

next.

3.5.2 Clustering Improves Robustness to Blocking

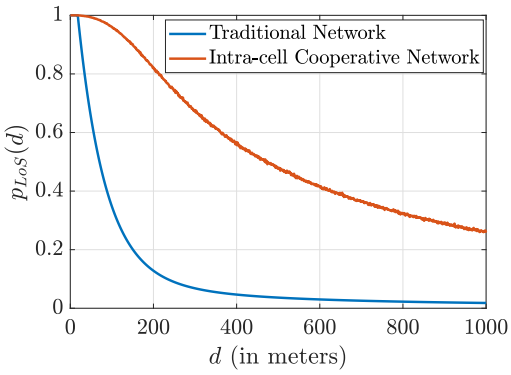
Another benefit of V2V cluster relaying is the flexibility to circumvent large blocking objects, such as buildings, that may interrupt LoS links and attenuate the received SINR. Indeed, vehicle clustering not only provides diversity through additional candidate links that have the potential to have a LoS to the BS, but closer vehicles are clearly more likely to benefit from such LoS links (see Equation 3.3). More precisely, if the probability that a typical vehicle a distance $D = d$ from its BS sees a LoS link is $p_{\mathrm{LoS}}(d)$ in a traditional network setting, we denote the equivalent metric in the cooperative relaying setting by $p^*_{\mathrm{LoS}}(d)$. In this formulation, a typical vehicle a distance $D = d$ from its BS is in a sub-cluster of size $\tilde{Z}$ (containing vehicles in the same cluster and same cell as the typical vehicle) whose vehicles are at distances $\mathbf{D} = (D_1, D_2, \ldots, D_{\tilde{Z}})$ from the BS, where there is an i such that $D_i = D$ almost surely. Now $p^*_{\mathrm{LoS}}(d)$ has the following form, assuming the LoS probabilities of vehicles within the same sub-cluster are conditionally independent given $\mathbf{D}$:

$$p^*_{\mathrm{LoS}}(d) = \mathbb{E}^0_{\Phi_{\mathrm{BS}},\Phi_{\mathrm{V}}}[1 - \prod_{i=1}^{\tilde{Z}}(1 - p_{\mathrm{LoS}}(D_i)) \mid D = d] \tag{3.10}$$

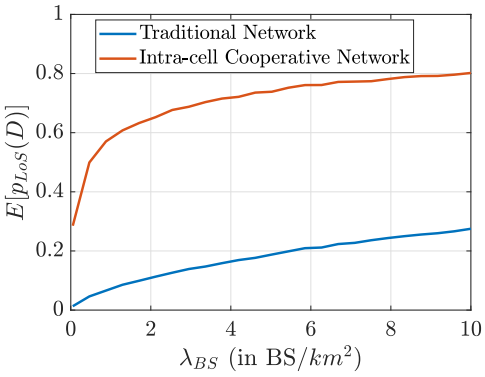
where $\mathbb{E}^0[\cdot]$ denotes the Palm expectation with respect to the typical vehicle and its subcluster. Clearly, we have $p^*_{\mathrm{LoS}}(d) \geq p_{\mathrm{LoS}}(d), \forall d$, i.e., the probability that at least one of the vehicles in the typical vehicle's sub-cluster sees a LoS link is always larger than the probability that the typical vehicle sees one

without cooperation. We note that the conditional independence assumption is realistic when the inter-vehicle distance d_{V} is not too small. This ordering is exhibited in Figure 3.4a, which shows the LoS probability with and without V2V cluster relaying for a vehicle at a given distance from its closest BS, taking into account the random sub-cluster sizes. In the sequel, we adopt the following baseline $p_{\mathrm{LoS}}(\cdot)$ function, as proposed in the 3GPP Release 14 standard in an Urban Macro-cell (UMa) environment [53]:

$$p_{\mathrm{LoS}}(d) = \begin{cases} 1, & \text{if } d \leq 18m, \\ \frac{18}{d} + (1 - \frac{18}{d}) \cdot e^{-d/63}, & \text{if } d > 18m. \end{cases} \tag{3.11}$$



(a) LoS probability for a vehicle a distance d away from its closest BS, for $\lambda_{\mathrm{BS}} = 2$ BSs/km^2.



(b) LoS probability for a typical vehicle as a function of λ_{BS}.

Figure 3.4: Study of LoS probability for $\lambda_{\mathrm{V}} = 30$ vehicles/km and $d_R = 100$m.

Figure 3.4a shows a considerable improvement in the probability of benefiting from a LoS link. The gains are particularly significant for large values of d, i.e., for cell-edge vehicles. Vehicle clustering can then be seen as a mechanism that reduces the need to densify the network with BSs, hence

reducing the infrastructure deployment costs as well as the mean interference power level in the network. This phenomenon is noticeable in Figure 3.4b showing how $\mathbb{E}_D[p_{\text{LoS}}(D)]$ varies as a function of the BSs density λ_{BS}.

First, one observes that when the distribution of the distance D between a typical vehicle and its closest BS is taken into consideration, the probability of a LoS link increases 6-fold between the traditional and intra-cell cooperative network scenarios, for the network parameters selected in Figure 3.4a, i.e., $\lambda_{\text{BS}} = 2$ BSs/km^2. Second, vehicle clustering enables substantial savings in the density of BS needed to achieve a specific p_{LoS} level for a typical vehicle. For instance, to guarantee that a typical vehicle sees a LoS with probability 0.28, λ_{BS} needs to be equal to 10 BSs/km^2 in a traditional network, while the same performance can be achieved with $\lambda_{\text{BS}} = 0.05$ BSs/km^2 in the cooperative setting for the selected network parameters. Hence, by using V2V cluster relaying, a network operator could achieve considerable savings in infrastructure deployment, when providing service only to vehicles.

3.5.3 Mean Shared Rate Gains through Intra-cell Opportunism

We now study the joint effect of the reduced effective distance among vehicles and their associated BSs, and the improved probability that they benefit from LoS links on the vehicles' mean shared rate. The vehicles' mean shared rate is defined to be the rate received by the vehicles after sharing the wireless resources amongst the vehicles and UEs. In the intra-cell cooperative network scenario, we consider a proportionally fair resource allocation scheme,

i.e., all the users receive the same fraction of resources regardless of their link quality, and see a shared rate proportional to their average transmission rate. Hence, for a given network realization, vehicle v belonging to cluster c attached to BS b would receive in the traditional network a shared rate

$$s_v = \frac{r_v^b}{|\phi_V^b| + |\phi_M^b|}, \qquad \forall v \in \phi_{\mathrm{V}}^b, \tag{3.12}$$

and in the intra-cell cooperative network scenario a shared rate

$$s_v^{*,\mathrm{intra}} = \frac{r_c^{b,*}}{|\phi_V^b| + |\phi_M^b|}, \qquad \forall v \in \phi_{\mathrm{V}}^b \cap \phi_{\mathrm{V,c}}. \tag{3.13}$$

In addition, for a mobile UE m, s_m is defined similarly to s_v, and we have $s_m = s_m^{*,\mathrm{intra}}$. Figure 3.5 exhibits how both effects presented in this section can improve the mean shared rate by comparing s_v to $s_v^{*,\mathrm{intra}}$, using the network parameters in Table 3.2 based on the 3GPP standard [53].

One observes that intra-cell opportunism leads to a considerable shared rate boost for the vehicles in the network, allowing them to experience higher shared rate especially when the roads are congested. At this stage, we emphasize that these considerable gains in mean shared rate result from the sole effect of intra-cell opportunism. In reality, clusters can cross cell edges (and are not artificially interrupted as assumed here for tractable analysis) leading to longer clusters providing even further opportunities for reduction in effective distance, and finding a LoS link to the BS. As argued in the rest of this chapter, further gains are to be expected through effective network-level resource allocation strategies, allowing for instance the balancing of vehicular

Table 3.2: Network Simulation Parameters

Parameter	Value	Units
λ_{BS}	2	BSs/km^2
λ_{M}	10	UEs/km^2
λ_{R}	4.5	road km/km^2
d_R	100	m
p_{BS}	40	dBm
w	100	MHz
k_{LOS}	-34	dB
k_{NLOS}	-19.5	dB
α_{LoS}	2.2	–
α_{NLoS}	3.9	–
σ^2	$-199 + w\|_{dB}$	dBm
Γ	3	dB

loads across cells, benefiting vehicles, but also mobile UEs that do not have any relaying abilities.

3.6 Resource Allocation Algorithms for Cluster-Based Cooperative Relaying Networks

In order to study the full performance gains associated with V2V cluster relaying, i.e., leveraging both opportunism and load balancing, the wireless resource (e.g., time and/or bandwidth) allocation mechanisms need to be defined as the network performance depends considerably on the adopted policy. We no longer assume that clusters are interrupted at the cell edges, and we allow for cluster multihoming.

The resource allocation problem can be broken down into two sub-problems:

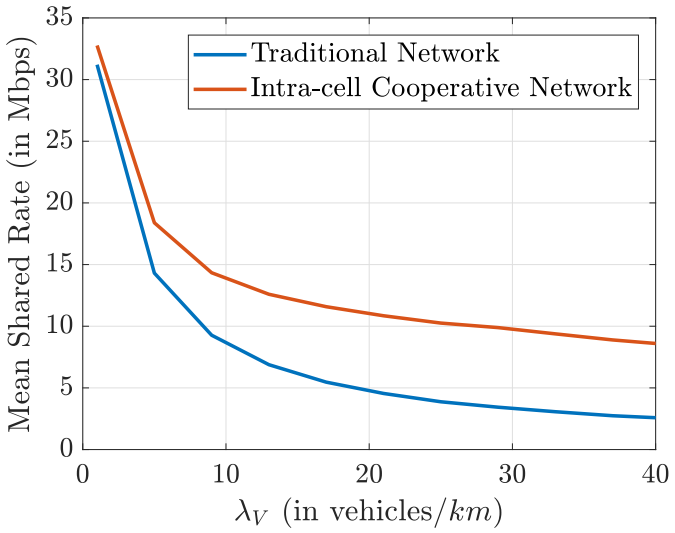


Figure 3.5: Comparison of the vehicles' mean shared rate in the traditional and intra-cell cooperative scenarios as a function of λ_V, for the simulation parameters in Table 3.2.

1. How should resources be allocated by BSs to serve the clusters and mobile UEs?

2. How should resources be shared amongst vehicles within each cluster?

A reasonable sharing strategy within a cluster is to divide resources equally amongst the associated vehicles. In our formulation, we will assume such a sharing policy, relegating the discussion of alternative strategies to Section 3.8. Below, we introduce two different algorithms addressing the first sub-problem, while Section 3.7 focuses on evaluating their respective performance.

3.6.1 Network-Level Fairness-Optimal Joint Rate Optimization

We first consider a centralized network-level joint optimization of opportunistic relaying and load balancing, aiming at fairly allocating the wireless resources among the mobile UEs and vehicle clusters. The optimization framework solves the first sub-problem for a given network configuration in a finite region. As described in Section 3.4, we shall assume that a BS $b \in \phi_{\mathrm{BS}}$ serves the set of mobile UEs ϕ_{M}^b in its cell, where for $m \in \phi_{\mathrm{M}}^b$ the average transmission rate is r_m^b given in Equation 3.5. Similarly, BS b may serve vehicles in any cluster $c \in \phi_{\mathrm{C}}^b$ which would in turn perceive an average rate $r_c^{b,*}$, as defined in Equation 3.6. However, as the BSs need to share their resources among the vehicle clusters and mobile UEs in their cells, each entity shall be served only for a fraction of the transmission time. Specifically, BS b decides on an allocation vector $\boldsymbol{\pi}^b = (\pi_i^b : i \in \phi_{\mathrm{M}}^b \cup \phi_{\mathrm{C}}^b) \geq 0$, representing the fraction of time allocated to the mobile UEs and clusters it can serve, such that $\|\boldsymbol{\pi}^b\|_1 = \sum_{i \in \phi_{\mathrm{M}}^b \cup \phi_{\mathrm{C}}^b} \pi_i^b = 1$, and we let $\boldsymbol{\pi} = (\boldsymbol{\pi}^b : b \in \phi_{\mathrm{BS}})$. For a given allocation vector (potentially a function of the network realization), we define the shared rate $s_m^{*,\mathrm{inter}}$ perceived by mobile UE m attached to BS b as

$$s_m^{*,\mathrm{inter}} = \pi_m^b r_m^b, \qquad \forall m \in \phi_{\mathrm{M}}. \tag{3.14}$$

while vehicle v in cluster c multihomed through a set of BSs $\phi_{\mathrm{BS},c}$ perceives $s_v^{*,\mathrm{inter}}$ such that

$$s_v^{*,\mathrm{inter}} = \frac{1}{|\phi_{\mathrm{V},c}|} \sum_{b \in \phi_{\mathrm{BS},c}} \pi_c^b r_c^{b,*}, \qquad \forall v \in \phi_{\mathrm{V},c}. \tag{3.15}$$

Recall that while mobile UEs are assumed to be served by only one BS, vehicle clusters can be *multihomed*, i.e., served by multiple BSs, explaining the summation in Equation 3.15.

One way to improve the network users' QoS is to provide them with steady data rates. By ergodicity, this can be achieved by ensuring a fair distribution of resources among the mobile UEs, and the vehicles by selecting an appropriate resource allocation vector $\boldsymbol{\pi}$. In general, there is a tradeoff between performance (measured in terms of mean shared rate per user) and fairness among the user allocations, see, e.g., [179]. One can define fairness is different ways, allowing one to control this tradeoff. For instance, a *max-min fair* resource allocation might be relevant for our scenario; but other fairness measures could also be used such as *proportional fairness*, that allocates resources proportionally to the link quality between the cluster and the BSs. In order to keep our framework as general as possible, we shall use α-fair utility functions, that model a range of fairness definitions via the parameter α, see [95]. For instance, proportional fair resource sharing corresponds to $\alpha = 1$, and max-min fair to a value of $\alpha \to \infty$. For each network user we posit an increasing concave utility function $\mathcal{U}_\alpha(\cdot)$ of its allocated shared rate s, where:

$$\mathcal{U}_\alpha(s) = \begin{cases} \frac{s^{1-\alpha}}{1-\alpha}, & \text{if } \alpha \geq 0, \alpha \neq 1, \\ \log(s), & \text{if } \alpha = 1. \end{cases} \tag{3.16}$$

With this notation in place, and given a network realization, the net-

work utility maximization problem is given as follows:

$$\max_{\boldsymbol{\pi}} \sum_{v\in\phi_{\mathrm{V}}} \mathcal{U}_\alpha(s_v) + \sum_{m\in\phi_{\mathrm{M}}} \mathcal{U}_\alpha(s_m)$$

$$\text{s.t.} \begin{cases} s_m = \pi_m^b r_m^b, & \forall m \in \phi_{\mathrm{M}}^b,\ \forall b \in \phi_{\mathrm{BS}}, \\ s_v = \frac{1}{|\phi_{\mathrm{V},c}|} \sum_{b\in\phi_{\mathrm{BS},c}} \pi_c^b r_c^{b,*}, & \forall v \in \phi_{\mathrm{V},c},\ \forall c \in \phi_{\mathrm{C}}, \\ \parallel \boldsymbol{\pi}^b \parallel_1 = 1, \boldsymbol{\pi}^b \geq 0, & \forall b \in \phi_{\mathrm{BS}}. \end{cases} \tag{3.17}$$

The above optimization problem is convex (as we maximize a concave objective function over a convex set) but may not have a unique optimizer [67]. Intuitively if there were a cycle of BSs linked by overlapping vehicle clusters it may be possible to shift resource allocations around the cycle while maintaining the same overall network utility.

While the proposed optimization framework allows the network to reach a proportional fairness-optimal resource allocation, one major issue associated with using such a network-level and centralized optimization algorithm is that it requires excessive computation, especially for large and congested networks. This is likely to hinder the ability to deploy and run such an algorithm in real-time, especially with highly mobile users such as vehicles that would impose a continual network re-optimization making such a solution impractical. In particular, most known algorithms to solve Problem 3.17 can find an ϵ-optimal solution within $\mathcal{O}(\delta/\epsilon^2)$ iterations, e.g., projected gradient ascent, or $\mathcal{O}(\delta/\epsilon)$, e.g., ADMM, where δ is the problem dimension, i.e., $\delta = |\phi_{\mathrm{BS}}||\phi_{\mathrm{C}} + \phi_{\mathrm{M}}|$, see [162]. Note that the real algorithm complexity might be even worse as each iteration typically has a dimension-dependent per-iteration complexity. Clearly, as the network grows larger, the overall complexity of this problem

also scales up, making it unusable in large-scale networks.

3.6.2 Cluster-Level Load-Balancing Algorithm

While the proposed optimization framework allows the network to reach a fairness-optimal resource allocations, one major issue associated with using such a network-level and centralized optimization algorithm is that it requires excessive computation, especially for large and congested networks. We propose in this section a decentralized algorithm solving the user-association and resource-allocation problems with a network-size independent complexity, allowing to spread the computations across multiple nodes in the network, while being less computationally intensive overall. As the vehicles are highly mobile, the optimal user association is likely to change quickly over time and it might be preferable to equip the network with an agile potentially sub-optimal algorithm, rather that a slow one that leads to an optimal yet obsolete solution. Unlike the centralized optimization framework we have proposed, this algorithm first solves the user-association problem and then allocates an equal amount of wireless resources to all the users associated to each BS. The idea is to have clusters asynchronously trigger a re-association routine at random or periodic times.

This re-association routine consists in having each cluster c update how many of its set $\phi_{\mathrm{V},c}$ of vehicle based users should be served by each of the BSs in $\phi_{\mathrm{BS},c}$, i.e., the BSs serving cells crossed by c. In particular let $\mathbf{n}_c = (n_c^b \in \mathbb{Z}_+ : b \in \phi_{\mathrm{BS},c})$ where n_c^b denotes the number vehicles in cluster c

served by BS b. Let $\mathbf{k}_c = (k_c^b \in \mathbb{Z}_+ : b \in \phi_{\mathrm{BS},c})$ where k_c^b denotes the number of *other* mobile UEs and vehicle based users BS b is currently serving (i.e., excluding the ones in c). Finally we shall define $\mathbf{r}_c^* = (r_c^{b,*} \in \mathbb{R}_+ : c \in \phi_{\mathrm{BS},c})$ where $r_c^{b,*}$ denotes the highest transmission rate BS b can achieve amongst cluster c's vehicles in its cell, as defined in Equation 3.6.

When the cluster management update is engaged, it takes the current vectors $\mathbf{k}_c$ and $\mathbf{r}_c^*$, and determines $\mathbf{n}_c^*$ that maximizes the cluster-level utility, defined as

$$\mathcal{L}_{c,\alpha}(\mathbf{n}) = \sum_{b \in \phi_{\mathrm{BS},c}} n^b \cdot \mathcal{U}_\alpha \left(\frac{r_c^{b,*}}{n_c^b + k_c^b} \right) \tag{3.18}$$

such that

$$\mathbf{n}_c^* \in \arg\max_{\mathbf{n}} \left\{ \mathcal{L}_{c,\alpha}(\mathbf{n}) \mid \sum_{b \in \phi_{\mathrm{BS},c}} n^b = |\phi_{\mathrm{V},c}| \right\} \tag{3.19}$$

i.e., each cluster greedily maximizes the network utility function over its own the set of vehicles based on local information. Note the above assumes each BS allocates an equal fraction of time to each of its UEs and vehicles. As a final step, the vehicles in $\phi_{\mathrm{V},c}$ aggregate their resources in a common pool and redistribute them uniformly among themselves, in such a way that all the vehicles in a cluster perceive similar rate allocation. Finding the optimal cluster association vector $\mathbf{n}_c^*$ (with respect to the cluster-level utility function) may require solving an NP-hard integer program, or attempting a brute force search over the set of weak integer compositions of $|\phi_{\mathrm{V},c}|$ into $|\phi_{\mathrm{BS},c}|$ parts, of cardinality $\binom{|\phi_{\mathrm{V},c}|+|\phi_{\mathrm{BS},c}|-1}{|\phi_{\mathrm{V},c}|}$, see [135]. While both options are computationally inefficient, we propose an alternative approach in Theorem 3.6.1. The proof

of correctness of this algorithm is provided in Appendix B.2.

Theorem 3.6.1 (Sequential Association Solves the Cluster-level Maximization Problem)**.** *Consider a cluster c and let* $\{\tilde{\mathbf{n}}_c^{(i)}\}_{i=0}^{|\phi_{V,c}|}$ *be a sequence of vectors in* $\mathbb{Z}_+^{|\phi_{BS,c}|}$*, defined as:*

$$\tilde{\mathbf{n}}_c^{(i)} \triangleq \arg\max_{\mathbf{n}} \left\{ \mathcal{L}_{c,\alpha}(\mathbf{n}) \mid \mathbf{n} = \tilde{\mathbf{n}}_c^{(i-1)} + \mathbf{e}_b, \text{ for some } b \right\}, \tag{3.20}$$

where $\mathbf{e}_b$ *is the* b^{th} *basis vector in* $\mathbb{Z}_+^{|\phi_{BS,c}|}$ *and* $\tilde{\mathbf{n}}_c^{(0)} \triangleq \overrightarrow{\mathbf{0}} \in \mathbb{Z}_+^{|\phi_{BS,c}|}$*, then* $\mathcal{L}_{c,\alpha}(\mathbf{n}_c^*) = \mathcal{L}_{c,\alpha}(\tilde{\mathbf{n}}_c^{(|\phi_{V,c}|)})$.

Therefore, determining $\mathbf{n}_c^*$ is a straightforward task of computational complexity $O(|\phi_{\mathrm{BS},c}||\phi_{\mathrm{V},c}|)$, that is independent of the network size. Assuming BSs track k_c^b and r_c^b the data requirement to perform a cluster c update is $O(|\phi_{\mathrm{BS},c}|)$.

For this algorithm, we envision vehicles follow a *dual-handoff* protocol wherein cluster-head handoffs, i.e., the decision to select a different cluster-head and BS handoffs, i.e., the decision of a cluster-head to associate to a different BS are taken separately, potentially on different time-scales. While one can rely on existing standardized protocols to manage the latter, the former needs further investigation. Clearly, in a dynamic network where vehicles and UEs are mobile, the cluster-head selection and cluster re-association routine described above need to be performed repeatedly, and the necessary rate of updates will depend on the vehicles' velocity. One mode of operation is to have clusters trigger updates after random exponential timeouts with given

mean rate, proportional to the vehicles' velocity. In order to study the significance of the mean update rate, we performed time-domain simulations and we examined the *spatial density of updates*, i.e., the expected number of re-association updates performed by a cluster per meter traveled, and computed as the ratio of the cluster update rate and the cluster velocity. This metric is motivated by the fact that the network performance expressed, e.g., in terms of mean shared rate seen by a typical vehicle, is invariant to an increase in both the network users' velocities and the mean update rate by a common factor. To perform the time-domain simulations, we extend the network and wireless link models described in Section 3.4 by assuming a linear constant-velocity trajectory for all the clusters on their roads, while mobile UEs move in random linear directions in the network at a velocity set to be $10\times$ slower than the vehicles. In addition, to capture the spatial correlation in the wireless channels in a mobile setting, we define LoS_t to be the event that a user moving at velocity v sees a LoS link to its closest BS at a distance d_t at time t, we propose the following simple Markovian model capturing the temporal correlation in the users' LoS link probability, which is consistent with results presented in [138]:

$$p_{\text{LoS}}(d_t|LoS_{t-1}) = \beta \cdot p_{\text{LoS}}(d_t) + (1-\beta) \cdot \mathbb{1}\{LoS_{t-1}\} \tag{3.21}$$

where $\beta = \min(\frac{v \cdot \Delta t}{d_{\text{corr}}}, 1)$, and d_{corr} models the LoS correlation distance. Figure 3.6 exhibits the network performance in terms of the typical user's mean shared rate, as a function of the spatial density of updates.

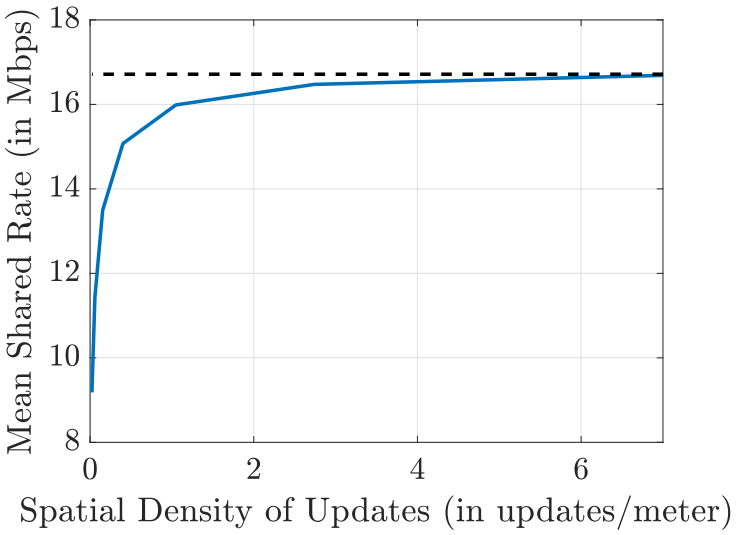


Figure 3.6: Figure of a typical vehicle's mean shared rate under the distributed cluster-level algorithm as a function of the spatial density of updates, for the parameters in Table 3.2, $\lambda_{\mathrm{V}} = 30$ vehicles/km, and $d_{\mathbf{corr}} = 30$m. The dashed line shows the asymptote when the network users are static.

The key takeaway of Figure 3.6 is that while the typical vehicle's mean shared rate improves as the re-association rate increases and/or the cluster velocity decreases, reasonable mean cluster-head handoff rates (e.g., on the order of one update every 50ms for clusters moving at medium to high velocity) are sufficient to ensure satisfactory performance.

3.7 Performance Evaluation of V2V Clustering

In this section, we discuss Monte-Carlo simulations, aiming at providing additional insight regarding different resource allocation and network management strategies. We compare the performance of the two resource allocation

algorithms presented in Section 3.6, and quantify the full gains associated with V2V clustering, i.e., joint opportunism and load balancing gains. We then study how V2V clustering can also improve the network robustness to instantaneous traffic surges.

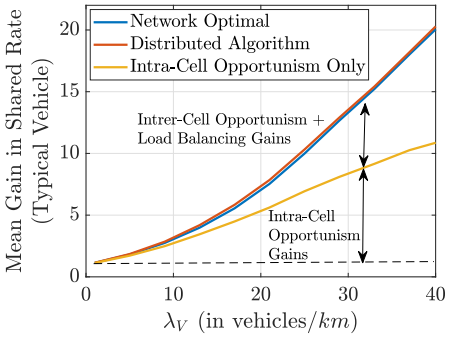
3.7.1 Resource Allocation Algorithms Performance Analysis

We now analyze and compare the network performance under the resource allocation strategies introduced in Section 3.6, and assess the full gains associated with opportunism and load balancing when vehicles are cooperating using V2V communication.

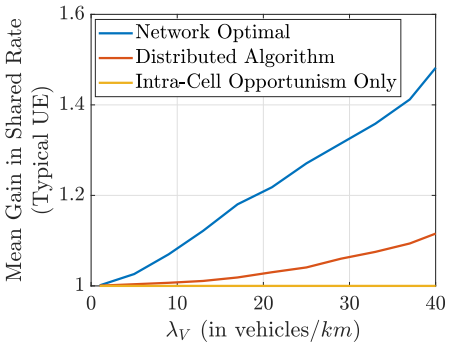
Performance is evaluated via two metrics: (1) the mean gain in per-user shared rate, and (2), the Jain's index measure of fairness in the per-user shared rate. All the mean gains in shared rate are relative to the non-cooperative scenario, i.e., we evaluate the *average of the ratio* of the shared rate received by a network user in the cooperative setting over the shared rate that the same user would perceive if vehicles were not cooperating, or more formally the spatial averages of $\frac{s_v^{*,\text{inter}}}{s_v}$ and $\frac{s_m^{*,\text{inter}}}{s_m}$, i.e., averaged over all the vehicles and UEs in space, respectively. We present the results as a function of the cluster density λ_{V} representing how congested the roads are. We evaluated network performance in Figure 3.7 for random network configurations based on the network parameters shown in Table 3.2.

Four settings were considered:

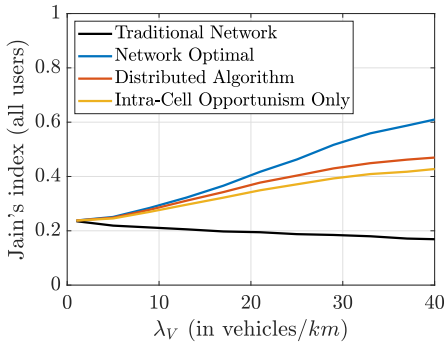
Traditional cellular: all users associate with their closest BS which shares



(a) Typical vehicle's gain in shared rate.



(b) Typical UE's gain in shared rate.



(c) Jain's index of per-user shared rate.

Figure 3.7: Resource Allocation Algorithms Performance Comparison for the parameters in Table 3.2.

its resources equally amongst them. This setting is used a baseline for the mean shared-rate gain computations, and does not explicitly appear in Figures 3.7a and 3.7b.

Intra-cell opportunism: setting analyzed in Section 3.4 were only cluster-based intra-cell opportunism was exploited, and all users in a cell also receive an equal amount of resources.

Network Optimal (PF): corresponds to solving the centralized network utility maximization Problem 3.17 introduced in Section 3.6 where all users have log utilities, i.e., Proportionally Fair (PF) resource allocation.

Distributed algorithm: refers to the cluster-side distributed algorithm previously described where the Voronoi cells were used to initialize the association, before conducting multiple cluster re-balancing updates. The number of such updates was five times the number of clusters in the simulated area, the updated clusters were selected at random.

Note that only the last two exploit both opportunism and load balancing. Multiple takeaways can be extracted from Figure 3.7.

First, one can observe on Figure 3.7a that a typical vehicle can see considerable gains in mean shared rate, regardless of the adopted algorithm, that may even reach 20× gains for policies leveraging both opportunism and load balancing. Such gains arise from the substantial inequities in link capacities amongst vehicles at the cell-edge compared to the ones in LoS with the BS. V2V cluster-based relaying is therefore an effective mechanism to bridge this gap. One can clearly distinguish the gains associated with intra-cell opportunism reaching around 10.86× with inter-cell and load balancing gains providing an additional 1.87× gain factor when $\lambda_{\mathrm{V}} = 40$ vehicles/km. We note that the reported gains are under the PPP assumptions for the BS, UE, and clusters placement, and we expect the load balancing gains to be even more substantial under spatially bursty traffic (as may happen on roads), when load balancing across cells is the most needed. Similarly, all strategies lead to a

considerable improvement in the users' shared rates Jain's index, as observable in Figure 3.7c, compared to the traditional cellular setting.

Second, we observe that our proposed cluster-level algorithm performs slightly better than the network fairness-optimal allocations in terms of mean gains in shared rate. In retrospect this is not surprising as the latter strategy optimizes for proportional fairness, rather than sum shared rate. This is reflected in Figure 3.7c, where the network optimization framework logically outperforms the other resource allocation policies in terms on rate fairness.

Third, we observe on Figure 3.7b that, on average, the mobile UEs also benefit from V2V relaying if the resource allocation policy leverages load balancing, although they are not actively relaying. While not all the UEs will benefit from the cooperative scheme, a typical UE will. Indeed, a typical UE is likely to belong to a large cell, that is expected to be highly loaded. This cell will benefit from load balancing by shifting the vehicles away, letting them to associate to smaller and less loaded neighboring cells. This mechanism frees up additional resources that can be shared with all the cell users (including the typical UE). In addition, UEs tend to benefit more from a network fairness-optimal resource allocation strategy, compared to the local cluster-based algorithm. This results from the fact that their perceived shared rate is explicitly taken into account in the problem formulation 3.17, whereas it only implicitly appears in 3.19, where clusters mainly consider the shared rate perceived by of their own vehicles while still attempting to balance the load across cells. The fact that the network fairness-optimal allocates more resources to the mobile

UEs compared to vehicles also provides an additional explanation to the fact that this policy underperforms the cluster-level distributed strategy from the perspective of the typical vehicle's shared rate gain, but largely outperforms it in terms of per-user shared-rate fairness when both vehicles' and UEs' rates are taken into account, as seen in Figure 3.7c.

In summary, the adoption of the distributed cluster-level algorithm seems to be justified. Aside from its computational advantages, it has been shown to outperform the network fairness optimal algorithm in terms of shared rate gain seen by a typical vehicle in the network, although it does not provide as much gains to the mobile UEs. In addition, the distributed algorithm's fairness performance remains satisfactory as compared to the network fairness optimal strategy, making it a suitable solution for the resource allocation and user association problem.

3.7.2 Robustness to Load Surges

The main benefit of balancing the network user load across cells is that it allows vehicles in highly congested cells to perceive satisfactory throughput by "sharing" wireless resources with less congested neighboring BSs. Clearly, the potential for effective load balancing is closely related to the cluster sizes, as longer clusters are more likely to be crossing cell boundaries and be multi-homed. We now study the relationship between cluster size and such balancing potential. We introduce the notion of *offloading potential* defined next, so as to best understand how to manage the network, e.g., arranging for specific traffic

configurations, imposing a maximum cluster size may come at a reasonably small performance cost.

Definition 3.7.1 (Offloading Potential)**.** *A cell's offloading potential is the maximum number of vehicles it can offload to neighboring cells, i.e., by leveraging V2V cluster relaying to serve vehicles in its cell via the resources of another BS.*

This metric captures the ability of the network to diffuse user/vehicular loads across the network. For instance, it is relevant to study settings where a cell observes a momentary surge in the number of mobile UEs requesting service due to, e.g., a temporary large-scale event gathering a substantial amount of people. For simplicity, we shall assume (for this experiment only) that all the cluster sizes are equal and deterministic. We simulate different scenarios, wherein we vary the cluster size Z, while keeping the vehicle density λ_{V} constant, i.e., increasing the cluster size proportionally decreases the cluster density (controlled by the mean of the typical inter-cluster distance T distribution). We show in Figure 3.8 the empirical c.c.d.f. of the offloading potential for different cluster sizes, exhibiting a tradeoff between the offloading potential of a typical BS and the fraction of BSs able to offload user loads.

Indeed, for a fixed vehicle density, a decrease in the cluster size induces an increase in the cluster density, making them more spatially scattered. This scattering is beneficial for the network as more BSs can be reached and hence, will be able to offload traffic. However, highly scattered networks are associated with very small clusters that are unlikely to be crossing cell boundaries

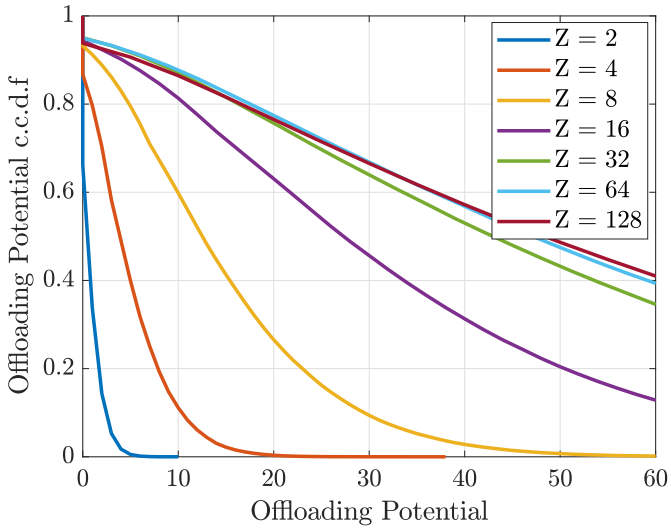


Figure 3.8: Offloading potential empirical c.d.f.'s for different cluster sizes Z (in number of vehicles), for $\lambda_{\mathrm{V}} = 30$ vehicles/km^2.

(i.e., be multihomed), and only meagerly contribute to increasing the cells' offloading potential. This is reflected in Figure 3.8 by a c.c.d.f. for a cluster size of 3 vehicles decaying quickly to 0. As the cluster size increases, the number of vehicles that can be offloaded out of a cell increases – the c.c.d.f.'s decrease at a slower rate. An interesting observation is the diminishing benefits of increasing the cluster size in the regime where clusters can cross multiple cells, as the offloading potential is constrained by the cell size (or more specifically by the length of random cell cross-sections). Hence, the distributions of offloading potential for cluster sizes of 50 and 100 vehicles are almost identical. However, increasing the cluster size comes at the cost of reducing the number of cells that can benefit from traffic offloading as the cluster density is reduced, hence more cells are likely not be serving any vehicle due to the reduced vehicle scattering,

explaining the fact that the offloading potential c.c.d.f. curve for $Z = 128$ for instance might be below the one associated with $Z = 64$ on the right-hand side of the figure. For the simulated network parameters, intermediate cluster sizes between 8 and 16 vehicles appear to be the most judicious, by allowing a considerable number of cells to benefit from network user offloading, while providing them with substantial offloading potential without facing substantial cluster management complications associated with large cluster sizes.

Remark: We have only considered networks where the roads are homogeneously placed in space. In reality, these roads may be spatially clustered, e.g., near metropolitan areas. While the theoretical analysis of such networks is out of scope of this chapter, we conjecture that while spatial correlation in the roads (and hence the clusters over different roads) does not impact intra-cell opportunism gains, it may improve inter-cell opportunism gains as highly congested cells would lead to additional benefits from load-balancing.

3.8 Technical Challenges

We complement our theoretical and numerical analysis with additional technical challenges that would need to be examined if V2V cluster relaying was deployed.

3.8.1 Incentive Mechanism for Cluster Relaying

The first challenge that needs to be addressed is the design of an effective incentive mechanism for vehicles to be willing to route traffic for others

in their cluster. Indeed, vehicles can be seen as selfish and greedy agents that need to be compensated for the additional transmission power and potentially additional overheads associated with routing and relaying.

To address this issue, two types of solutions can be proposed. First, the cluster-head vehicles are naturally incentivized to cooperate, when the network dynamics are considered. A cluster-head is indeed unlikely to keep this role continuously, as the vehicles are in motion and the channel qualities vary over time. A cluster-head is clearly better-off routing traffic for its cluster to benefit from future throughput improvement when its channel quality deteriorates. While this may not be necessary, a token-based mechanism can be implemented, as proposed in [184], where vehicles in each cluster would pay their cluster-head(s) through a virtual currency/tokens that can be redeemed at a later stage, when the latter can benefit from V2V-relaying.

Second, cooperate can be incentivised through more direct mechanisms. For instance, instead of equally dividing the wireless resources among all the vehicles in a cluster, they can be re-distributed in a way that rewards the cluster-heads, e.g., as a function of the rate each vehicle would have perceived without V2V cluster relaying. This solution has the advantage of providing instantaneous incentives to cluster-heads by providing them with additional throughput, but at the cost if negatively impacting the shared-rate fairness amongst the vehicles in the network. Additional benefits of instantaneous incentives over token-based have been discussed in [107].

Finally, we note that the incentive mechanism can also be designed to

benefit vehicles that are not cluster-heads, but that are willing to forward other vehicles packets in the cluster, i.e., intermediate nodes. Suitable incentive mechanisms would include the ones providing rewards (wireless resources or tokens) proportionally to the volume of data forwarded.

3.8.2 Delay Management in Cluster Relays

The second challenge that would need to be accounted for would be the additional delays associated with packet routing, especially when the clusters are large. In this chapter, we have assumed that the V2V links have very high capacity, hence leading to negligible transmission delays. However, this assumption may not always hold, and other types of delays may also be taken into account, such as packet processing and forwarding delays. Clearly, V2V-relaying is a promising technology that has been shown to provide considerable potential in terms of throughput, but it may be best fitted for applications that are not too delay sensitive.

One solution to partially address this issue would be to artificially break the clusters, e.g., by fixing a maximum cluster size to limit the overall packet delay to the furthest vehicle in the cluster. This would clearly impact the mean gain in shared-rate performance, but it would help to provide better delay guarantees. Network operators can then tune the maximum cluster size parameter to control the tradeoff between mean throughput and packet delay.

Finally, we note that even though the clusters can have considerable sizes, they are also likely to be multihomed and traffic can be effectively routed

from the closest serving BS, according to the association/resource allocation algorithm in effect. We show in Figure 3.9 a comparison of the distributions of the typical vehicle's cluster size and the number of hops experienced by the typical vehicle's packets, for the cluster-level distributed algorithm presented in Section 3.6. We observe that the typical vehicle experiences substantially

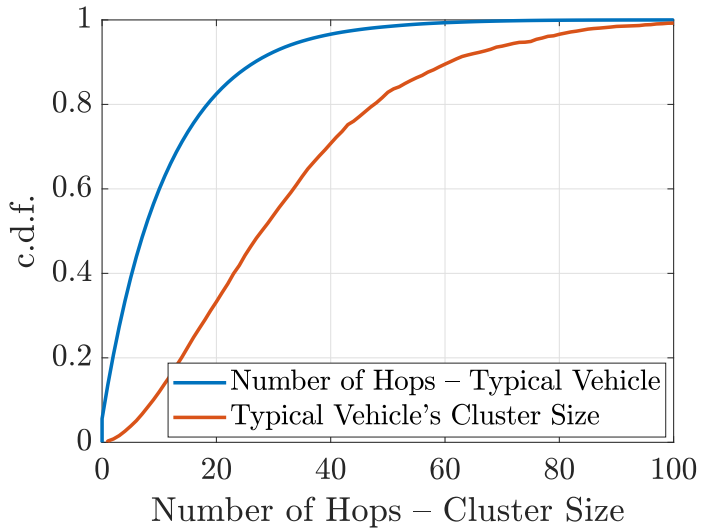


Figure 3.9: C.d.f.s of the typical vehicle's cluster size and the number of hops experienced by the typical vehicle's packets, for the parameters in Table 3.2.

fewer hops than its cluster size, and may not always leverage multihoming if the load across cells is very imbalanced. For instance, while the median typical vehicle's cluster size is 27 vehicles using the network parameters in Table 3.2, the median number of packet hops for a typical vehicle is only 7 V2V hops.

3.8.3 Real-Time Cluster Management

The third practical challenge associated with V2V clustering is the ability to manage the cluster in real-time. As discussed in Section 3.6, vehicles may be moving at high velocity, hence the cluster-head role and routing decisions will need to be frequently updated to ensure the best performance.

A perhaps more restrictive issue is that an efficient signaling protocol needs to be adopted to detect and adapt to changes in the vehicle-clustering and association. For instance, clusters may divide or merge over time, and BSs and cluster-heads need to be aware of such changes in real-time to adapt/re-optimize the resource allocation amongst users in the network. Two approaches could be used to resolve this issue. In the first solution, changes can be detected locally (e.g., by vehicles in the cluster) and the information can be propagated to the relevant cluster-head(s). For instance, cluster-heads of merging clusters can negotiate the best association strategy and propagate the information back to the relevant BSs. In the second solution, a centralized virtual controller (e.g., placed in an edge server) can aggregate information collected from vehicles to recompute the optimal re-association/vehicle clustering, and propagate the information back to the BSs and the vehicles. While the second solution may lead to better performance as more information is aggregated to take better decisions, the first solution may be preferable due to the simpler signaling required, and less computations, making it more reactive to sudden changes.

3.9 Chapter Conclusion

In this chapter, we studied the gains associated with V2V cluster-based relaying by identifying and characterizing the sources of opportunistic and load balancing gains analytically and via simulations. We established a network-level optimization framework to associate users to BSs and ensure a fair wireless resource allocation scheme. The results show that shared-rate gains from V2V-relaying can exceed an order of magnitude, and are associated with shared-rate fairness improvements across network users, stemming from both opportunism and load balancing. We then used this framework as a benchmark to study the performance of a cluster-level, efficient, and distributed vehicle association and resource allocation algorithm. We show that excellent performance gains can be achieved via this policy, while being more convenient than the network-level algorithm when executed in real-time. Furthermore, V2V relaying has been shown to help to provide improved service to all the network users, including mobile UEs that do not actively participate in the relaying scheme. While some technical challenges still need to be addressed, the considerable gains associated with V2V cluster relaying will motivate future development.

Part II

Timely Information Sharing in Collaborative Cloud/Edge Networks

Chapter 4

Timely Information Sharing in Fog Networks

Many of the emerging mobile applications require unprecedented compute power, e.g., autonomous vehicles, remotely controlled robots, Augmented Reality (AR) technologies, unmanned aerial vehicles, cloud gaming platforms, etc. Equipping mobile devices with the compute resources needed can be a considerable challenge for manufacturers due to cost, complexity, battery longevity, weight, and size constraints. A solution to overcome this challenge and bring to market such computation-hungry services is to (partially) offload compute to the cloud via wireless connectivity to remote servers. This chapter[1] explores the major communication/compute tradeoffs associated with computation offloading to cloud/edge servers and the induced timely information sharing with the physical device.

A flexible approach to support mobile devices with remote compute resources is through a server-side process running on a Virtual Machine (VM). If kept up-to-date, the process can keep track of a device's state in real-time, perform computations, and possibly send back control commands. Such processes

[1]Publications based on this chapter: [85] S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, Service Placement for Real-Time Applications: Rate-Adaptation and Load-Balancing at the Network Edge. IEEE EdgeCom 2020, August 2020.

are expected to become prevalent to support the management and control of mobile devices, e.g., robots, self-driving cars, smart cities devices [115]. However, in real-time settings, associating remote processes to devices poses several technical challenges. In particular, in order to maintain safety or offer an appropriate Quality of Service (QoS), the process needs to closely track the state of its device. In other words, updates among mobile device and server-side processes should not have "aged" too much to remain relevant. Maintaining such timeliness depends both on the update rate as well as communication/compute delays.

To support possibly stringent timeliness requirements, edge computing architectures have been proposed as means to reduce network delays, by moving the servers closer to the devices. By contrast, the alternative of hosting VMs in the cloud, typically further away from the devices, provides an attractive solution leveraging large pools of shared resources. Deploying mobile services at scale will require careful study of cost/performance tradeoffs of edge/cloud infrastructure based solutions.

4.1 Related Work

There has been substantial work in this area. We identify two relevant classes of work. The first class focuses on the need for mobile edge computing. The natural way to introduce the concept is to compare the characteristics of edge and cloud computing, as in [147, 35, 111]. In this chapter, we take this one step further by characterizing precisely the tradeoffs for real-time applications.

Additionally, we propose an intuitive hierarchical network model materializing the idea of "Cloud-to-Thing continuum", or Fog-to-Cloud, suggested in [35] and [113], where service providers can place compute resources anywhere in the network. This softens the dichotomy between edge and cloud, leading naturally to the optimal placement problem.

The second line of work focuses on service placement, i.e., where to instantiate VMs once a provider has dimensioned a graph of compute resources, e.g., [185, 178, 81, 46, 148, 108, 124]. These works propose various policies to optimize placement based on different performance metrics. For instance, [46] considers power consumption and transmission delay, [148] examines the number of services placed, [108] focuses on minimizing the violation in QoS, i.e., latency, while [124] uses user-specific reward functions. Other studies suggest approximation or genetic algorithms to solve the service placement problem. Furthermore, [140] suggests to solve a Mixed Integer Linear Program to minimize capital and operating expenditures to dimension the network, but the authors to not analyze the communication vs. compute tradeoff explicitly, and do not address heterogeneity in the device requirements (e.g., latency constraints and compute job size). In this work, we follow a different approach. We simplify the network model which allows us to extract basic insights, that we leverage to propose a more general service placement algorithm. Unlike the above-mentioned work, we propose a service placement policy that addresses the need to adapt to network congestion by adapting the update rates associated with mobile devices supporting real-time applications.

4.2 Chapter Contributions and Organization

In this chapter, we first explore the fundamental characteristics of provisioning edge/cloud compute resources for real-time mobile services. To that end we propose a stylized network model allowing us to capture the salient features of the network dimensioning problem. Based on the initial insights developed from studying the resource provisioning problem, we propose an online, adaptive and distributed joint service-placement and rate-adaptation policy that is more generally applicable, and that describes how the network is ought to be managed while operating.

The framework introduced in this chapter allows us to reach multiple conclusions. First, we identify key tradeoffs between cloud and edge computing, and show how the optimal provisioning and placement depend on the application's characteristics.

Second, we show how the relative cost of compute vs. communication impacts the optimal location of compute resources. In particular the most cost effective placement may not be in the cloud or at the edge, but rather at an intermediate level.

Third, we show that for any use-case, as the density of mobile devices grows placing compute resources at the edge becomes more cost effective. However, perhaps counter-intuitively, stricter timeliness guarantees makes it beneficial to shift compute resources further away in the cloud.

Finally, we introduce a device-side online distributed algorithm to man-

age dynamic mobile device loads by determining both the device's update rate and a server to host its VM. Our approach adapts its decisions to measured network congestion, which may not be under service provider's control. We show that under the proposed joint placement and rate-adaptation policy, near-optimal service availability can be achieved in large networks, and show the benefit over load balancing policies when applications choose fixed update rates.

The remaining of this chapter is organized as follows. Section 4.3 describes four mobile applications serving as running examples throughout the chapter. Section 4.4 proposes a highly stylized system model and network architecture, as well as an appropriate timeliness metric. Section 4.5 includes our problem formulation and result analysis. In Section 4.6, we study a more general setting, and analyze the performance of our online device-side joint service placement and rate-adaptation algorithm. We conclude the chapter in Section 4.7.

4.3 Mobile Edge Computing Services: Use Cases

Our work is motivated by several emerging applications/use cases including those developed in the context of 5G networks [7, 2]; specifically we focus on four types of applications:

(a) **XR Traffic:** Augmented Reality, Virtual Reality and Mixed Reality, generally referred to as extended reality (XR), have been the subject of extensive study in industry and academia as it is considered one of the in-

novative services to be supported by next generation wireless networks. XR devices have the particularity of requiring both considerable bandwidth and low latency, making the design of networks supporting such services challenging [52, 152, 51].

(b) **Vehicular Network Traffic:** Supporting self-driving and/or coordination amongst next generation vehicles may be based on exchanging basic safety messages or localization data. To be relevant, update messages will typically require tight timeliness constraints, but may require relatively little compute and communication resources.

(c) **Cloud Gaming Traffic:** In the near term cloud gaming may become the leading use-case. It has the potential to reduce the compute requirements on the gaming devices by performing computations in a remote server, enabling complex multiplayer games to be more accessible on-demand. Similarly to XR traffic, considerable data may be streamed from the server to the devices to enable high-quality graphics, but timeliness constraints may be looser.

(d) **IoT Device Traffic:** We shall also consider IoT devices that do not have strict and tight latency budgets, but that can potentially be massively deployed, e.g. smart home devices, or agricultural sensor networks. Typical traffic for such use cases consists of short and sporadic packets.

We summarize the requirements for these use-cases in Table 4.1.

Table 4.1: Network Requirements and Parameters per Use-Case

Use Case	Timeliness Constraint (τ_0, in ms)	Devices per BS (η, in devices/BS)	UL/DL Update Size (p_u, p_d, in kB/update)		Compute per update (ψ^{-1}, in Ops/update)	Refs.
XR	15	10	50	50	1e9	[96, 109, 130]
Vehicular Networks	10	40	0.4	0.4	1e7	[72, 1, 117, 130]
Cloud Gaming	100	5	5	100	1e10	[40, 77, 130]
IoT	10,000	500	0.2	0	1e4	[133, 130]

4.4 System Model

In this section, we introduce a network architecture and performance metrics that we use to explore the characteristics and tradeoffs associated with service placement and provisioning decisions for real-time applications.

4.4.1 Network Model

We shall initially take the perspective of a *(virtual) service provider* who pays for communication and compute resources from one or more *infrastructure providers*. Initially we assume that the service provider provides custom services to a homogeneous customer base of *mobile devices*, i.e., with the same application requirements.

We consider a setting where the network resources lie on a binary tree where compute resources could be made available on any node, while the edges correspond to communication links carrying traffic between compute nodes and mobile devices, see Figure 4.1. The root of the tree is at height h_c and will be interpreted to correspond to a cloud compute service provider, while the leaves at height 1 will be viewed as edge compute nodes co-located with cellular Base Stations (BS). Meanwhile intermediate levels are introduced to study the potential benefits of placing compute resources between the two extremes, i.e., in the *fog network*.

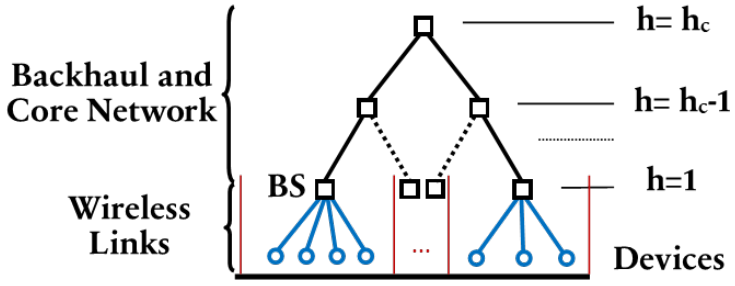


Figure 4.1: Tree Topology Model

We model mobile device service requests as arriving at each BS as a Poisson Point Process (PPP) with intensity λ. Each request corresponds to a server-side process running on a VM hosted in a compute node, and has a random duration of mean μ^{-1} seconds. Hence, if sufficient resources were provisioned the number of active mobile devices at each BS, illustrated in blue in Figure 4.1, follows a Poisson distribution with mean $\eta = \lambda/\mu$ devices, corresponding to the stationary distribution of an $M/GI/\infty$ queue. However, if limited resources are provisioned mobile devices may experience blocking. In

particular, suppose the service provider provisions sufficient compute resources to k simultaneous sessions at each node resources at level h of the tree. Assuming requests at all leaf nodes are served by the parent node at level h, the total offered mean load on such a node will be $\eta 2^{h-1}$ and the blocking probability probability ϵ is given by the Erlang function $E(\eta 2^{h-1}, k)$ associated with an $M/GI/k/k$ queue. The service availability, i.e., $1 - \epsilon$, thus depends not only on the resources provisioned k but also on the level h at which they are located – more on this later.

When a session is active, we assume the device sends updates of size p_u bits at a fixed rate ρ updates/sec. to its associated process, which in turn performs a fixed number of operations ψ^{-1} and may send back an update of size p_d bits to the device – see Figure 4.2. As discussed below, we consider applications where these tasks must be performed in a timely manner.

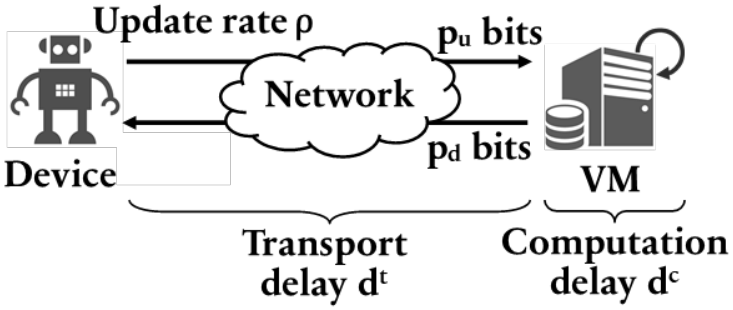


Figure 4.2: Interaction between a device and its server-side process.

We assume for now that the devices supported by the service provider share the same application and their VMs are hosted at the same level in the

tree. Section 4.6 revisits this assumption by examining a general service and network model.

4.4.2 Delay Model

To achieve timely service, three key metrics need to be considered: (1) the device update rate ρ affecting the amount of compute needed and the load on communication links, (2) the transport delay d^t experienced by updates depending on the level the VM is placed; and (3) the compute delay d^c that depending on the amount of compute resources allocated to a VM.

To get at the main characteristics of such systems we shall consider a simple delay model. If compute resources are placed at level h the round-trip transport delay $d^{\mathrm{t}}(h)$ is given by:

$$d^{\mathrm{t}}(h) = 2 \times \left(\frac{p_u + p_d}{l} + \phi h \right) \quad \text{sec.}, \tag{4.1}$$

where l is the bottleneck link capacity, likely the wireless link, and ϕ is a constant forwarding delay per hop to the compute resources. The first term captures the overall transmission delay of a file while the second term captures the forwarding delay. This idealized model assumes that the service providers have access to uncongested links with minimal queuing from their infrastructure provider, i.e., by over-provisioning their communication resources, see [60], or prioritizing such traffic.

Meanwhile, the delay to process an update depends on application specific tasks such as database look-ups, GPS-coordinates processing, video

frames rendering, etc. Given that an update requires ψ^{-1} operations and assuming perfect parallelism, c CPU cores each able to deliver ν operations/sec would complete the update task in

$$d^{\mathrm{c}}(c) = \frac{1}{c\psi\nu} \text{ sec.} \tag{4.2}$$

Note that we will allow c take fractional values.

4.4.3 Timeliness Metric

In this chapter we adopt an end-to-end timeliness metric based on the *Age-of-Information* (AoI), see e.g., [88, 41, 93]. The key difference with traditional end-to-end or round-trip delay, is capturing the difference between the current time (at the mobile device) and the time at which the server has completed processing the last update, i.e., acted upon and possibly delivered back to the mobile device. Hence, the AoI requires factoring both the update rate and compute/communication delays, and captures the tension between these two variables. For instance, if the device's update rate is low, then the remote process may often be out of sync even if the transport and compute delays are low. Conversely, if the update rate is high, but updates experience large delays, the server-side process decisions would be outdated most of the time.

Several works have studied ways of characterizing the AoI in multi-user settings, see e.g., [172, 75, 19]. For the most part, they use variations of Theorem 3 in [172] which captures the AoI for a specific device. For simplicity

we shall use a natural variant of this timeliness metric τ, given by:

$$\tau = \frac{1}{2\rho} + d^{\mathrm{t}}(h) + d^{\mathrm{c}}(c) \tag{4.3}$$

As can be seen, low delays and high update rates improve timeliness. Further support for this performance metric can be found in Appendix C.1. We denote by τ_0 the application specific timeliness constraint, i.e., resources need to be provisioned so as to ensure $\tau \leq \tau_0$ for the devices subscribed to the service.

Putting Equations 4.1 and 4.3 together, one can observe the dependence of the timeliness τ on the device update rate ρ and the level h at which the service provider rents/places its compute resources. It is clear that both $d^{\mathrm{t}}(h)$ and τ increase with h. However, fixing a timeliness constraint τ_0 forces ρ to increase with h to compensate for the additional delay, increasing the compute resources required at the compute node side to process the additional updates. Therefore, one can distinguish two clear tradeoffs, one between the VM level and timeliness, the other between the VM level and compute resources.

4.5 Problem Formulation and Results

The service provisioning problem reduces to jointly determining (1) the optimal level h^* at which to place the VMs, (2) the required number of cores c^* per VM, (3) the minimum number of VMs k^* that can be hosted per compute node, as well as (4) the minimum device update rate ρ^*, that will dictate the amount of traffic, i.e., the communication cost, on the links below level h^*.

4.5.1 Problem Formulation

Given the simple system model proposed in the previous section, the service provider's cost $\mathcal{C}_P$ in the network resource provisioning phase can be approximated as the sum of the communication and compute cost:

$$\mathcal{C}_P(\rho, c, k, h) = \theta^{\mathrm{t}} \eta 2^{h_c - 1} h(p_u + p_d)\rho + \theta^{\mathrm{c}} 2^{h_c - h} kc \tag{4.4}$$

where θ^{t} is the communication cost (in \$/Mbps/hop/link) and θ^{c} is the compute cost (in \$/core). Note that $\eta 2^{h_c-1}$ is the mean number of devices in the network assuming high availability (no blocking) and $h(p_u + p_d)\rho$ is the mean load $\times$ links per device if compute is placed at level h. Meanwhile 2^{h_c-h} is the number of nodes at level h where compute resources are placed and kc is number of cores per node if each VM requires c cores. Now given a timeliness constraint τ_0 and an availability requirement, i.e., blocking probability ϵ, one can characterize the minimum cost level at which to dimension the network in four steps.

First, given Equation 4.3 and the timeliness constraint τ_0 one can determine the smallest feasible update rate, when compute resources are placed at level h, i.e., that with the smallest communication/compute cost. Specifically, to ensure no queuing at the VM we need $d^c \leq \frac{1}{\rho}$ so setting this to equality we get:

$$\rho(h) = \frac{3/2}{\tau_0 - d^{\mathrm{t}}(h)}. \tag{4.5}$$

Second, given the compute delay constraint, the number of CPU cores

allocated per VM can be found from Equation 4.2:

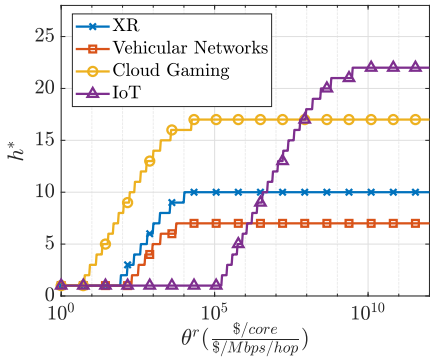
$$c(h) = \frac{\rho(h)}{\psi\nu}. \tag{4.6}$$

Third, the number of VMs $k(h)$ compute nodes at level h would need to support to limit blocking to ϵ can be obtained via the Erlang-B formula, see [17], i.e., solving $\epsilon = E(\eta 2^{h-1}, k(h))$, where $\eta 2^{h-1}$ is the mean load such nodes would see. Note service providers benefit from statistical multiplexing gains when compute resources are placed higher in the tree, allowing increased aggregation of traffic on shared resources. As h increases, the compute load variability per node decreases, hence placing compute at the top of the tree, i.e., in the cloud, reduces the need for slack compute resources in order to ensure high availability, and thus reduces compute costs.

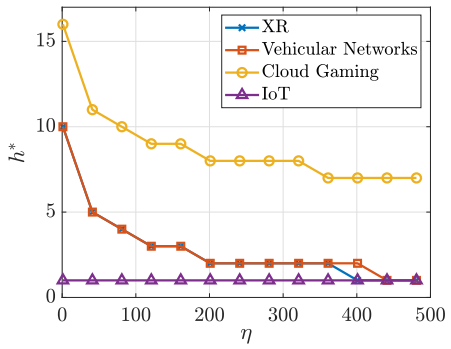
Finally, the optimal service level can be trivially found by evaluating $h^* = \arg\min_{h\in\{1,\cdots,h_c\}} \mathcal{C}_P(\rho(h), c(h), k(h), h)$. We can then obtain $\rho^* = \rho(h^*)$, $c^* = c(h^*)$ and $k^* = k(h^*)$. This is tractable since h_c is reasonably small.

4.5.2 Results Analysis

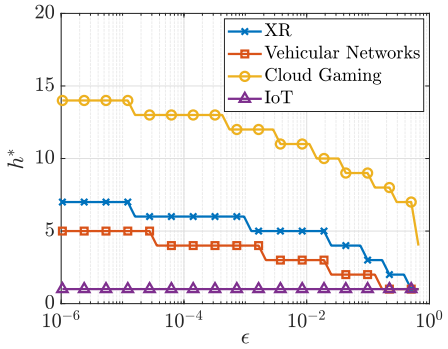
We now present and analyze our results for different application parameters shown in Table 4.1. We analyze successively the effect of the relative costs of compute and communication, density of devices, and service availability constraint.



(a) Effect of the relative cost of compute to communication θ^{r}; $\epsilon = 1 \times 10^{-5}$.



(b) Effect of the number of devices η attached to each BS, $\theta^{\mathrm{r}} = 1 \times 10^3$; $\epsilon = 1 \times 10^{-5}$.



(c) Effect of the blocking probability ϵ, $\theta^{\mathrm{r}} = 1 \times 10^3$.

Figure 4.3: Optimal service level h^*, for $h_c = 25$, $\phi = 100\mu$s/hop, $\nu = 30 \times 10^3$ MIPS/core, $l = 1$ Gbps.

4.5.2.1 Effect of Compute and Communication Costs

Figure 4.3a shows the optimal VM level h^* as a function of the relative cost of compute to communication $\theta^{\mathrm{r}} = \theta^{\mathrm{c}}/\theta^{\mathrm{t}}$. Clearly, as compute becomes more expensive with respect to communication, it is preferable for compute resources to be placed higher up in the tree so as to benefit from statistical multiplexing. It is also worth noticing that the optimal VM placement depends

on the use-case. While the optimal decision for all applications would be to place compute resources as close to the mobile devices as possible, i.e., at the edge, when compute is cheap, each use case has a different behavior as θ^{r} grows. For large θ^{r}, the cost effective placement for each use-case is as far from the edge as possible. The highest level possible for large θ^{r} is dictated by the application's timeliness constraint. More specifically, the looser τ_0, the larger h^* is for large θ^{r}. Any value of h larger than this level would be infeasible, as the transport delay $d^{\mathrm{t}}(h)$ would exceed the timeliness requirement τ_0, for any ρ.

4.5.2.2 Effect of Device Density

As discussed earlier, having multiple VMs share compute resources leads to statistical multiplexing gains. Resource pooling can either be achieved by placing compute resources higher up in the tree, or by increasing the device load per BS. Therefore, as η grows, we expect to need fewer compute resources per unit demand, pulling the optimal service level down closer to the edge. This is indeed what is exhibited in Figure 4.3b, where θ^{r} has been estimated based on realistic compute and communication cost values [8, 23].

4.5.2.3 Effect of Service Availability Requirement

The network is dimensioned so as to guarantee an availability of $1 - \epsilon$. This naturally leads to higher dimensioning cost versus mean load provisioning, as slack resources will need to be allocated to address load variations. In

fact, the smaller ϵ is, the more compute resources will need to be provisioned to ensure the desired service availability level is met. Since more compute resources need to be reserved, the most cost effective strategy is to place compute resources higher in the tree. Figure 4.3c illustrates this trend, showing that one can afford to place the resources closer to the edge under relaxed constraints.

4.6 Online Service Placement of Heterogeneous Traffic in the Fog

So far in this chapter, we have tackled the problem of service placement and dimensioning for real-time applications on mobile devices. The analysis presented in Section 4.5 was based on a simple network topology, delay models and timeliness constraints. These assumptions were necessary to abstract what in practice is quite a complex system. In this section, we explore the design of an algorithm that jointly adapts devices' update rates and VM placement in a heterogeneous network, using delay measurements instead of model-based predictions. We assume a service provider has already provisioned resources on a general network topology and consider the case where mobile devices running different applications are co-hosted on shared resources. We emphasize that the problem addressed in the previous sections is a joint network dimensioning and service placement problem faced by the service provider, while the one presented in this section is a joint network management (through device rate-adaptation) and service placement problem faced by the devices requesting the

service. The former problem is hence faced during the network deployment phase, while the latter is faced when the network is operating.

4.6.1 Network Model and Algorithm Description

In our general network model we let $\mathcal{S}$ denote the set of compute nodes, where each $s \in \mathcal{S}$ has capacity κ_s. These nodes shared by mobile devices of different types where $\mathcal{A}$ denotes the set of types. Requests of Type $a \in \mathcal{A}$ arrive as a PPP of intensity λ_a, and are active for a random time with mean μ_a^{-1} seconds. The types also have potentially different compute requirements per update ψ_a^{-1} and application timeliness requirement τ_a. Note that request types capture devices' requests generated at different locations and associated with different application requirements, whence Type a requests are restricted to be served by a subset of compute nodes $\mathcal{S}_a \subseteq \mathcal{S}$.

In practice, delays experienced by updates might be roughly quasistatic or constant over time, congestion dependent, i.e., depend on previous placement decisions made by the algorithm, or vary due to exogenous traffic which is not under the service provider's control. Hence, in the sequel several metrics including network delays are denoted as depending on time.

Our proposed Algorithm 4.1 extends traditional Least Ratio Routing (LRR) based algorithms, see [6], to realize joint service placement and rate-adaptation along with possibly service migration. Thus it is executed when new mobile devices arrive to the network, but also subsequently if a device moves and/or observes changes in network congestion that warrant the migra-

tion of its VM to another location. As devices execute Algorithm 4.1 more frequently, they will be able to react to more sudden changes in the delay profile, but at the cost of more frequent compute node pings. For simplicity, we focus on the algorithm's behavior upon arrival of a new request.

Algorithm 4.1: Rate-Adaptive Least Ratio Routing

Data: Device Type a, time t
Result: Solves for best $s^* \in \mathcal{S}_a, \rho^*_{a,s^*}(t)$
Ping/measure $d^{\mathrm{t}}_{a,s}(t)$, $r_s(t)$, κ_s, $f_s(\cdot)$, $\forall s \in \mathcal{S}_a$
$\rho_{a,s}(t) = 1.5/(\tau_a - d^{\mathrm{t}}_{a,s}(t))$, $\forall s \in \mathcal{S}_a$
$\Delta_{a,s}(t) = \psi_a^{-1}\rho_{a,s}(t)$, $\forall s \in \mathcal{S}_a$
$u_s(t) = r_s(t)/\kappa_s$, $\forall s \in \mathcal{S}_a$
$u'_{a,s}(t) = (r_s(t) + \Delta_{a,s}(t))/\kappa_s$, $\forall s \in \mathcal{S}_a$
$\tilde{\mathcal{S}}_a = \{s \in \mathcal{S}_a | u'_{a,s}(t) \leq 1\}$
$s^* = \arg\min_{s \in \tilde{\mathcal{S}}_a} \int_{u_s(t)}^{u'_{a,s}(t)} f_s(u)\,\mathrm{d}u$
return $s^*, \rho^*_{a,s^*}(t)$

When a Type a device arrives at time t, it first pings the compute nodes that can potentially host its VM to *estimate* the current transport delays $d^{\mathrm{t}}_{a,s}(t)$ to all $s \in \mathcal{S}_a$. It also gathers the amount of resource $r_s(t)$ currently allocated at s. Given $d^{\mathrm{t}}_{a,s}(t)$, the device can use Equation 4.5 to determine the update rate $\rho_{a,s}(t)$ it would currently require if its VM was instantiated on node s while satisfying its timeliness constraint τ_a. It then deduces its compute requirements $\Delta_{a,s}(t) = \rho_{a,s}(t)\psi_a^{-1}$. The device can then determine the current utilization $u_s(t) = \frac{r_s(t)}{\kappa_s} \in [0, 1)$ and projected utilisation $u'_{a,s}(t) = \frac{r_s(t)+\Delta_{a,s}(t)}{\kappa_s}$ if the VM was placed on $s \in \mathcal{S}_a$. The nodes that can support this request at time t are given by $\tilde{\mathcal{S}}_a = \{s \in \mathcal{S}_a | u'_{a,s}(t) \leq 1\}$. If none are available, the

request is blocked. Otherwise, each compute node has a strictly increasing function $f_s : [0,1] \to \mathbb{R}_+$ which we refer to as Marginal Utilization Cost Function (MUCF) capturing the cost of using an extra compute resource unit at a given utilization. The algorithm greedily places the VM on the feasible node having the smallest marginal cost at time t, defined as the integral of the MUCF from $u_s(t)$ to $u'_{a,s}(t)$.

The MUCF can be designed with different objectives in mind. For instance, a natural objective would be to balance the compute nodes' loads. This strategy ensures that there are as much available resources as possible in all the nodes, which may help reduce the blocking rate. Different MUCFs would attempt to greedily balance the loads on the compute nodes. In fact, any convex function would achieve this goal, and the function convexity would control the extent to which the service provider wants to balance the load, at the cost of risking to consume more compute resources. In light of these observations, being proportionally fair with respect to the available resources among them is a reasonable policy, i.e., greedily maximizing the network-level utility function $\sum_{s \in \mathcal{S}} \log(1 - u_s(t))$ for an arrival at time t. Theorem 4.6.1, proved in Appendix C.3, confirms this objective can be achieved by properly selecting the MUCF.

Theorem 4.6.1 (Proportional Fairness MUCF)**.** *Choosing* $f_s(u) = \frac{1}{1-u}, u \in [0,1)$ *for a compute node* s *is equivalent to greedily maximizing the proportional fairness utility function.*

4.6.2 Algorithm Performance Analysis

In our framework, we aim at minimizing device blockage. Since we are studying a heterogeneous system, where devices running different applications can request service from the same pool of resources, we assign a reward w_a to Type a devices per unit time spent in the network, which can represent, e.g., revenue generated by serving a Type a device.

We observe that this problem reduces to the Multiple Knapsack Problem (MKP) for fixed compute requirements Δ. This problem has been thoroughly studied in the literature and several strategies based on approximation algorithms and heuristics have been proposed to solve it [61]. In Theorem 4.6.1, we propose an MUCF that greedily balances the loads across the compute nodes to solve a variant of the MKP where the item sizes $\Delta_{a,s}$, depend on the knapsacks $s \in \mathcal{S}$.

A natural definition for the cost function of the device-centered problem $\mathcal{C}_D$ in the network operation phase is the expected rate of loss in revenues due to blockage, for a fixed Δ, defined as:

$$\mathcal{C}_D(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) = \sum_{a \in \mathcal{A}} w_a \lambda_a \mu_a^{-1} P(B_a; \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) \tag{4.7}$$

where $P(B_a; \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa})$ captures the probability that a typical Type a device is blocked. $\mathcal{C}_D$ can be lower-bounded by solving a relaxed MKP as stated in Theorem 4.6.2, proved in Appendix C.4.

Theorem 4.6.2 (Lower-Bound on the Rate of Loss in Revenue)**.** *Let A be an assignment matrix representing the mean number of Type a devices assigned to*

node s, such that $A \in \mathcal{B}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) = \{A \in \mathbb{R}^{|\mathcal{A}|\times|\mathcal{S}|} | \sum_{s\in\mathcal{S}} A_{a,s} \leq \lambda_a \mu_a^{-1}, \forall a \in \mathcal{A}\ ,\ \sum_{a\in\mathcal{A}} \Delta_{a,s} A_{a,s} \leq \kappa_s, \forall s \in \mathcal{S}\}$.

Let A^* *be a feasible assignment, solution of the Linear Program relaxed Multiple Knapsack Problem (LP-MKP):*

$$\textit{LP-MKP}(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}): \qquad \max_{A} \sum_{a\in\mathcal{A}} \sum_{s\in\mathcal{S}} w_a A_{a,s}$$
$$s.t.\ A \in \mathcal{B}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa})$$

$$\textit{Then,} \quad \underline{\mathcal{C}}_D(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) = \sum_{a\in\mathcal{A}} w_a(\lambda_a \mu_a^{-1} - \sum_{s\in\mathcal{S}} A^*_{a,s})$$
$$\leq \mathcal{C}_D(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa})$$

The authors in [6] discuss a special case of the suggested framework, where $w_a = 1$, $\mu_a^{-1} = 1, \forall a \in \mathcal{A}$, and $\Delta_{a,s} = 1, \forall (a,s) \in \mathcal{A} \times \mathcal{S}$. In this specific setting, $\mathcal{C}_D$ was proven to converge to $\underline{\mathcal{C}}_D$ in the fluid limit, i.e., when both the arrival rate vector λ and capacity vector κ are scaled by a large fluid-scale factor γ. In this chapter we study whether our proposed local and adaptive LRR policy can asymptotically drive the value of the network-wide cost function to its theoretical lower bound in large systems in more general settings than in [6].

4.6.3 Algorithm Performance Evaluation

We now evaluate the performance of our joint service placement and rate-adaptation algorithm via simulation in the fluid-scaled network with fac-

tor γ. In these simulations, we assumed a more general underlying network model than Figure 4.1, depicted in Figure 4.4.

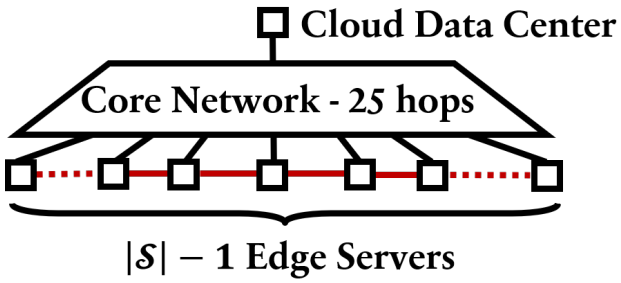


Figure 4.4: General Topology Model

This network features a cloud compute node located 25 hops away from the edge, the cloud compute node having a compute capacity set to be 100 times the one of an edge node. Each of the $|\mathcal{A}|$ device types is attached to a random BS colocated with an edge compute node. A type is a collection of devices of one of the use-cases described in Section 4.3, and having technical requirements given in Table 4.1. Each device can place its service in its closest compute node, or any adjacent node including the cloud at the cost of higher transport delay, as modeled in Equation 4.1. Moreover, Type a's reward w_a is set to be proportional to ψ_a^{-1} depicting a pricing model based on the amount of compute a Type a update requires.

We compare the performance of our joint placement and rate-adaptation policy to a similar placement algorithm, i.e., using the MUCF suggested in Theorem 1, but for devices having static update rates, set such that compute nodes up to five hops away can be reached without violating their timeliness

constraint.

In Figure 4.5, we show that the rate of loss in revenue $\mathcal{C}_D$ converges to the lower-bound $\underline{\mathcal{C}}_D$ as γ increases for the rate-adaptive algorithm, but not for the static-rate one. In this simulation, network delays are assumed to be static, and $\boldsymbol{\kappa}$ has been set so as to ensure that the total capacity in the network is larger than the expected network load.

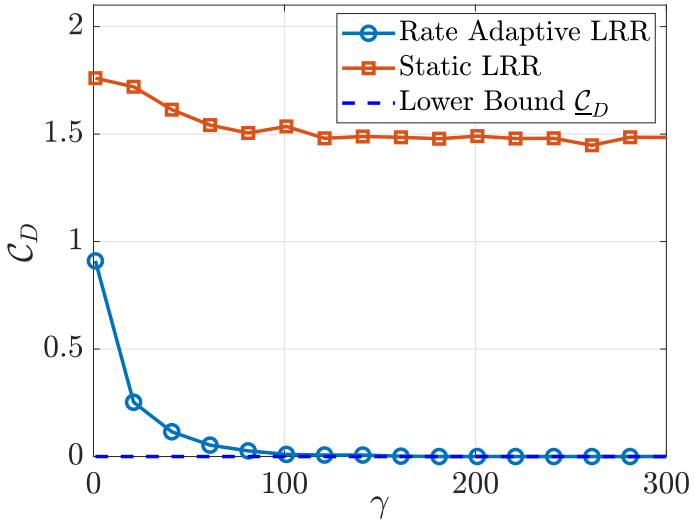


Figure 4.5: Performance comparison of the rate-adaptive and static LRR algorithms in the fluid-limit and the theoretical lower-bound on the mean rate of loss in revenue; $|\mathcal{A}| = 50$, $|\mathcal{S}| = 20$.

We note that the value of the lower-bound is 0, meaning that in large-scale systems a zero-blocking regime can be achieved.

Figure 4.5 may, however, look different if the fixed update rate is designed differently. Figure 4.6 shows the value of $\mathcal{C}_D$ in the fluid limit for the static LRR algorithm as a function of the devices' rates. Note that the figure

only shows the rate of XR devices, but all the devices' rates increase along with ρ_{XR}. We observe that $\mathcal{C}_D$ first drops quickly once the update rate is large enough to reach nodes two hops away, as it gives the devices the ability to balance the load among edge nodes. This effect is not as beneficial for larger rates as the VM compute requirements also increase, leading to a higher blocking rate. The cost function value then dramatically drops once the update rate allows the devices to reach the cloud node and benefit from its substantial capacity, yet without reaching zero-blocking as the rate-adaptive LRR. Finally, larger update rates are associated with larger values of $\mathcal{C}_D$ as the devices' VM requirements keep on increasing while not gaining any load-balancing opportunity.

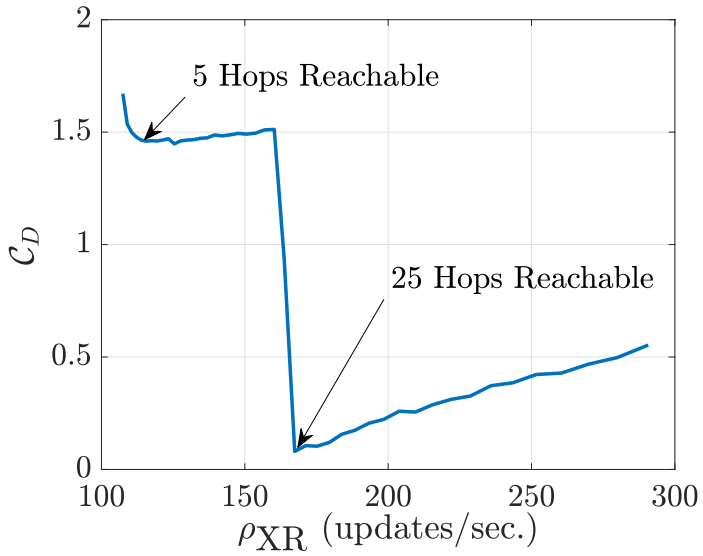


Figure 4.6: Plot of the mean rate of loss in revenue as a function of the devices' fixed update rate; $|\mathcal{A}| = 50$, $|\mathcal{S}| = 20$, $\gamma = 300$.

The key takeaway is that manufacturers can design devices with slow

fixed update rates, requiring little compute resources at the server side and little power at the device side, but at the cost of reducing the set of reachable nodes given the timeliness constraint, hence reducing the placement algorithm's balancing ability. Conversely, for large rates, farther nodes such as powerful cloud servers can be reached leading to better load-balancing, but devices may occupy unnecessary resources if their VMs are placed at the edge, leading to wasted compute resources and reduced service availability. Manufacturers may optimize for an optimal update rate balancing these two effects, but it is unlikely to perform well in arbitrary network topologies.

Rate-adaptation allows for more flexibility in the service placement, without occupying unnecessary compute resources, explaining the better performance of the rate-adaptive LRR algorithm in Figure 4.5 over static rate policies.

Another major strength of Algorithm 4.1 is the fact that it can closely adapt to changes in network delays. Figure 4.7 shows the performance over time of the joint placement and rate-adaptation policy under stochastic delays and compares it to the one of the static algorithm. The delay process experienced by the devices is now assumed to depend on previous decisions taken by the algorithm, whose mean is simulated as an increasing and convex function in the congestion level. In this simulation, the compute node capacities $\boldsymbol{\kappa}$ were slightly under-provisioned to exhibit the policies' performances in a compute-limited settings.

One can clearly observe that the rate-adaptive LLR policy's perfor-

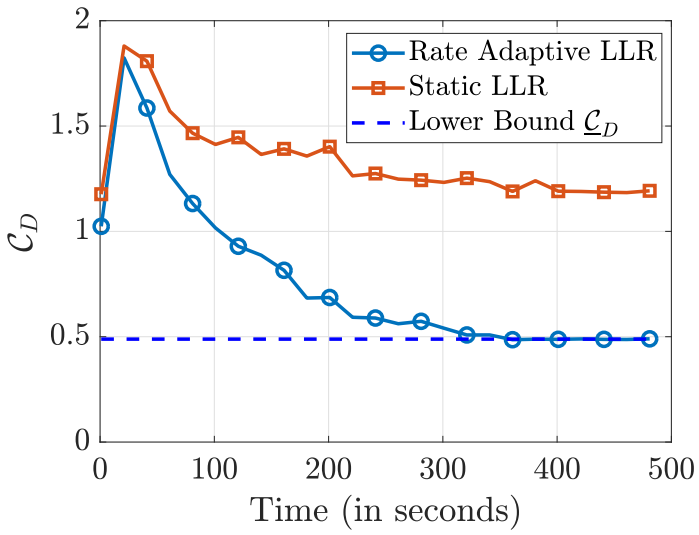


Figure 4.7: Performance comparison of the rate-adaptive and static LRR algorithms, and the theoretical lower-bound on the mean rate of loss in revenue under stochastic network delays; $|\mathcal{A}| = 20$, $|\mathcal{S}| = 10$, $\gamma = 500$.

mance eventually moves very close to the steady-state lower-bound $\underline{\mathcal{C}}_D$, while the static policy does not perform as well. Here again, the rate-adaptive LRR algorithm has the advantage of being able to reach further compute nodes when needed than the static rate algorithm, explaining the gap between the two curves. Moreover, when performing rate adaptation, all the devices currently served in the network can quickly react to changes in the network delay they experience, making use of the available compute nodes' resources more efficiently, hence reducing blocking.

The presented results indicate that rate-adaptation is a requirement and needs to be associated with load-balancing policies to achieve low blocking rates, i.e., high service availability, and serve real-time timeliness constrained

devices.

4.7 Chapter Conclusion

In this chapter, we studied the service placement and dimensioning problem in the fog network. We introduced a simple framework allowing us to identify the most cost-effective VM placement and network resource dimensioning strategies, and understand the fundamental tradeoffs associated with this problem. Unlike results presented in related work, we use the notion of *Age-of-Information* as a timeliness metric, as it demonstrated to be more relevant than network delay when devices send real-time updates. We showcased that different "forces" influence the optimal VM placement and resource dimensioning decisions in the *Cloud-to-Thing continuum*, which may greatly vary from use-case to use-case. We then proposed an online and decentralized joint service placement and rate adaptation policy based on delay measurements. This algorithm is aware of stochasticity in the network delays, ensuring near-optimal availability in large-scale networks by balancing the load on the different compute nodes that can host the devices' service. Our algorithm showed to outperform static rates policies, revealing that rate-adaptation is a requirement in the design of real-time applications.

Chapter 5

Timely Information Sharing in Multiplayer Cloud Gaming Networks

In this chapter[1], we analyze the performance of Multiplayer Cloud Gaming (MCG) systems.

Multiplayer Cloud Gaming (MCG) is emerging as one of the possible future dominant applications. Benefiting from the latest technological advances in communication networks, GPU virtualization, and high-performance computing systems in the cloud and/or edge network, this new *Gaming as a Service* business model is appealing to the three parties involved: gamers, game developers and game service providers (GSPs) [26]. Unlike traditional online gaming frameworks, the game is hosted at a remote server, allowing the players to interact by sending regular updates and receiving a video stream without the need for dedicated hardware or need to own the game license. Multiple platforms are already being commercialized, such as GeForce Now by Nvidia [120], Project xCloud by Xbox [114], Google Stadia [65], Playstation Now [151], and Amazon Luna [9].

[1]Publications based on this chapter: [86] S. Kassir, G. de Veciana, N. Wang, X. Wang, P. Palacharla, Joint Update Rate Adaptation in Multiplayer Cloud-Edge Gaming Services: Spatial Geometry and Performance Tradeoffs. ACM MobiHoc 2021, July 2021.

While such platforms are becoming increasingly popular, many network design questions associated with delivering the best MCG-Quality of Service (MCG-QoS) across players remain unexplored. These include the choice of effective network architectures, resource allocation/provisioning strategies, performance guarantees under stochastic network congestion, or ensuring fairness amongst heterogeneous players. These technical challenges are likely to be exacerbated by the tight performance guarantees required by Extended Reality (XR) enhanced collaborative applications/games [51].

5.1 Related Work

Multiple researchers have proposed approaches to optimize the performance of MCG networks, in a wide variety of settings, e.g., proposing energy-aware solutions [39] or cost-effective resource allocation strategies, Virtual Machine (VM) placement and network architectures subject to QoS constraints [49, 85, 71, 47]. However, this body of work does not place emphasis on the impact that each individual player has on the overall gaming experience. Other studies have focused on the interaction among players. In [33], the authors distinguish the notions of absolute response delay and inter-player delay which in turn allows them to study the fairness among the players, and place the VM resources accordingly. In [62], the authors solve a multi-objective optimization problem to solve the network resource provisioning problem, by minimizing both the worst inter-player delay and the network operating cost. However, none of these works consider the effect of adapting the players' update rates

to improve on the freshness of the game state data received/computed at the server side. This notion of freshness has been studied in the context of Cloud Gaming in [173]. While the game server update rate is fixed, the authors analyze the effect of synchronizing the game server's and the players' phases under stochastic network delays, to minimize the mean *Age-of-Information* at the player side. Unlike this study, this chapter focuses on the player-to-server traffic, characterizing the timeliness of the information processed at the game server side.

5.2 Chapter Contributions and Organization

A key challenge associated with supporting multiplayer games is that they may involve a (possibly large) number of geographically dispersed players increasing the games' exposure to congestion variations across a large set of network regions/resources. This makes the MCG-QoS potentially volatile making it particularly difficult to provide stable guarantees. In this chapter, we address this general challenge through four key contributions.

The first contribution is the introduction of a novel multiplayer game model and MCG-QoS metric which captures the joint impact of the network delays/congestion experienced over different time-scales by the all the players participating in the game.

The second contribution is the development of a measurement-based Joint Multiplayer Rate adaptation Algorithm (JMRA) geared at ensuring that the information transmitted by all the players is delivered and processed in a

timely manner at the game server. We show how players can overcome large network delays through increased update rates to improve the MCG-QoS.

The third contribution lies in showcasing how GSPs can leverage the benefits of JMRA to cost effectively provision network resources so as to guarantee high service coverage. Leveraging tools from multivariate majorization theory, we relate the MCG-QoS to the player's spatial configuration and identify the worst-case geometry for a given "geographical spread". We show how the spread impacts the MCG-QoS, and how GSPs might envision provisioning network resources to meet service coverage requirements, e.g., by exploiting edge computing to deliver services with tight timeliness constraints.

The fourth contribution consists in providing a basis to study the MCG service placement problem using majorization theory. We propose a strategy that can be adopted by GSPs to make the MCG-QoS robust to stochastic variations in the network delays/congestion.

We note that the framework we present in this work is not limited to gaming applications. It is indeed also relevant to the general setting of provisioning real-time collaborative cloud-based services to geographically scattered participants/contributors, e.g., collaborative document editing, source code version control, etc.

The remaining of this chapter is organized as follows. In Section 5.3, we present our system model and MCG-QoS metric. We then introduce and study the properties of the JMRA algorithm towards optimizing the MCG-QoS

in Section 5.4. In Section 5.5, we model and study the impact of the players' geographical locations on the MCG-QoS, and we deduce some approaches that can be used by GSPs to provision the network resources. We then capitalize on the models and performance characterization in Section 5.6 to provide additional insights about service placement strategies to face network delays variability. We conclude the chapter in Section 5.7.

5.3 The MCG System Model

5.3.1 Network Architecture

We consider a multiplayer cloud gaming system composed of three entities, consistent with MCG network architectures studied in the literature, see [101, 47, 62]:

1. A set $\mathcal{P}$ of n players in a geographic configuration $\mathbf{x} = (\mathbf{x}_i, i \in \mathcal{P})$, where $\mathbf{x}_i \in \mathbb{R}^2$ corresponds to the location of player i.

2. A Game-server (G-server) running on a VM in a compute node at location $\mathbf{g} \in \mathbb{R}^2$ that hosts the game, i.e., that keeps track of the state of the game by receiving and processing updates from the players.

3. Rendering servers (R-servers) that receive aggregate state information from the G-server, render the video feed and stream it to the players. R-servers are typically placed closer to the players than the G-server to reduce network congestion, but they can also be colocated in the same datacenter.

5.3.2 Network Delay Variation Model

We model network delay variations happening on two different time scales: (1) slow time-scale variations, happening on the order of seconds or minutes, modeling overall network congestion level, and (2) fast time-scale packet delay variations, happening on the order of milliseconds or microseconds, modeling jitter and instantaneous bottlenecks in the network. In this work, we investigate the effect of slow time-scale variations due to network congestion, while abstracting out the fast time-scale variations, using a point estimate, e.g., the mean, median, or 90^{th} percentile of this stochastic process over a small time window, and over which the slow time-scale delay variations are assumed to be constant. Hence, this point statistic is itself slowly varying over time. We model the slow-variations of this point statistic due to network congestion delays between player i and the G-server via a random variable D_i^t. In the sequel, we loosely refer to it as the *typical transport delay* experienced by player i. In addition, we assume that the impact of the players' updates on the network congestion they see is negligible.

5.3.3 Game Operation Model

The game's operation model, depicted in Figure 5.1, can be summarized in three stages.

1. Player i sends periodic updates at a rate ρ_i updates per second to the G-server, containing instructions from the game controller (e.g., character's

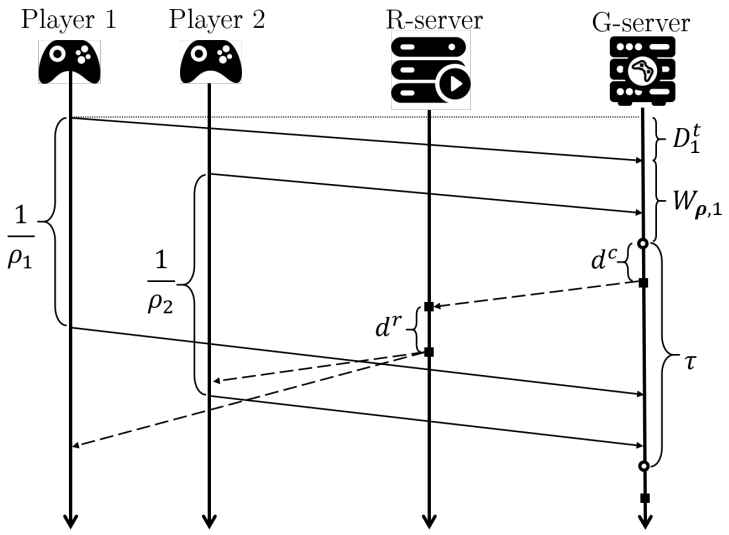


Figure 5.1: Figure of the game operation timeline

movement, camera/headset rotation) and experiences a typical transport delay D_i^t seconds.

2. The G-server, in turn, forms periodic update batches by aggregating and processing everything it has received from the players every τ seconds. This model can be characterized by two types of delays (1) the individual random waiting time $W_{\boldsymbol{\rho},i}$ capturing the time between the latest update received from player i and the beginning of a typical compute cycle, and (2) a shared batch compute delay to process the updates aggregated in a batch, denoted by d^c. We model this batch compute delay $d^c(s)$ as a deterministic differentiable convex function of the game's overall update load $s = \sum_j \rho_j$ at the G-server side. While the load is stochastic (as it depends on the relative phases of the players' update cycles and

the server's compute cycles), we assume that the load variations are negligible, and s can be seen as the mean batch load. In addition, we impose the constraint $d^c(s) \leq \tau$ to ensure that the G-server completes the processing of a batch before the end of a compute cycle of period τ, see Figure 5.1.

3. Once the aggregated game's state is computed, the G-server sends it to the R-servers, that will render the video feed in d^r seconds and stream it back to the associated players.

In this chapter we focus on the timeliness of the player to G-server traffic, leaving the analysis of the round-trip loop as future work.

5.3.4 Game Timeliness Model

In our framework, we seek to ensure that the state of the game is updated in a timely fashion. We use as our timeliness metric the *age of the game* $A_{\mathbf{d}^t,\boldsymbol{\rho}}$, conditioned on a given delay vector $\mathbf{d}^t$ (a realization of the random vector $\mathbf{D}^t = (D_i^t, \forall i \in \mathcal{P})$) and a choice of update rate vector $\boldsymbol{\rho} = (\rho_i, \forall i \in \mathcal{P})$, and defined as:

$$A_{\mathbf{d}^t,\boldsymbol{\rho}} = \max_{i=1,\ldots,n} [d_i^t + W_{\boldsymbol{\rho},i}] + d^c(\sum_{j=1}^{n} \rho_j) \tag{5.1}$$

where $W_{\boldsymbol{\rho},i} \sim \text{Uniform}([0, \frac{1}{\rho_i}]), \forall i \in \mathcal{P}$ as we assume that the transmissions are asynchronous with the G-server's batch processing regular schedule, hence a typical compute batch starts to get processed at a uniform random time in an interval of $1/\rho_i$ seconds.

In this formulation, the age of each player's updates processed at the G-server is the sum of the transport delay, waiting time at the G-server, and batch compute delay. The age of the game is then modeled as the maximum age of each player's updates to capture the fact that stragglers considerably impact the quality of experience of all the other players interacting on a common virtual space. This formulation also hints at the need to ensure that network resources are "fairly" shared among the players involved in the game. This model also recognizes that the game's age depends jointly on the players' update rates and the network delays. As the update rates can be controlled, we identify a basic tradeoff involving this decision. While larger update rates lead to smaller waiting times, and help reduce the age of the game's state, this also results in larger computation delays, increasing the game's age.

5.3.5 The JMRA Problem

Our objective is to solve the Joint Multiplayer Rate Adaptation (JMRA) problem that ensures a small game age, defined as follows:

Problem 5.3.1 (Joint Multiplayer Rate Adaptation)**.** *Given the current transport delay vector* $\mathbf{D^t} = \mathbf{d^t}$*, and a desired age level* a_0*, the JMRA problem consists in finding the update rate vector that maximizes the probability that the age of the game does not exceed some desired level* a_0*, i.e., solving:*

$$\begin{aligned}\boldsymbol{\rho}^*(\mathbf{d}^t) &= \arg\max_{\boldsymbol{\rho}} \left\{ \mathbb{P}(A_{\mathbf{d}^t,\boldsymbol{\rho}} \leq a_0) \ : \ d^c(\sum_j \rho_j) \leq \tau \right\} \\ &= \arg\max_{\boldsymbol{\rho}} \mathbb{P}(\max_i [d_i^t + W_{\boldsymbol{\rho},i}] + d^c(\sum_j \rho_j) \leq a_0)\end{aligned} \tag{5.2}$$

$$s.t.\ d^c(\sum_j \rho_j) \leq \tau \tag{5.3}$$

$$= \arg\max_{\boldsymbol{\rho}} \prod_{i=1}^{n} \mathbb{P}(W_{\boldsymbol{\rho},i} \leq a_0 - d_i^t - d^c(\sum_j \rho_j))$$

$$s.t.\ d^c(\sum_j \rho_j) \leq \tau \tag{5.4}$$

$$= \arg\max_{\boldsymbol{\rho}} \sum_{i=1}^{n} \left[\log\left(\rho_i \cdot (a_0 - d_i^t - d^c(\sum_j \rho_j)) \right) \right]_-$$

$$s.t.\ d^c(\sum_j \rho_j) \leq \tau \tag{5.5}$$

where we define $x_- = \min[x, 0]$. We can now define the MCG Quality of Service as follows:

Definition 5.3.2 (MCG-QoS)**.** *For a given transport delay vector* $\mathbf{d}^t \in \mathbb{R}_+^n$*, we define the MCG-QoS* $q(\mathbf{d}^t)$ *as:*

$$q(\mathbf{d}^t) = \begin{cases} \mathbb{P}(A_{\mathbf{d}^t, \boldsymbol{\rho}^*(\mathbf{d}^t)} \leq a_0), & \text{if } \boldsymbol{\rho}^*(\mathbf{d}^t) \text{ exists,} \\ 0, & \text{otherwise,} \end{cases} \tag{5.6}$$

i.e., the probability that the age constraint is met under JMRA.

In the sequel, we will present an efficient algorithm that solves for $\boldsymbol{\rho}^*(\mathbf{d}^t)$, hence computing $q(\mathbf{d}^t)$, under slowly varying delays.

5.4 The Rate Adaptation Algorithm

In this section, we derive the Joint Multiplayer Rate Adaptation algorithm (JMRA), which jointly uses measured network delays to solve the JMRA problem described in Problem 5.3.1. We then study some of its properties, before evaluating its performance.

5.4.1 Algorithm Description

We envision an algorithm to be executed periodically at the G-server side. At each iteration, the G-server characterizes the transport delays to each player in the game. This can be done, for instance, by estimating the distribution of the (fast time-scale) packet delays experienced by previous updates in a sliding window, and computing the desired point estimate, e.g., the mean, median or 90^{th} percentile. We assume that the players and G-server can measure time using synchronized clocks, hence the packet delays can be estimated with reasonable accuracy. Based on this information, the G-server re-optimizes the players' update rates accordingly, hence adapting to slow-time variations in the delays/congestion. Naturally, the frequency of execution of the algorithm would depend on the time scale for (slow) variations in the network delays' statistics. We describe below the procedure to optimize for the players' update rates given the vector of measured network delays $\mathbf{d}^t$.

Observe that Problem 5.3.1 is convex but it has a non-differentiable cost function leading to slow convergence of a numerical solver. We propose an alternative algorithm to compute the optimal update rate vector, by decomposing the optimization problem into a set of simpler (smooth) sub-problems. In addition, this analysis allows us to extract some basic properties of the JMRA policy.

Our approach is to first consider the case where the aggregate game server load $s = \sum_j \rho_j$ is fixed and known. Under this assumption, one can use the Lagrange multiplier procedure to solve for the optimal $\boldsymbol{\rho}^*(\mathbf{d}^t, s)$ in

Problem 5.3.1 as a function of s. We find:

$$\rho_i^*(\mathbf{d}^t, s) = \min\Big[\frac{1}{\mu(\mathbf{d}^t, s)}, \frac{1}{a_0 - d_i^t - d^c(s)}\Big], \forall i \quad (5.7)$$
$$\text{where } \mu(\mathbf{d}^t, s) \text{ is s.t. } \sum_j \rho_j^*(\mathbf{d}^t, s) = s.$$

Observe that for any delay vector $\mathbf{d}^t$, and any choice of the sum-rate s, two types of players can be distinguished. We have on one hand a set $\mathcal{S}(\mathbf{d}^t, s)$ of *support players*, that will all pick the same update rate of $\frac{1}{\mu(\mathbf{d}^t, s)}$. These players see a transport delay to the G-server that is too large to be able to fully adapt their update rate to compensate for the large transport delays. We have from Equation 5.7, player $i \in \mathcal{S}(\mathbf{d}^t, s)$ if $d_i^t \geq a_0 - \mu(\mathbf{d}^t, s) - d^c(s)$. We have on the other hand a set $\overline{\mathcal{S}}(\mathbf{d}^t, s)$ of *non-support players* that can flexibly trade-off delay for update rate, where $i \in \overline{\mathcal{S}}(\mathbf{d}^t, s)$ if $d_i^t < a_0 - \mu(\mathbf{d}^t, s) - d^c(s)$. The players in this set do not impact directly the MCG-QoS, except by contributing to the total server congestion, as their respective terms in the sum vanish after substituting $\rho_i^*(\mathbf{d}^t, s)$. Equipped with the notion of support set, we can state the following property:

Property 5.4.1 (Support Set Monotonicity in Transport Delays)**.** *For a given delay vector $\mathbf{d}^t \in \mathbb{R}_+^n$, and sum update rate $s \in \mathbb{R}_+$, if $d_i^t \geq d_j^t$, then $j \in \mathcal{S}(\mathbf{d}^t, s) \implies i \in \mathcal{S}(\mathbf{d}^t, s)$. Thus, there are only $n+1$ possible support sets.*

We now use this property along with the solution of Equation 5.7 to solve for the optimal load s. After substituting $\boldsymbol{\rho}^*(\mathbf{d}^t, s)$, Problem 5.3.1 reduces

to solving the one dimensional problem:

$$s^*(\mathbf{d}^t) = \arg\max_s \sum_{i=1}^{n} [\log \frac{a_0 - d_i^t - d^c(s)}{\mu(\mathbf{d}^t, s)}]_- \quad (5.8)$$
$$\text{s.t.} \begin{cases} \sum_{i=1}^{n} \min(\frac{1}{\mu(\mathbf{d}^t,s)}, \frac{1}{a_0 - d_i^t - d^c(s)}) = s \\ d^c(s) \leq \tau \end{cases}$$

This problem is still not easily solved as the cost function is non-differentiable and the constraint set is non-convex. However, by integrating more information about the support set, one can simplify this problem further. Invoking Property 5.4.1, we can distinguish $n+1$ cases corresponding to the $n+1$ possible support sets. Let $\mathcal{S}_m(\mathbf{d}^t)$ be the possible support set that contains m players with the largest measured delays, for $0 \leq m \leq n$. Each sub-problem reduces to solving for the optimal game server load $s_m^*(\mathbf{d}^t)$ assuming that $\mathcal{S}_m(\mathbf{d}^t)$ is the support set. For each sub-problem m, the Lagrange multiplier $\mu(s, \mathbf{d}^t)$ becomes $\mu_m(\mathbf{d}^t, s)$ and is dictated by the first constraint in Equation 5.8. Without loss of generality, we index the players in descending order of transport delays, and we get that:

$$\mu_m(\mathbf{d}^t, s)^{-1} = \frac{1}{m} \left(s - \sum_{i=m+1}^{n} \frac{1}{a_0 - d_i^t - d^c(s)} \right) \quad (5.9)$$

After substituting this constraint into the cost function of the m^{th} sub-problem $\mathcal{U}_m(\mathbf{d}^t, s)$, the latter reduces to:

$$\mathcal{U}_m(\mathbf{d}^t, s) = \sum_{i=1}^{m} \log(a_0 - d_i^t - d^c(s)) \quad (5.10)$$
$$+ m \cdot \log(s - \sum_{i=m+1}^{n} \frac{1}{a_0 - d_i^t - d^c(s)}) - m \cdot \log(m)$$

and the new sub-problem becomes:

$$s_m^*(\mathbf{d}^t) = \arg\max_s \left\{ \mathcal{U}_m(\mathbf{d}^t, s) : \mu_m(\mathbf{d}^t, s) > 0,\ d^c(s) \leq \tau \right\} \tag{5.11}$$

where $\mathcal{U}_m(\mathbf{d}^t, s)$ is smooth and concave in s over the range of values where $\mu_m(\mathbf{d}^t, s) > 0$ and the constraint set is convex. We note that the range of s values where $\mu_m(\mathbf{d}^t, s) > 0$ can be found by solving for the roots of the non-linear Equation 5.9. Hence, the solution to this problem can be found using a convex optimization solver, e.g., using Gradient Ascent, along with a non-linear equation solver. Finally, the optimal $s^*(\mathbf{d}^t)$ is such that $s^*(\mathbf{d}^t) = s_{m^*}(\mathbf{d}^t)$ where $m^* = \arg\max_m \mathcal{U}_m(\mathbf{d}^t, s_m^*)$. We summarize JMRA in Algorithm 5.1.

Algorithm 5.1: Joint Multiplayer Rate Adaptation (JMRA)

Result: Solves for $\boldsymbol{\rho}^*(\mathbf{d}^t)$

Estimate $d_i^t,\ \forall i$;
Solve $s_0 = \sum_{i=1}^{n} \frac{1}{a_0 - d_i^t - d^c(s_0)}$;
if s_0 *exists AND* $d^c(s_0) < \tau$ **then**
    $\rho_i^* = \frac{1}{a_0 - d_i^t - d^c(s_0)}, \forall i$;
else
    for *m=1 to n* **do**
        Find range of feasible s s.t. $\mu_m(\mathbf{d}^t, s) > 0$;
        Solve for $s_m^*(\mathbf{d}^t)$ in Eq. 5.11 using Gradient Ascent;
    end
    Pick $s^*(\mathbf{d}^t) = s_{m^*}(\mathbf{d}^t)$, $m^* = \arg\max_m \mathcal{U}_m(\mathbf{d}^t, s_m^*(\mathbf{d}^t))$;
    Compute $\boldsymbol{\rho}^*(\mathbf{d}^t) = \boldsymbol{\rho}^*(\mathbf{d}^t, s^*(\mathbf{d}^t))$ using Equation 5.7
end

5.4.2 Algorithm Analysis

In Section 5.4.1, we proposed and established the optimality of the JMRA algorithm. We now study some interesting properties thereof, and compare its performance to a baseline where all the players share the same update rate. We start by stating a theorem that relates the players' update rates to their delays to the G-server.

Proposition 5.4.2 (Rate-Proximity Tradeoff)**.** *Given a delay vector* $\mathbf{d}^t \in \mathbb{R}_+^n$ *and any players* $i, j \in \mathcal{P}$*, if* $d_i^t \geq d_j^t$ *then* $\rho_i^*(\mathbf{d}^t) \geq \rho_j^*(\mathbf{d}^t)$.

This proposition follows directly from Equation 5.7 and Property 5.4.1. Intuitively, this property states that players experiencing large transport delays, can compensate for this by increasing the rate at which they send information to the game server. While a unilateral increase in a player's update rate increases the G-server load, and hence all the players' compute delays, the algorithm finds the appropriate congestion level for the given player's configuration. An alternate interpretation would be that the players that experience the smallest delays are willing to reduce their update rates to allow players with larger transport delays to benefit from increased communication/compute resources.

We now compare the performance of the JMRA algorithm with one that distributes equal communication/compute resources among the players. We introduce the Best Static Rate algorithm (BSR), that solves Problem 5.3.1 subject to the additional constraint that all the players' update rates are the

same. We note that this algorithm outperforms any algorithm that uses a fixed update rate imposed by the game designer, as the static update rate is still optimized based on the delay vector. Hence, the reported performance of the BSR algorithm should be viewed as as an upper-bound on what can be achieved when players used fixed update rates. We compare the two algorithms using the following performance metric:

Definition 5.4.3 ($\boldsymbol{\epsilon}$-Playable Games)**.** *A game is said to be ϵ-playable for a given transport delay vector $\mathbf{d}^t \in \mathbb{R}^n_+$, and $\epsilon \in [0,1]$, if its MCG-QoS function satisfies the condition $q(\mathbf{d}^t) > 1 - \epsilon$.*

To compare the performance of the JMRA and BSR update rate selection algorithms, we characterize the probability of a game being ϵ-playable under randomly generated player configurations. Towards evaluating the impact of the game's spatial geometry, i.e., the relative placement of the players and the G-server, we consider a setup where n players are placed randomly and uniformly in a disk of radius r meters forming a player configuration $\mathbf{X} \in \mathbb{R}^{n\times 2}$, while the G-server is placed in the center of the disk assumed to be the origin, i.e., $\mathbf{g} = \mathbf{0}$. In addition, we model the network delay experienced by player i to simply be a deterministic increasing linear function of its distance to the G-server, hence:

$$d_i^t(\mathbf{x}, \mathbf{g}) = \beta_1 + \beta_2 \cdot \|\mathbf{x}_i - \mathbf{g}\|_2 \tag{5.12}$$

where β_1 captures the communication delay components that are independent of the distance between the player and the G-server, e.g., contention delay on

the wireless interface, and β_2 is a coefficient inversely proportional to the speed of light in fiber. We pick values of $\beta_1 = 3 \times 10^{-3}$ seconds and $\beta_2 = 1 \times 10^{-8}$ seconds/meter, consistently with empirical studies in WANs, see [82].

We also assume a specific model for the batch compute delay d^c:

$$d^c(s) = \frac{s \cdot \tau}{n \cdot k_G} \tag{5.13}$$

where k_G corresponds to the compute capacity reserved per player resulting from the compute resources allocated to the G-server's VM. This functional form is motivated by the fact that the expected number of updates received in a batch, i.e., in a window of τ seconds, is $s \cdot \tau$ updates, while the overall compute capacity allocated for the G-server's VM is such that it can process $n \cdot k_G$ updates/second.

Equipped with these models, we compare in Figure 5.2 the likelihood that a randomly generated game configuration is ϵ-playable, under the JMRA and BSR algorithms.

The figure clearly shows that JMRA outperforms the BSR algorithm, as a random player configuration is more likely to lead to a playable game configuration. The performance gap is most significant for small ϵ, i.e., in the regime we expect to operate. For instance, if a GSP seeks to offer high-quality games by picking $\epsilon = 0$, then around 79% of the randomly generated game configurations can be supported under JMRA, while none can be supported under BSR. The main takeaway of these results lies in the observation that allowing flexibility in the choice of the players' update rates considerably

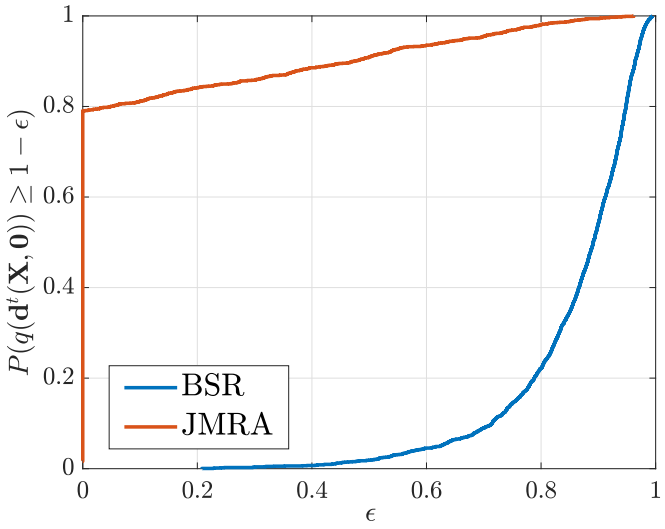


Figure 5.2: Comparison of the probability of the game being ϵ-playable under JMRA and BSR algorithms as a function of ϵ, under random players' configurations; $n = 20$, $r = 3 \times 10^6$m, $a_0 = 50$ms, $\tau = 20$ms, $k_G = 150$ updates/s/player.

improves the MCG-QoS, or equivalently, allows the GSPs to deploy reduced network resources for a desired MCG-QoS.

5.5 Service Coverage Analysis and Network Resource Provisioning

So far we have presented an efficient algorithm to maximize the MCG-QoS through joint multiplayer rate adaptation for a given network congestion regime as captured by the network delay vector. A GSP will, however, want to guarantee that newly instantiated games are playable for configurations that are not "too spread-out" and/or experiencing network congestion outliers. In

this section, we further introduce a model tying geometry to network congestion and analyze the large-scale compute resources a GSP would need to deploy (e.g., rent from spatially distributed cloud service providers) so as to ensure that MCGs with a given geographical spread will be playable under JMRA – i.e., ensure MCG service coverage.

5.5.1 Linking Players' Spatial Geometry to Network Congestion

To capture the relationship between players' configuration geometry and large-scale network congestion, we model the *typical* transport delay vector $\mathbf{D}^t$ experienced by the players, via a random vector $\mathbf{D}^t_{\boldsymbol{\delta}} = (D^t_{\boldsymbol{\delta},i}, i \in \mathcal{P})$. It is parametrized by a distance vector $\boldsymbol{\delta} = (\delta_i, i \in \mathcal{P})$, where δ_i is the distance from player i to the G-server and $D^t_{\boldsymbol{\delta},i}$ models the typical slow variations in transport delay experienced by player i a distance δ_i from the G-server.

Remark. We emphasize that this model is not intended to capture the specific characteristics of congestion/delay variations as seen at a particular location, but instead what would be typically experienced by players in a large scale network to enable a study of the large-scale resources the GSP would require.

Assumption 5.5.1 (Network Congestion Model)**.** *We assume for any player configuration with associated distance vector* $\boldsymbol{\delta} \in \mathbb{R}^n$ *that* $\mathbf{D}^t \sim \mathbf{D}^t_{\boldsymbol{\delta}}$ *where the entries of* $\mathbf{D}^t_{\boldsymbol{\delta}}$ *are mutually independent and furthermore for* $\mathbf{z} \in \mathbb{R}^n_+$ *and* $i \in \mathcal{P}$ *we have that*[2] $\mathbb{E}[D^t_{\boldsymbol{\delta},i}] \leq \mathbb{E}[D^t_{\boldsymbol{\delta}+\mathbf{z},i}]$ *and* $\mathbf{D}^t_{\boldsymbol{\delta}} \leq^{icx} \mathbf{D}^t_{\boldsymbol{\delta}+\mathbf{z}}$.

[2]As in[142], we define *increasing convex dominance* as $\mathbf{X} \leq^{icx} \mathbf{Y} \iff \mathbb{E}[\phi(\mathbf{X})] \leq \mathbb{E}[\phi(\mathbf{Y})]$, for all increasing convex $\phi : \mathbb{R}^n \to \mathbb{R}$. See also Appendix A.1.

Intuitively players that are further away from the G-server would on average experience higher transport delays, e.g., due to increased propagation delay and number of hops traversed. Perhaps more importantly with higher distances one might expect a higher variability due to congestion on intervening network resources. The multivariate *increasing convex (icx) ordering*, assumed above, see [142], partially captures an ordering in delay "variability" with distance.

As previously mentioned, the distribution of $\mathbf{D}^t_{\boldsymbol{\delta}}$ reflects slowly varying network congestion an MCG game will need to overcome through rate adaptation. For simplicity, the GSP might provision network resources respect to a point estimate of the transport delay distributions, e.g., the mean, median or 90^{th} percentile of the delay distribution for each user, depending on the GSP's risk tolerance. In Section 5.6, we discuss how the risk of under-provisioning the network due to variability in the network delays/congestion can be curtailed by the GSP during network operation. We denote as $\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{g})$ the transport delay vector associated with the desired network delay statistic, as a function of the player configuration $\mathbf{x}$ and the G-server location $\mathbf{g}$. For instance, $\bar{d}^t_i(\mathbf{x}, \mathbf{g}) = \mathbb{E}[D^t_{\boldsymbol{\delta}(\mathbf{x},\mathbf{g}),i}]$, where $\delta_i(\mathbf{x}, \mathbf{g}) = \|\mathbf{x}_i - \mathbf{g}\|_2$, if the GSP decides to provision the network resources for the mean transport delays.

5.5.2 Characterization of Geographical Spread

We characterize the players' geographical spread as follows:

Definition 5.5.2 (Geographical Spread)**.** *Let* $\mathbf{x} \in \mathbb{R}^{n\times 2}$ *be an n-player con-*

figuration, we define the configuration's geographical spread $\sigma(\mathbf{x})$ *as the radius of the smallest disk containing all the players, i.e.,:*

$$\sigma(\mathbf{x}) = \min_{r\in\mathbb{R}_+,\mathbf{c}\in\mathbb{R}^2} \left\{ r\colon \ \|\mathbf{c} - \mathbf{x}_i\|_2 \leq r,\ \forall i \right\} \tag{5.14}$$

Clearly, the larger $\sigma(\mathbf{x})$ is, the more "spread out" the players are, and hence, the harder it will be to find a server location that can ensure the game is ϵ-playable. We also introduce the notion of a regular configuration, which will be useful to characterize the class of configurations with n players with a given geographical spread.

Definition 5.5.3 (Regular Configuration)**.** *An* n *player configuration of radius* r $\boldsymbol{\omega}(n, r) \in \mathbb{R}^{n\times 2}$ *is said to be regular iff all the players are equispaced on a circle of radius* r*.*

For instance, the configuration $\boldsymbol{\omega}(n, r)$ such that $\boldsymbol{\omega}_i(n, r) = (r{\cdot}\cos\frac{2\pi(i-1)}{n}, r{\cdot}\sin\frac{2\pi(i-1)}{n}), 1 \leq i \leq n$ is a regular configuration.

5.5.3 Service Coverage Analysis

Next we focus our attention on how the players' geographical spread impacts the service coverage and the GSP's network resource provisioning strategies. We assume GSPs whose goal is to ensure that games involving players with a given geographical spread will find a G-server (with high probability) such that the game is ϵ-playable. To that end, a GSP can control the density of compute nodes, as well as the compute capacity allocated to the

G-servers' VMs. We study how these decisions are coupled and impacted by the target games' geographical spread to be supported.

We shall define first a useful service coverage metric that we adopt in this framework, as a function of the delay vector $\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{g})$.

Definition 5.5.4 (ϵ-Feasible Region)**.** *For a given player configuration* $\mathbf{x} \in \mathbb{R}^{n\times 2}$*, we define the* ϵ*-feasible region* $\mathcal{F}_\epsilon(\mathbf{x})$ *to include all G-server locations for which the game would be* ϵ*-playable as:*

$$\mathcal{F}_\epsilon(\mathbf{x}) = \{\mathbf{g} \in \mathbb{R}^2 : q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{g})) > 1 - \epsilon\}, \tag{5.15}$$

Moreover, the area of region $\mathcal{F}_\epsilon(\mathbf{x})$ can be expressed as:

$$|\mathcal{F}_\epsilon(\mathbf{x})| = \iint_{\mathbb{R}^2} \mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{g})) > 1 - \epsilon\} d\mathbf{g} \tag{5.16}$$

Large feasible areas $|\mathcal{F}_\epsilon(\mathbf{x})|$ are preferred as they are associated with large likelihood to find a G-server that can support the MCG service given the player configuration $\mathbf{x}$, see Figure 5.3.

To formally characterize the network provisioning problem, we model the compute nodes to be deployed according to a homogeneous spatial Poisson Point Process (PPP) $\Phi(\lambda)$ of intensity λ compute nodes/m^2. The GSP aims to provision the network resources so as to ensure an $(n, \sigma, \epsilon, \alpha, \nu)$-service coverage, defined as:

Definition 5.5.5 ($(n, \sigma, \epsilon, \alpha, \nu)$-Service Coverage)**.** *An MCG network is said to ensure an* $(n, \sigma, \epsilon, \alpha, \nu)$*-service coverage, for* $n \in \mathbb{N}, \sigma \in \mathbb{R}_+, \epsilon \in [0, 1], \alpha \in$

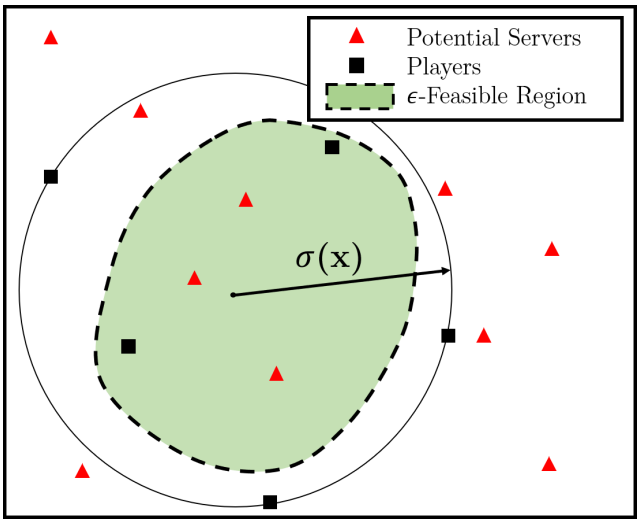


Figure 5.3: Example of an ϵ-feasible region, with $n = 5$ players.

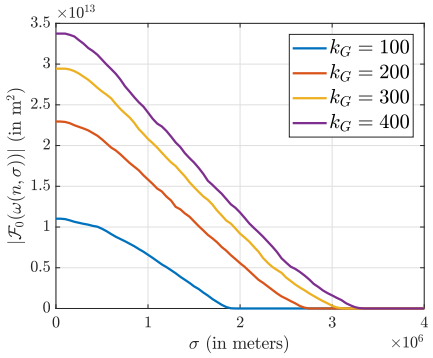
$[0, 1], \nu \in \mathbb{N}$, *if for any* n*-player configuration* $\mathbf{x}$ *with a geographical spread less than* σ*:*

$$\mathbb{P}(|\Phi \cap \mathcal{F}_\epsilon(\mathbf{x})| \geq \nu) \geq 1 - \alpha, \tag{5.17}$$

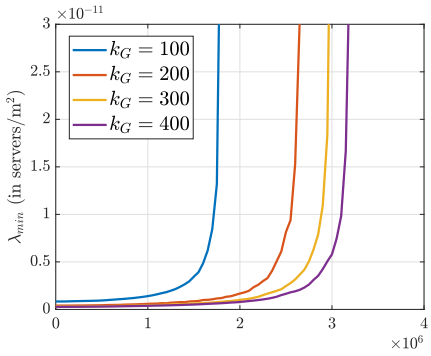
i.e., if the probability that a randomly located game involving n *players with a geographical spread less than* σ *finds at least* ν *compute nodes that would make the game* ϵ*-feasible exceeds* $1 - \alpha$*.*

More formally, the GSP wishes to solve the network resource provisioning problem described as follows:

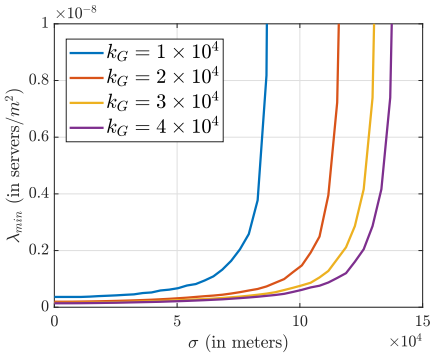
Problem 5.5.6 (Network Resource Provisioning)**.** *The network resource provisioning problem consists in finding the smallest density of compute nodes* λ *guaranteeing an* $(n, \sigma, \epsilon, \alpha, \nu)$*-service coverage.*



(a) Area of the ϵ-feasible region vs. the players' geographical spread σ; $n = 20$, $a_0 = 50$ms, $\tau = 20$ms, $\epsilon = 0$.



(b) Smallest feasible server density $\lambda_{\min}$ vs. the players' geographical spread σ; $n = 20$, $a_0 = 50$ms, $\tau = 20$ms, $\epsilon = 0$, $\alpha = 1 \times 10^{-4}$.



(c) Smallest feasible server density $\lambda_{\min}$ vs. the players' geographical spread σ; $n = 20$, $a_0 = 5$ms, $\tau = 3$ms, $\epsilon = 0$, $\alpha = 1 \times 10^{-4}$.

Figure 5.4: Figures of the impact of the geographical spread σ and the per-user compute capacity k_G on the area of the ϵ-feasible region of an n player regular configuration, and the induced minimum density $\lambda_{\min}$ of compute nodes required to guarantee $(n, \sigma, \epsilon, \alpha, 1)$-service coverage, in traditional MCG and XR-MCG settings. For scale comparison, the distance from New York City, NY to Los Angeles, CA is on the order of 4×10^6 meters, while the area of the USA is on the order of 1×10^{13} square meters.

We note that while a single compute node falling inside an ϵ-feasible region may be enough to guarantee that the game is ϵ-playable, the GSPs may want to provision the network resources so as to ensure that multiple compute node ($\nu > 1$) fall within the region for three reasons. First, such a choice increases load-balancing flexibility across compute nodes in the network, which has been shown to improve the service availability in large networks [85]. Second, it provides additional guarantees in case some feasible servers are unable to support additional MCG instances due to limited resources. Third, it improves the robustness to variability in the transport delays experienced by the players, that may trigger costly migrations of the G-server's VM, more on this in Section 5.6.

We now state an important result, enabling the GSPs to solve the network provisioning problem, defined in Problem 5.5.6, giving a lower bound on $|\mathcal{F}_\epsilon(\mathbf{x})|$ for any configuration of a given spread.

Theorem 5.5.7 (Lower-Bound on the ϵ-Feasible Area)**.** *Let $\mathbf{x} \in \mathbb{R}^{n\times 2}$ be any configuration of n players with geographic spread $\sigma(\mathbf{x})$. Under the JMRA algorithm, we have $\forall \epsilon \in [0,1]$:*

$$|\mathcal{F}_\epsilon(\mathbf{x})| \geq |\mathcal{F}_\epsilon(\boldsymbol{\omega}(n, \sigma(\mathbf{x})))|. \tag{5.18}$$

A proof of this theorem is found in Appendix D.1. An intuitive interpretation is that regular configurations are the most "spread-out" among the class of n players configurations with a geographical spread equal to $\sigma(\mathbf{x})$, hence leading to the smallest ϵ-feasible region.

We now present a corollary of Theorem 5.5.7, allowing the GSPs to solve the network resource provisioning problem. For tractability, we shall present the case where $\nu = 1$.

Corollary 5.5.8 (Smallest Server Density)**.** *Let $\Phi(\lambda)$ be a homogeneous PPP of intensity λ compute nodes/m^2. The smallest compute node density λ_{min} required to guarantee an $(n, \sigma, \epsilon, \alpha, 1)$-service coverage, for $n \in \mathbb{N}, \sigma \in \mathbb{R}_+, \epsilon \in [0, 1], \alpha \in [0, 1]$ is given by:*

$$\lambda_{min}(n, \sigma, \epsilon, \alpha) = \frac{-\ln(\alpha)}{|\mathcal{F}_\epsilon(\boldsymbol{\omega}(n, \sigma))|} \tag{5.19}$$

The proof directly follows from Theorem 5.5.7 and the fact that for a PPP, see [14]: $\mathbb{P}(|\Phi(\lambda) \cap \mathcal{F}_\epsilon(\boldsymbol{\omega}(n, \sigma))| \geq 1) = 1 - e^{-\lambda|\mathcal{F}_\epsilon(\boldsymbol{\omega}(n,\sigma))|}$.

Now that we established the optimal strategy for GSPs to densify the network, we study through numerical simulations how they should dimension the G-server VMs' compute capacity. Specifically, Figure 5.4 exhibits results capturing the effects of the geographical spread σ and the per-player compute capacity k_G on the area of the ϵ-feasible region corresponding to a regular configuration, and ultimately, on the required $\lambda_{\min}$. The results presented correspond to two different scenarios. In Figures 5.4a and 5.4b, we use parameters relevant to classical MCG instances, e.g., involving players interacting on a common virtual first-person shooter game, requiring a somewhat loose timeliness constraints (on the order of 100 milliseconds end-to-end [40], or around 50 milliseconds for the player-to-server leg). However, Figure 5.4 corresponds to an XR-MCG game setting, where players are equipped with

XR headsets, requiring much tighter timeliness guarantees (on the order of 10 millisecond end-to-end [51], or around 5 milliseconds for the player-to-server leg). We study both scenarios separately. In these experiments, we adopt the functional forms introduced in Equations 5.12 and 5.13 to model the transport and batch computation delays.

The General MCG Setting. One can first observe in Figure 5.4a that the area of the ϵ-feasible region decreases with the players' geographical spread σ. This effect leads in turn to a sharp increase in the required density $\lambda_{\min}$, see Figure 5.4b, to compensate for the reduced area. This rapid increase is explained by the fact that $|\mathcal{F}_\epsilon(\boldsymbol{\omega}(n,\sigma))|$ eventually vanishes as the players become too widely spread, leading to the vertical asymptotes shown in Figure 5.4b. Therefore, for a fixed capacity per G-server's VM instance, we witness a geographical spread limit after which densification can no longer help in guaranteeing $(n,\sigma,\epsilon,\alpha,\nu)$-service coverage. Supporting larger spreads can then only be achieved by increasing the servers' compute capacity. One direct implication of this observation is that GSPs that can perform efficient matchmaking, i.e., match players in close proximity of each other, can afford to reduce the servers' compute capacity k_G, while keeping the server density reasonably low. In addition, we recognize in Figure 5.4b a law of diminishing returns on feasible σ with increasing k_G, pointing to the existence of a fundamental limit on the maximum geographical spread that can be supported for any n-player game, regardless of the network resources deployed and rate adaptation policy, due to the sole impact of the transport delay on the age of

the game, see Equation 5.1.

We note that while the initial model proposed in this chapter does not capture this effect, servers and players are in reality likely to be more densely located in cities. When the player's geographical spread is small enough, e.g., games involving players in the same city, then the GSPs can afford to provision compute nodes mostly in cities as per Figure 5.4, and the G-server would be placed nearby the players' city. If, however, GSPs want to support games with higher geographic spread, e.g., involving players across different cities, then they may need to densify the compute nodes between the cities, in the associated ϵ-feasible regions that is intuitively close to "center" of the players' configuration.

The XR-MCG Setting. Comparing Figure 5.4b to Figure 5.4c, one can highlight three key challenges faced by GSPs with extremely tight timeliness constraints, e.g., supporting XR-MCG. The first challenge is the need to ensure that the compute delay is as small as possible. To this end, the compute capacity per player k_G needs to be large enough to guarantee that the constraint in Problem 5.3.1 can be satisfied. Hence, XR-MCG instances require additional compute capacity compared to traditional MCG games.

The second challenge is the need to ensure that the players' geographical spread is small such that all the players are close enough to the G-server, leading to low transport delays. This is reflected by the scale of the horizontal axis, showing that XR-MCG instances can only be supported by connecting local players, e.g., in the same neighborhood/city, as opposed to the coun-

try/continent scale for traditional MCG instances.

The third challenge is the need to heavily densify the network to ensure small transport delays (and hence low variability under Assumption 5.5.1) so as to meet the service coverage requirement with a tight game age. The required density is on the order of $10^{-9} - 10^{-8}$ compute nodes per square meter, which clearly calls for leveraging the edge computing infrastructure to host the G-servers, in addition to the (potentially colocated) R-servers. Hefty network resource provisioning costs stemming from allocating considerable compute power in densely deployed edge compute nodes are unavoidable for XR-MCG GSPs to meet the tight game age constraint associated with such types of applications.

5.6 MCG Network Management

In Section 5.5, we showed how GSPs can ensure high service coverage by appropriately provisioning the network resources. We now assume that these resources have been provisioned, and we use insights extracted from our previous analysis to investigate strategies that can be adopted by GSPs to improve the MCG-QoS. We identify and study two complementary problems faced by GSPs: (1) the G-server placement problem, i.e., finding the best compute node to host the G-server for a particular player configuration, and (2) the matchmaking problem, i.e., finding the grouping of players to assign to a particular G-server/MCG instance.

5.6.1 The Service Placement Problem

We study first the particular problem of G-server placement, consisting in selecting the best compute node to host the G-server's VM among a set of feasible options given a players' configuration, see Figure 5.3. Previously, we showed how the JMRA algorithm can help to ensure that the spatial region that may contain feasible servers has the largest area. This region is likely to contain multiple compute nodes, all of them satisfying the game QoS requirement. While this may imply that all of the options are equivalent in the framework formulated in this chapter, additional considerations such as robustness to network delay variability may motivate the GSPs to prefer some options over others. For instance, considerations such as cost of service deployment [47, 169], load balancing among edge servers to ensure high service availability [85], or energy consumption [170] may be relevant factors to take into account in the final service placement decision. However, for the specific case of MCG networks, we envision that resiliency to delay variations will be a major criterion to consider in the service placement decision of G-servers. Indeed, delay variations may considerably change the shape and area of the region of feasible servers over time, and frequent costly migrations of service entailed by those variations can considerably impact all the players' gaming experience. So far, we have been considering problems with fixed values to capture network delays, either because they are measured in real-time as in Section 5.4, or because we assumed that GSPs provision the network by only considering a relevant statistic of the instantaneous network delays (e.g., its

estimated mean) as in Section 5.5. We now investigate how the GSPs might go about selecting the G-servers' VMs locations to improve the robustness of the MCG-QoS under varying network delays/congestion statistics. This is an important consideration, as one can expect MCG sessions to potentially last several hours.

We first observe that JMRA can increase robustness to slow variations in network/congestion delays over time as the optimal choice of update rate can adapt to such variations. However, this reactivity feature of JMRA may not be sufficient to keep the game ϵ-playable under significant variations, or if the players are mobile. Indeed, a change in the joint delay statistics experienced by the players may induce a substantial change in the shape of the region $\mathcal{F}_\epsilon(\mathbf{x})$ causing a potential need to trigger a costly G-server VM migration. Therefore, given the opportunity to select a server among multiple feasible options, a simple strategy would be to select the one that maximizes the expected value of the MCG-QoS, as it would keep the game ϵ-playable under the largest delay variations. Hence, the GSPs might maximize a new service placement MCG-QoS:

Definition 5.6.1 (MCG-QoS for Service Placement)**.** *Given a player configuration $\mathbf{x} \in \mathbb{R}^{n\times 2}$, and a feasible G-server location $\mathbf{g} \in \mathbb{R}^2$, inducing a distance vector $\boldsymbol{\delta} \in \mathbb{R}^n_+$, s.t. $\delta_i = \|\mathbf{x}_i - \mathbf{g}\|_2$, we define the MCG-QoS $\bar{q}(\boldsymbol{\delta})$ for service placement, for a given $\epsilon \in [0,1]$, as:*

$$\bar{q}(\boldsymbol{\delta}) = \mathbb{P}(q(\mathbf{D}^t_{\boldsymbol{\delta}}) > 1 - \epsilon) \tag{5.20}$$

i.e., the probability that the game remains ϵ-playable under JMRA and variable network congestion statistics.

Based on this MCG-QoS, we formally define the service placement problem as follows:

Problem 5.6.2 (Service Placement)**.** *Given a player configuration $\mathbf{x} \in \mathbb{R}^{n\times 2}$ and a realization ϕ of the spatial server deployment Φ, inducing a set $\mathcal{G}(\mathbf{x}, \phi) = \{\mathbf{g_1}, \ldots, \mathbf{g_l}\}$ of l ϵ-feasible server locations, the service placement problem consists in finding server $\mathbf{g}^*(\mathbf{x}, \phi)$, s.t.:*

$$\mathbf{g}^*(\mathbf{x}, \phi) = \arg \max_{\mathbf{g}_k \in \mathcal{G}(\mathbf{x},\phi)} \left\{ \bar{q}(\boldsymbol{\delta}) \ : \delta_i = \|\mathbf{x}_i - \mathbf{g}_k\|_2, \forall i \right\} \tag{5.21}$$

A straightforward way to solve Problem 5.6.2 would be to compute the MCG-QoS for service placement assuming that each of the candidate servers is selected to host the G-server, and choose the one that maximizes it. However, computing the MCG-QoS function may be impractical and computationally expensive as it involves solving numerous optimization problems, and laboriously estimate the distribution of $\bar{q}(\boldsymbol{\delta})$ through advanced sampling techniques.

To overcome this issue, we envision a 3-step algorithm, that can run in a centralized server, and that is aware of all the players' locations and the map of compute nodes:

Step 1: Exploration. First, one needs to identify the search space $\mathcal{G}(\mathbf{x}, \phi)$ of candidate servers. This can be performed either by considering all the compute nodes within a vast region containing all the servers "within

reach" of any player, i.e., such that the transport delay does not exceed the age constraint. As this solution would likely lead to an excessively large search space, heuristics can be leveraged to restrict the set to candidate servers, e.g., considering the l closest servers to the center of mass of configuration $\mathbf{x}$.

Step 2: Elimination. Second, one can considerably simplify the search space by only using the geometry of the players' configuration, as presented in Theorem 5.6.3.

Theorem 5.6.3 (Preferred G-Server Location)**.** *Given a player configuration $\mathbf{x} \in \mathbb{R}^{n\times 2}$ and a compute node deployment ϕ, let $\mathbf{g}$ and $\mathbf{g}' \in \mathcal{G}(\mathbf{x}, \phi)$ be the coordinates of two servers inducing distance vectors $\boldsymbol{\delta}$ and $\boldsymbol{\delta}'$, respectively. We have under Assumption 5.5.1:*

$$\boldsymbol{\delta} \prec_w \boldsymbol{\delta}' \implies \bar{q}(\boldsymbol{\delta}) \geq \bar{q}(\boldsymbol{\delta}') \tag{5.22}$$

hence the server at location $\mathbf{g}$ is to be preferred over the one at $\mathbf{g}'$.

where $\prec_w$ denotes the weak majorization ordering, see [112]. The proof of this theorem can be found in Appendix D.2.

Using this result, some of the candidate servers can be eliminated in $\mathcal{O}(l^2)$ time only by inspecting the distance vectors induced by the players' configuration $\mathbf{x}$ and each potential server in $\mathcal{G}(\mathbf{x}, \phi)$. We note however that weak majorization is merely a partial order, hence not any pair of distance vectors can be compared and this procedure does not guarantee to single out the best candidate server. In such a case, the algorithm needs to execute Step 3.

Step 3: Approximation. Third, once the number of candidate servers has been reduced to only a few candidates, additional heuristics can be exploited to select the final server. For instance, $q(\mathbf{d}^t)$ can be used as a surrogate for $\bar{q}(\boldsymbol{\delta})$, where $\mathbf{d}^t$ can be measured/estimated as discussed in Section 5.4. Finally the best compute node is confirmed if its MCG-QoS function exceeds the desired level ϵ.

We assess the performance of the elimination step by studying the effect of the size of the search space l and the number of players n on the average number of survivors (i.e., options that were not eliminated in step 2), for a fixed players' geographical spread and density of servers. The average is taken over random player configurations of given spread, and over realizations of Φ. Clearly, values close to 1 are associated with an effective elimination. In this experiment, we initialize $\mathcal{G}(\mathbf{x}, \phi)$ to contain the l closest compute nodes in ϕ to the center of gravity of the configuration $\mathbf{x}$, as suggested in Step 1. We present the results of this experiment in Figure 5.5.

One can observe that the average number of survivors increases slowly with l, as additional options are increasingly more likely to be eliminated. This confirms that proximity to the center of gravity of the player's configuration is a valid criterion to initialize the search space. In addition, the elimination step appears to be the most effective in games involving a large number of players. Indeed, larger values of n lead to a *hardening* of the spatial distribution of players, homogenizing it over a disk of radius σ, and bringing the center of gravity closer to the center of this disk. This in turn increases the likelihood for

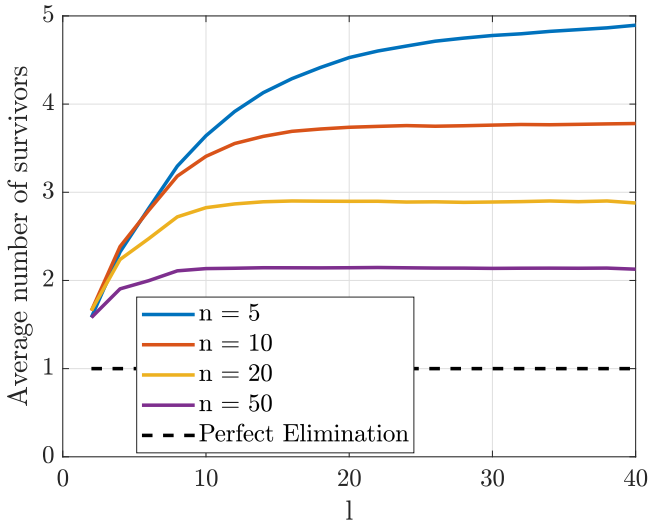


Figure 5.5: Effect of the size of the search space l and the number of players n on the average number of survivors; $\sigma = 2 \times 10^6$m, $\lambda = 4 \times 10^{-12}$ servers/m^2.

any sub-optimal server to find at least one of the players being prohibitively far, hence making it more likely to be eliminated as its associated distance vector will weakly majorize the ones of servers that are closer to the center of gravity.

5.6.2 The Player Matchmaking Problem

Another problem the GSPs may face is the player matchmaking problem. If the game allows it, the GSPs/game designers may wish to ensure that players are in close proximity of each other to improve the MCG-QoS as previously argued in this chapter.

Matchmaking has already been studied in the context of multiplayer games to match players that have similar game expertise [44], but not for

the purposes of clustering players geographically. Given the analytical framework presented in this chapter, the matchmaking problem can be reduced to a clustering problem, that aims, e.g., at finding the partition of players that minimizes the maximum geographical spread of the induced player configurations, given all the players' coordinates. More practical strategies can also be proposed. For example, one might impose a geographic restriction regarding the players' locations, e.g., all the players being in the same building, see, e.g., XR-MCG solutions such as [163]. Another solution would be to ensure that all the players are not too spread out by matching only players in the same city or neighborhood. Clearly, optimal and efficient matchmaking is a difficult problem, and judicious strategies may greatly depend on considerations such as the game type, game rules, demand in the service, or willingness of players to be matched with random players. Regardless of the specifics of the game, results and insights from Section 5.5 can be exploited to design customized matchmaking algorithms.

5.7 Chapter Conclusion

In this chapter, we studied fundamental questions that arise in the design of MCG systems. We introduced an MCG-QoS capturing the freshness of the information processed by the G-server, as well as the joint impact of the variable delays experienced by the players. We proposed JMRA, an efficient measurement-based joint update rate adaptation algorithm maximizing the MCG-QoS. We then related the game's geometry to the network delays

experienced by the players, and showed how GSPs can benefit from JMRA to combat the effect of geographical spread and slow-variability in the network delays/congestion, through effective network resource provisioning and service placement. We note that MCG player matchmaking, i.e., finding the best set of players to match on the same G-server, is a network operation problem that is complementary to service placement, and is also worth studying. Given the analytical framework presented in this chapter, the matchmaking problem might be reduced to a clustering problem aiming at finding the partition of players that minimizes the geographical spread of the induced configurations. We intend to study this problem in future work.

Chapter 6

Timely Information Sharing in Edge-supported Vehicular Collaborative Sensing Networks

As vehicles are progressively gaining in autonomy, accurate environment awareness is becoming an increasingly important feature. The conventional way to provide them with such awareness is by equipping them with various complementary sensors such as cameras, ultrasonic sensors, radars and/or LiDARs. These sensors may however be insufficient to provide the vehicles with a complete perception of the environment in some situations, e.g., when a target (such as a pedestrian, a cyclist, or another vehicle) is obstructed. This may lead to undesirable consequences such as needing to reduce the safe driving velocity of the vehicle to lower the risk and severity of a potential collision. To handle this issue, getting assistance from other network nodes, e.g., the cellular infrastructure, or other vehicles, has been proposed as a solution [48, 155]. This chapter[1] explores how the vehicles' situational awareness can be improved at the cost of using valuable wireless communication resources. This opportunistic assistance is particularly useful as the vehicles

[1]Publications based on this chapter: S. Kassir, G. de Veciana, Opportunistic Collaborative Estimation for Vehicular Networks. Submitted to ACM MobiHoc 2022.

may have different sensor qualities and their measurements may be autocorrelated over time, affecting the amount of information collected individually over time. Nevertheless, vehicles need to remain able to operate independently in case no network access or no other network members is available to provide any complementary data. Given this requirement, it is critical to adopt a suitable network architecture and design a parsimonious information sharing scheme while guaranteeing the safety of all the vehicles in the network.

6.1 Related Work

The power of sensor collaboration has been well studied in the literature, e.g., in the context of Collaborative Signal and Information Processing (CSIP) [56, 57], analyzing wireless sensor networks where nodes cooperate to achieve a common goal (e.g., target tracking [56, 57], classification [128, 50]) in an energy-efficient and fault-tolerant manner. In this work, we consider a CSIP system, but focus on features that are more specific to vehicular networks, e.g., wireless spectrum usage and estimation accuracy.

Multiple works have investigated techniques for collaborative sensing applied to vehicular networks. Two common approaches to improve the situational awareness, or equivalently the vehicles' tracking error, are adaptation of the sensors' transmission rates, see [5, 181], transmission power [157], or both jointly [74, 134]. In [5] for instance, the authors propose an age-minimizing mechanism where vehicles broadcast periodically situational information to each other at an optimal rate. While the adopted model captures essential

communication features in collaborative sensing networks such as contention and transmission delays, no specific environment estimation strategy is discussed and the quality of the observations from each sensor/vehicle is not taken into account in the analysis. In other studies such as [74], the authors present an adaptive transmission control strategy aiming at jointly controlling the vehicles' transmission rates and transmission power to adjust to the current channel congestion leading to improved tracking accuracy for the participating vehicles. Most of these works focus however on strategies based on broadcast sensing messages using protocols such as DSRC, interconnecting all the vehicles through a mesh network whose reliability may depend on the channel congestion level [3]. While distributed environment estimation is common in wireless sensor networks [121, 161, 186] for its effective computational and failure resilience properties, this topology often leads to hefty communication costs as the network scales up. Consensus-based solutions disseminating information to a subset of the nodes have been proposed [121, 29] but often affecting the convergence rate of the local nodes' estimation filters. We envision a more centralized data dissemination protocol performed via the cellular infrastructure using dedicated and reliable unicast links from/to participating vehicles. This solution might be more suitable for such safety-sensitive data. As wireless resource utilization is costly, we aim to minimize the communication load on the network and we integrate rate adaptation as part of our solution.

Another relevant class of work studies information fusion mechanisms

that could be applied to collaborative vehicular sensing. For instance, Distributed Kalman Filters (DKF) have been proposed to allow individual sensors to share their respective state estimates and error covariance matrices with a central node, that combines them optimally so as to minimize the mean-squared error (MSE) of the fused estimate, see, e.g., [69, 156]. In this work, we leverage results from the track-to-track fusion literature (see [156]), studying optimal combination of local state estimates from distributed nodes at a global aggregation node, to build and analyze the performance of our proposed collaborative sensing network.

Finally, the idea of timely information sharing and the role of data rate adaptation has been thoroughly studied via the notion of *Age-of-Information* [87, 154, 85], particularly relevant when network congestion is considered. This work proposes an alternative approach to capture information timeliness by studying the evolution of the estimation mean squared error over time, that has a more pragmatic interpretation for safety-critical applications.

6.2 Chapter Contributions and Organization

This chapter makes five major contributions.

First, we propose a collaborative environment estimation framework allowing vehicles to operate independently using their respective tracking filters while being able to opportunistically receive assistance from other vehicles with better sensing abilities via the network infrastructure as needed.

Second, we propose a novel information sharing mechanism and characterize how the data rate from the infrastructure to the assisted vehicles improves their local estimation accuracy.

Third, we present our Vehicle Information Sharing Algorithm (VISA) combining three data sharing policies which minimize the communication overheads while ensuring that all the vehicles' local estimation errors are kept below their respective desired targets. We leverage supermodularity to derive an opportunistic greedy sensor selection algorithm with suboptimality guarantees allowing VISA to adapt in real-time to time-varying network parameters.

Fourth, we evaluate the performance of the proposed opportunistic collaborative-sensing framework as compared to unassisted environment estimation solutions via system-level simulations. We show how considerable gains in feasible estimation accuracy can be enabled by communication-efficient sensing information sharing.

Fifth, we study a pedestrian tracking scenario allowing us to characterize the marginal value of the information shared by vehicles with VISA, using the Safe Driving Throughput (SDT) as a performance metric.

The remaining of this chapter is organized as follows. In Section 6.3, we describe our proposed collaborative sensing system architecture and introduce our network and environment models. In Section 6.4, we formulate the Vehicle Information Sharing Problem and we present our information control mechanism that in turn allows us to derive a Vehicle Information Sharing Algorithm

in Section 6.5. In Section 6.6, we complement our analysis with simulation-based experiment results, and we provide additional discussion on the value of information sharing in vehicular sensing networks in Section 6.7. Finally, we conclude the chapter in Section 6.8.

6.3 Solution Architecture and Model

We propose a system designed to allow a set of geographically neighboring vehicles to collaborate and opportunistically improve the quality of their respective estimates for their shared dynamic environment. The system architecture we present is designed to satisfy the following requirements:

- the vehicles should not be fully dependent on the opportunistic collaborative framework, i.e., they should be able to operate independently (with lower but acceptable performance guarantees) even without any network assistance;
- the solution should be flexible and sufficiently computationally efficient to allow for frequent reconfiguration as the vehicles' dynamics may lead to intermittent vehicle availability and non-stationary dynamics/sensing ability;
- the solution should ensure that the lowest amount of communication resources is used for any desired estimation accuracy.

Our proposed opportunistic collaborative solution is compared to a

baseline network topology where vehicles do not cooperate which is introduced next.

6.3.1 Baseline Network Model

Consider a network composed of a set $\mathcal{V}$ of vehicles evolving in a common environment, each equipped with a sensor (or possibly multiple ones) and connected to a common infrastructure node, such as an edge server or a cellular base station. To gain situational awareness and navigate safely, vehicle $i \in \mathcal{V}$ observes its environment by taking periodic measurements $\{\mathbf{z}_{i,k}\}_{k=1}^{\infty}$ via its sensor. All the sensors operate synchronously at a common sampling rate of τ^{-1} samples/second. These measurements are fed to a Local (Kalman) Filter (LF) that allows each vehicle to keep track of both its own environment state estimate $\hat{\mathbf{x}}_{i,k}$ and its error covariance matrix $P_{i,k}$ at time k.

6.3.2 Proposed Collaborative Network Architecture

Next we present an alternative network architecture wherein vehicles do not rely solely on their own sensor's measurements, but can also benefit from other vehicles' sensing data to improve their local environment state estimates. In this system, we distinguish two (possibly overlapping) subsets of vehicles: (1) a set $\mathcal{C} \subseteq \mathcal{V}$ of data contributors, and (2) a set $\mathcal{R} \subseteq \mathcal{V}$ of data recipients. The data contributors take measurements of the environment and update their respective LFs as in the baseline model, but additionally send their local state estimates and local error covariance matrix to the infrastructure node after

each LF update, hence at a rate of τ^{-1} updates/s. The base station/edge server will in turn combine optimally (see [156]) all the local state estimates via a master filter (MF) to form an environment master state estimate $\hat{\mathbf{x}}_k$ and master error covariance matrix P_k. The infrastructure node finally resets the LFs of the vehicles in $\mathcal{R}$ with the master state estimate which will be more accurate than the local one, i.e., the LFs discard their local estimates/error covariance matrix and replace them with the corresponding ones from the MF. The vehicles in $\mathcal{R}$ then evolve independently by taking their own measurements and updating their own local model until the next reset signal from the MF. Each vehicle $i \in \mathcal{R}$ has a feedback rate from the infrastructure node equal to ρ_i resets/second, configured judiciously to ensure a low communication cost. This feedback synchronizes the environment estimates of the MF and LFs, allowing the vehicles to benefit from an environment estimate of improved quality.

As we explain and precisely characterize in the sequel, the rate ρ_i controls the magnitude of the peak local error covariance matrix $\tilde{P}_i^*$ of vehicle i when the MF error has reached its steady-state. This overall framework, summarized in Figure 6.1, allows the vehicles to leverage sensing information from other vehicles to improve the quality of their own environment state estimates. Finally, we neglect the effect of communication delays on the estimation error, as packet transmissions happen on a much smaller time scale than the sampling period and the environment velocity.

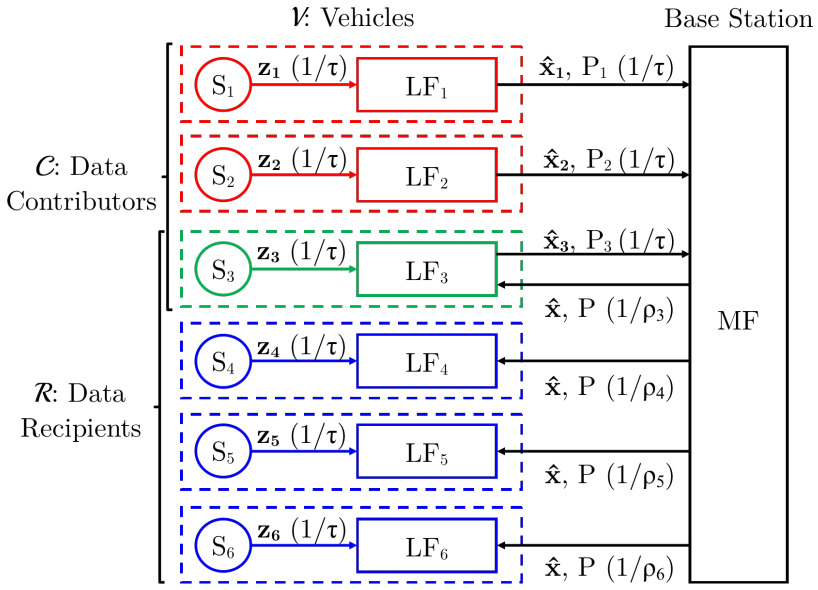


Figure 6.1: Cooperative Environment Sensing Data Flow. The value in parenthesis indicates the data rate.

6.3.3 Environment Estimation Model

In the sequel, we shall assume a simple representative model for the environment dynamics, amenable to analysis, yet capturing its essential features.

Assumption 6.3.1. *We assume that an element of the environment tracked by the vehicles, e.g., the x-coordinate of a pedestrian, cyclist, or other vehicle, is characterized as a Brownian motion, independent of the other elements being tracked.*

While we acknowledge this is a strong assumption, it allows us to extract essential insights that can be generalized when it is lifted. We provide

additional discussion and justifications for this simplified system in Section 6.7.

As elements of the environment can be tracked independently, we consider a discrete-time environment state evolution model of the form:

$$x_{k+1} = x_k + w_k, \tag{6.1}$$

where $(w_k)_{k=1}^{\infty}$ is a stationary white Gaussian process, such that $w_k \sim \mathcal{N}(0, Q)$ for all k, where Q is proportional to the sampling period τ, i.e., $Q = \nu^2\tau$ for a known network parameter ν that can be interpreted as the *environment velocity*.

6.3.4 Environment Observation Model

As vehicles navigate in their environment, they observe their surroundings by taking measurements via their respective sensors. The sensors in different vehicles are assumed to take independent and unbiased measurements of the environment, yet different vehicles may have different precision as they may be equipped with different sets of sensors of different quality/manufacturers. In addition, the sensors' measurement errors may emerge from (1) the sensors' intrinsic properties but also from (2) biases that may be dependent on the environment, e.g., the target is obstructed. The second type of errors depends on slowly varying factors (e.g., the existence of a wall near the target, or the relative position of the sensor and the target), leading to temporal correlations in successive measurement errors, that shall be modeled. We model a

measurement $z_{i,k}$ taken by sensor i at time k as

$$z_{i,k} = h_i x_k + v_{i,k}, \forall i \in \mathcal{V}, \tag{6.2}$$

where $h_i \in \mathbb{R}^*$, and $(v_{i,k})_{k=1}^{\infty}$ is a stationary Gaussian noise process modeled to be autocorrelated over time as follows

$$v_{i,k+1} = \alpha_i v_{i,k} + \zeta_{i,k}, \forall i \in \mathcal{V}, \tag{6.3}$$

and where $(\zeta_{i,k})_{k=1}^{\infty}$ is a stationary white Gaussian process, such that $\zeta_{i,k} \sim \mathcal{N}(0, \sigma_i^2)$ for all k, and $0 \leq \alpha_i < 1$ for all i in $\mathcal{V}$. In this model, $\boldsymbol{\alpha} = (\alpha_i : i \in \mathcal{V})$ and $\boldsymbol{\sigma^2} = (\sigma_i^2 : i \in \mathcal{V})$ are known system parameters that can be interpreted respectively as the autocorrelation factor between two consecutive sample errors, and the variance of an independent measurement noise process. For instance, α_i can be small in a fast-moving environment relative to the sampling rate τ^{-1}, or when the relative velocity between vehicle i and the target is large, while σ_i^2 can be large when vehicle i is equipped with few imprecise sensors.

In the sequel, we shall assume that $h_i = 1$ for any sensor i without loss of generality, as $z_{i,k}$ can be processed (more specifically divided by h_i) and the variance of $v_{i,k}$ can be adapted accordingly leading to an equivalent expression virtually independent of h_i. The linear dynamical system equations characterizing the environment and the sensors' measurements can therefore be summarized as:

$$\begin{cases} x_{k+1} = x_k + w_k, & \\ z_{i,k} = x_k + v_{i,k}, & \forall i \in \mathcal{V} \\ v_{i,k+1} = \alpha_i v_{i,k} + \zeta_{i,k}, & \forall i \in \mathcal{V} \end{cases} \tag{6.4}$$

such that $w_k \sim \mathcal{N}(0, Q)$, $\zeta_{i,k} \sim \mathcal{N}(0, \sigma_i^2)$ for any vehicle i, and $\mathbb{E}[\zeta_{i,k}\zeta_{j,k}] = 0$, for any vehicle $i \neq j$, for all k.

A classical filtering technique to track such a linear dynamical process with autocorrelated measurements is to perform time-differencing on successive measurements and feed the resulting process into a standard Kalman Filter. Petovello's method [126] is a standard approach that we shall adopt in this chapter. We provide a technical description of this method in Appendix E.1. Note, however, that we envision the collaborative sensing system to be implemented somewhat differently than in [126] as the MF is not filtering the local measurements at every iteration but is combining the local estimates instead, similarly to the Distributed Kalman Filtering (DKF) technique, see [69, 187]. The equivalence between the two filtering approaches has been established in the Track-to-Track fusion literature, see, e.g., [156].

6.4 Information Control Mechanism

In this section, we study the performance of the collaborative sensing network introduced in the previous section by formally characterizing its underlying information control mechanism. We then formulate our problem that we shall address in the sequel.

6.4.1 Peak Local Error Variance Characterization

We start by providing a more formal characterization of the peak local error variance function $\tilde{P}_i^*$ of vehicle i, and shed light on the role of the feed-

back loop from the MF to the LFs. We study the regime where the MF's (a posteriori) error variance has reached its steady-state of P^*.

Consider initially a setting where the LF of vehicle i receives a single reset signal from the MF at time t_0 and is left to operate autonomously indefinitely after that time. The sensor would accumulate information about the environment from its own sensor. Subsequently, the local estimation error variance continuous-time process alternately (1) increases linearly at a rate ν^2 m^2/s in between local measurements, consistently with the properties of the environment process noise introduced in Section 6.3.3, and (2) drops (by less than Q) when a new local measurement is integrated in the LF. As illustrated in black in Figure 6.2, this process eventually reaches a steady-state where the local error variance fluctuates between the a posteriori P_i^* and the a priori $\overline{P}_i^* = P_i^* + Q$ levels.

The sequence of sensor i's a posteriori local estimation error variances $(P_{i,k|k})_{k=0}^{\infty}$ can be fully characterized through a recurrence equation expressed in the result below.

Definition 6.4.1 (Local Estimation Error Variance Recurrence)**.** *Consider the dynamical system defined in Equation 6.4 tracked using a Kalman Filter, then the local a posteriori estimation error variance of sensor i can be characterized recursively as follows:*

$$P_{i,k+1|k+1} = \frac{P_{i,k|k}(\sigma_i^2 + Q\alpha_i^2) + Q\sigma_i^2}{P_{i,k|k}(1-\alpha_i)^2 + \sigma_i^2 + Q} \triangleq T_i(P_{i,k|k}) \tag{6.5}$$

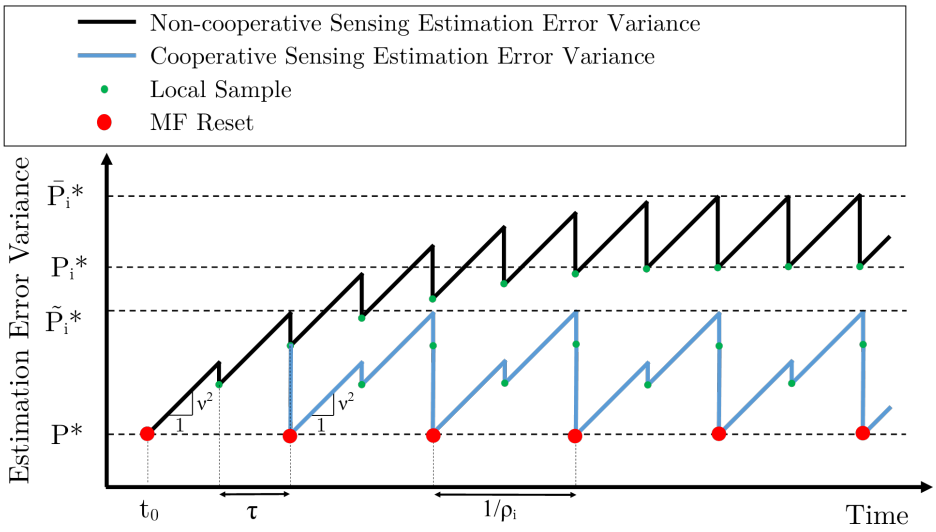


Figure 6.2: Illustration of the measurement error variance temporal process with and without assistance from the MF.

Proof. The result follows directly from combining Equations E.3 and E.10 for the system described in Equation 6.4. □

Now consider a setting where the MF resets the LF of vehicle i periodically at a rate $1/\rho_i$ - we motivate this constraint on the selection of ρ_i later in this section. At every multiple of τ, the MF estimation error variance process drops to P^*, as it just integrated sensing information from the LFs in $\mathcal{C}$. Thus, after every MF reset, the error variance of the specific LF also drops to P^*. The error variance process then evolves independently from this point as discussed earlier until a new reset signal is received, as illustrated in blue in Figure 6.2. The periodic feedback hence induces a peak LF error variance $\tilde{P}_i^*$ that is never exceeded for a specific choice of $\rho_i = \gamma\tau$ for $\gamma \in \mathbb{N}$, and that can be computed numerically using the local error variance recurrence

Equation 6.5 as follows:

$$\tilde{P}_i^* = T_i^{(\gamma-1)}(P^*) + Q \tag{6.6}$$

where $T_i^{(n)}(x) = T_i\left(T_i^{(n-1)}(x)\right), \forall n \in \mathbb{N}$, and $T_i^{(0)}(x) = x$.

6.4.2 Network Design Objective

We seek to design a collaborative sensing system that satisfies a target environment estimation accuracy level while using a minimal amount to communication resources.

To this end, we introduce the environment target peak error variance vector $\boldsymbol{\beta} = (\beta_i, i \in \mathcal{V})$ (in m^2) that each vehicle is willing to experience. For instance, vehicles desiring to drive at higher velocities might require lower thresholds to guarantee the safety of their passengers. We formulate the Vehicle Information Sharing Problem (VISP), for a given set of vehicles $\mathcal{V}$, autocorrelation parameter vector $\boldsymbol{\alpha}$, independent measurement noise variance vector $\boldsymbol{\sigma^2}$, sensor sampling rate τ^{-1}, and environment velocity ν^2, as follows:

Problem 6.4.2 (Vehicle Information Sharing Problem)**.** *Consider a vehicular network consisting of a set of vehicles $\mathcal{V}$, then the VISP consists in determining the optimal sets of data contributors $\mathcal{C}^* \subseteq \mathcal{V}$, sensing data recipients $\mathcal{R}^* \subseteq \mathcal{V}$, and the feedback rate vector $\boldsymbol{\rho}^* = (\rho_i^*, i \in \mathcal{V})$, such that:*

$$\min_{\boldsymbol{\rho},\mathcal{C},\mathcal{R}} \left\{ |\mathcal{C}| \cdot \tau^{-1} + \sum_{i \in \mathcal{R}} \rho_i : \ \tilde{P}_i^*(\mathcal{C}, \boldsymbol{\rho}) \leq \beta_i, \forall i \in \mathcal{V} \right\}, \tag{6.7}$$

i.e., minimizing the aggregate uplink and downlink data rate required to guarantee that all the vehicles' target peak estimation error variances are not exceeded in steady-state.

As the uplink and downlink packets have the same data size (both containing a state estimate and the associated variance), this parameter can be factored out of the cost function and does not appear in Equation 6.7, that only depends on the packet rates. Note that this problem is non-trivial as $\tilde{P}_i^*$ is a complex function of $\boldsymbol{\rho}$ and P^* that is itself a function of $\mathcal{C}$ (see Equation 6.6), and also depends on the vehicular network parameters $\boldsymbol{\alpha}$, $\boldsymbol{\sigma^2}$, τ^{-1}, and ν^2.

6.5 Vehicle Information Sharing Algorithm

In this section, we introduce our Vehicle Information Sharing Algorithm (VISA) to solve the VISP. We propose to decompose this complex problem into two stages: (1) assume a fixed contributor set $\mathcal{C}$ and determine the corresponding optimal recipient set $\mathcal{R}^*(\mathcal{C})$ and feedback rates $\boldsymbol{\rho}^*(\mathcal{C})$; (2) determine the optimal contributor set $\mathcal{C}^*$ given the associated optimal $\mathcal{R}^*(\mathcal{C})$ and $\boldsymbol{\rho}^*(\mathcal{C})$. We discuss these two stages separately, before summarizing the overall algorithm.

6.5.1 Recipient Set and Feedback Rate Determination

In this first stage, we consider a fixed contributor set $\mathcal{C}$ which reveals two key underlying parameters. First, it induces a MF steady-state a posteriori estimation error variance P^* that can be computed using the theorem below, proven in Appendix E.2:

Theorem 6.5.1 (MF steady-state estimation error variance)**.** *Consider a given a set of vehicles* $\mathcal{C}$ *contributing to a collaborative sensing system and tracking an environment characterized by Equation 6.4. The resulting MF steady-state a posteriori estimation error variance is given by*

$$P^*(\mathcal{C}) = \frac{-QE + \sqrt{Q^2FB + 4QB}}{2(B + Q(BA - CD))} \tag{6.8}$$

where

$$A = \sum_{\mathfrak{C}} \frac{\alpha_i}{\sigma_i^2},\ B = \sum_{\mathfrak{C}} \frac{(1-\alpha_i)^2}{\sigma_i^2},\ C = \sum_{\mathfrak{C}} \frac{\alpha_i(1-\alpha_i)}{\sigma_i^2},$$
$$D = \sum_{\mathfrak{C}} \frac{1-\alpha_i}{\sigma_i^2},\ E = \sum_{\mathfrak{C}} \frac{1-\alpha_i^2}{\sigma_i^2},\ F = \sum_{\mathfrak{C}} \frac{(1+\alpha_i)^2}{\sigma_i^2}.$$

Second, the corresponding peak local steady-state error $\tilde{P}_i^*$ can be determined by solving the fixed-point Equation 6.5 such that $P_{i,k+1|k+1} = P_{i,k|k}$. Equivalently, one could use Theorem 6.5.1 and the fact that $\overline{P}_i^* = P_i^* + Q$ leading to the following corollary:

Corollary 6.5.2 (Peak local steady-state error variance)**.**

$$\overline{P}_i^* = Q - \frac{Q}{2}\frac{1+\alpha_i}{1-\alpha_i} + \sqrt{\left(\frac{Q}{2}\frac{1+\alpha_i}{1-\alpha_i}\right)^2 + Q\sigma_i^2}, \forall i \in \mathcal{V}. \tag{6.9}$$

Given the peak local steady-state error variance expression, it becomes clear that the following policy determining the recipient set $\mathcal{R}^*$ is optimal:

Policy 6.5.3 (Recipient set selection)**.**

$$\mathcal{R}^*(\mathcal{V}) = \{i \in \mathcal{V} : \overline{P}_i^* > \beta_i\} \tag{6.10}$$

i.e., only the vehicles that need assistance to achieve their desired target error will receive data from the infrastructure node.

In addition, it directly follows from Problem 6.4.2 that the most spectrally efficient feedback rate ρ_i^* to any vehicle i in $\mathcal{R}^*$ is the slowest one that satisfies the constraint $\tilde{P}_i^* \leq \beta_i$. We obtain the following feedback rate selection policy:

Policy 6.5.4 (Feedback rate selection policy)**.**

$$\rho_i^*(\mathcal{C})^{-1} = \tau \cdot \arg\max_{\gamma}\{\gamma : T_i^{(\gamma-1)}(P^*(\mathcal{C})) + Q \leq \beta_i\} \tag{6.11}$$

We observe that Policy 6.5.3 ensures that $T_i^{(\gamma-1)}(P^*)$ is increasing in γ and hence $\rho_i^* > 0$ for any vehicle i in $\mathcal{R}^*$. We also note that the constraint in Problem 6.4.2 may not be slack under Policy 6.5.4, yet the following result proved in Appendix E.3 holds.

Theorem 6.5.5. *Consider the VISP defined in Problem 6.4.2 and a given contributor set $\mathcal{C}$. The feedback rate selection policy described in Policy 6.5.4 determining vector $\boldsymbol{\rho}^*(\mathcal{C})$ is optimal.*

6.5.2 Data Contributor Set Determination

We shall now use the two policies described earlier to determine one that selects the best set of data contributors $\mathcal{C}^*$. Here again, we shall solve this problem in two stages: (1) fix $|\mathcal{C}|$ and solve VISP given this constraint, (2) search exhaustively over all possible values of $|\mathcal{C}|$ from 0 to $|\mathcal{V}|$ and determine the one that minimizes the VISP's cost function.

Consider a fixed $|\mathcal{C}| = m$ and given Policy 6.5.3 and 6.5.4, VISP reduces to the simple form:

$$\min_{\mathcal{C}} \left\{ \sum_{i \in \mathcal{R}^*} \rho_i^*(\mathcal{C}) : \ |\mathcal{C}| = m \right\}. \tag{6.12}$$

where ρ_i^* is a function of P^* that is itself a function of $\mathcal{C}$. To solve this problem, we propose to solve an equivalent one as suggested in Theorem 6.5.6, proven in Appendix E.4:

Theorem 6.5.6. *Let $\mathcal{C}_m^*$ be a solution of the following problem*

$$\mathcal{C}_m^* = \arg\min_{\mathcal{C}} \{P^*(\mathcal{C}) : \ |\mathcal{C}| = m\}. \tag{6.13}$$

Then $\mathcal{C}_m^*$ is also a solution of Problem 6.12.

To solve Problem 6.13, we observe that it consists in minimizing a non-increasing supermodular function, see [158], over a uniform matroid constraint [143, 78]. This class of optimization problems is known to be NP-hard but a greedy approximation algorithm provided in Algorithm 6.1 has been shown to be $(1 - \frac{1}{e})$-optimal, see [143, 55].

Note that this algorithm ensures that $P^*(\mathcal{C}^*) \leq P_i^*$ for any vehicle $i \in \mathcal{V}$ as $P^*(\cdot)$ is a non-increasing set function. Equipped with Algorithm 6.1, the following policy can suggested

Policy 6.5.7 (Sensor selection policy)**.** *$\mathcal{C}^*$ can be determined via Algorithm 6.1 and searching exhaustively over $\{\mathcal{C}_m^*\}_{m=0}^{|\mathcal{V}|}$ for the element that solves the VISP.*

Algorithm 6.1: GREEDY$(\mathcal{V}, m, Q)$

Data: $\mathcal{V}, m \in \mathbb{N}, Q \in \mathbb{R}_+$
Result: Solves for $\mathcal{C}_m^*$

$\mathcal{C}_0^* \leftarrow \emptyset$
$\mathcal{S} \leftarrow \mathcal{V}$
for *k=1 to m* **do**
  $\mathcal{C}_k^* \leftarrow \mathcal{C}_{k-1}^* \cup \arg\min_{i \in \mathcal{S}} P^*(\mathcal{C}^* \cup \{i\})$
  $\mathcal{S} \leftarrow \mathcal{S} \setminus \mathcal{C}_k^*$
end
return $\mathcal{C}_m^*$

An interesting property of Algorithm 6.1 is that while the problem does not have an optimal substructure, the solution $\mathcal{C}_{m+1}^*$ returned by Algorithm 6.1 for $|\mathcal{C}| = m + 1$ is a superset of $\mathcal{C}_m^*$. This property allows us to find the final $\mathcal{C}^*$ efficiently by calling Algorithm 6.1 only once with $m = |\mathcal{V}|$, and saving the sequence $\{\mathcal{C}_k^*\}_{k=1}^{|\mathcal{V}|}$ in memory. Policy 6.5.7 can then be executed with an algorithmic complexity of $\mathcal{O}(|\mathcal{V}|^2)$.

6.5.3 Algorithm Summary

We now summarize VISA, the general procedure unifying the three policies described previously. We envision this subroutine described in Algorithm 6.2 to be continuously re-executed in an outer-loop as some network parameters such as ν, $\boldsymbol{\alpha}$ and $\mathcal{V}$ are varying over time and might need to be frequently re-evaluated, albeit at a slower time-scale than the sampling period τ. Indeed, the outer-loop need to give enough time for the MF to reach its estimation error steady-state and operate in it for a significant fraction of time.

Algorithm 6.2: Vehicle Information Sharing Algorithm

Data: $\mathcal{V}, Q \in \mathbb{R}_+$
Result: Solves for $\mathcal{R}^*, \boldsymbol{\rho}^*, \mathcal{C}^*$
$\mathcal{R}^* \leftarrow \mathcal{R}^*(\mathcal{V})$
if $\mathcal{R}^*$ *is empty* **then**
  $\mathcal{C}^* \leftarrow \emptyset$
else
  $\{\mathcal{C}^*_m\}_{m=1}^{|\mathcal{V}|} \leftarrow \textsc{Greedy}(\mathcal{V}, |\mathcal{V}|, Q)$
  for $m = 1$ *to* $|\mathcal{V}|$ **do**
    $\rho^*_{i,m} \leftarrow \rho^*_i(\mathcal{C}^*_m), \forall i \in \mathcal{R}^*$
    $Cost_m \leftarrow m \cdot \tau^{-1} + \sum_{i \in \mathcal{R}^*} \rho^*_{i,m}$
  end
  $m^* \leftarrow \arg\min_m Cost_m$
  $\boldsymbol{\rho}^* \leftarrow (\rho^*_{i,m^*} : i \in \mathcal{R}^*)$
  $\mathcal{C}^* \leftarrow \mathcal{C}^*_{m^*}$
end

6.6 Network Performance Evaluation

We now proceed to evaluate the performance of the collaborative sensing system presented in this chapter. We shall first provide simulation results that characterize a simple two-vehicle system before providing additional insights on the characteristics of more general networks.

6.6.1 Communication Cost Analysis in Two-vehicle Networks

We start by studying a simple scenario wherein two vehicles having the same sensing accuracy also have the same environment error threshold parameter, i.e., $\sigma_1^2 = \sigma_2^2 = \sigma^2$ and $\beta_1 = \beta_2 = \beta$. The two vehicles may have different perspectives of the environment modeled by potentially different measurement

autocorrelation parameters α_1 and α_2, which induces different individual estimation error performance (i.e., in general $P_1^* \neq P_2^*$). Hence, depending on the values of α_1 and α_2, the vehicles may require more or less assistance from each other to ensure that both can satisfy their respective estimation error constraint, leading to different communication costs as exhibited in Figure 6.3.

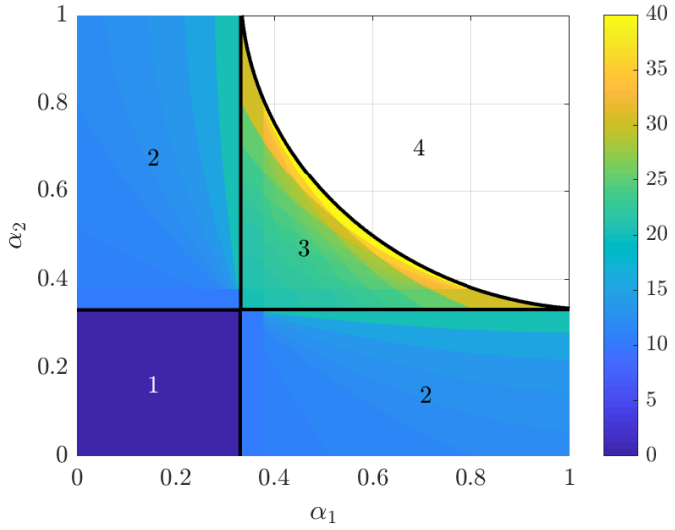


Figure 6.3: Communication Cost (in data transmissions/s) required to guarantee the two vehicles to achieve their estimation error target, for different combinations of (α_1, α_2), where $\sigma^2 = 1$ m^2, $\beta = 0.12$ m^2, $\tau = 0.1$ s and $\nu^2 = 0.1$ m^2/s.

One can distinguish four separate regimes in Figure 6.3, as marked on the figure. In the first regime, both α_1 and α_2 are small, leading to small individual estimation errors P_1^* and P_2^*. In fact, both P_1^* and P_2^* are below β, indicating that none of the vehicles needs assistance to satisfy their estimation error target. Thus, no communication cost is necessary.

In the second regime, one of the two vehicles has a large autocorrelation parameter requiring it to request assistance from the other, via the infrastructure node. Some communication cost is incurred as one vehicle sends its environment state estimation data to the base station/edge server, while the other receives it from the centralized node. Note that the communication cost increases (in discrete levels) within the region as either of the autocorrelation parameters increase in value. Indeed, as the contributor vehicle's estimate deteriorates, the value of P^* at the base station/edge server also deteriorates and the data recipient gets reset with estimates of poorer quality. Conversely, as the data recipient's estimate quality deteriorates, its associated local error sequence increases faster and hence risks to exceed β earlier. Both of these effects need to be compensated with larger feedback rate, inducing larger communication costs increasing in discrete steps consistent with Policy 6.5.4.

In the third regime, neither vehicle can satisfy their maximum target error requirement autonomously. Yet, they remain able to rely on each other by combining their estimates to form one of improved quality, leading to a peak error below the β threshold. As both vehicles need to send and receive data from/to the centralized node, the communication costs are the largest in this regime.

In the fourth regime too, none of the vehicles can satisfy their respective error threshold even with cooperation, as both provide estimates are of very poor quality. To operate in this regime, the vehicles will need to increase their estimation error target β (which may impact their safe driving speed), and/or

wait for assistance from another vehicle when one joins the network.

Note that we evaluated the network performance by studying its sensitivity to α_1 and α_2, but a similar analysis can be performed via the variations in σ_1 and σ_2 which also affect P_1^* and P_2^*.

6.6.2 Feasibility Analysis in General Networks

We now study properties of more general networks, and we more particularly attempt to provide additional insights on the extent to which vehicles in collaborative sensing networks can feasibly reduce their target error threshold as compared to the autonomous scenario. For instance, in Figure 6.3, we discussed how collaborative sensing can grow the α-feasibility region of a two-vehicle system from Region 1 to the union on Regions $1, 2$ and 3.

In general, we let $\beta^*(\boldsymbol{\alpha}, \boldsymbol{\sigma}^2, Q)$ represent the smallest feasible value of β that any vehicle can achieve in the network, regardless of the communication cost. We now consider a network composed of three vehicles having the same sensing accuracy (i.e., $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma^2$), and we study in Figure 6.4 the sensitivity of β^* to the vector $\boldsymbol{\alpha}$. More particularly, we consider six scenarios where α_2 and α_3 take different combinations of low, moderate and high values, and we study the effect of varying α_1 is each of these situations. While we consider a network of three vehicles for clarity of presentation, the analysis and conclusions hold for any general network.

The first major observation that can be made from Figure 6.4 is that $\beta^*(\cdot, \cdot, \cdot)$ is hardly sensitive to α_1 when at least one of the other vehicles has

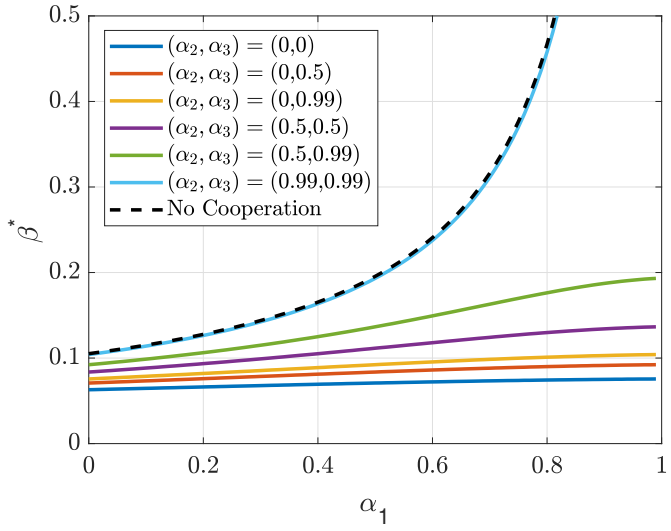


Figure 6.4: Study of the sensitivity of β^* to the measurement autocorrelation vector $\boldsymbol{\alpha}$, where $\sigma^2 = 1$ m^2, $\tau = 0.1$ s and $\nu^2 = 0.1$ m^2/s.

a low measurement correlation parameter, implying that having at least a single vehicle with a low P_i^* can be enough to ensure an excellent network performance as characterized by a small β^*. We note that the worse the joint quality of the observations from vehicles 2 and 3, the more critical it is for the ones originating from vehicle 1 to be of better quality to ensure that low values of β are feasible. The network performance is therefore mostly characterized by the quality of the estimate from the best vehicle, i.e., the one with the smallest P_i^*. In the extreme case, when both α_1 and α_2 are close to 1, β^* can be well approximated by $\overline{P}_1^* = P_1^* + Q$, as depicted in Figure 6.4. This property makes the proposed procedure particularly attractive in large-scale networks composed of a large collection of vehicles with various sensor qualities and measurement autocorrelation factors, as it becomes increasingly likely to find

at least one very accurate sensor as the network grows in size.

The second key observation that can be made from this figure is that the collaborative sensing procedure can allow vehicle i to considerably reduce its estimation error target, especially when vehicle i has a poor $\overline{P}_i^*$, e.g., when α_i is large, as observable by the considerable gap between the dashed and any solid line in Figure 6.4 that grows quickly with α_i. This reduction in threshold for instance can allow the concerned vehicles to increase their safe driving speed, benefiting the transportation system as a whole.

6.7 Value of Information in Vehicular Systems

6.7.1 Environment Evolution Model: an Information-centric Examination

The results presented in this chapter were derived under Assumption 6.3.1. Below we present an information-centric argument establishing that the analysis remains relevant in more general settings.

The first premise underlying Assumption 6.3.1 is that dynamic objects in the environment evolve as a Brownian motion. This may be seen as the evolution model with maximum entropy, among the class of continuous-time stationary processes whose random increments over a period of τ seconds have variance $\nu^2\tau$. While the dynamics of the environment may not be as uncertain in reality, the work presented in this chapter can be seen as providing a fundamental bound on the rate/quatity of information that can be exchanged in collaborative vehicular sensor networks, by considering the setting with the

least available *a priori* information. For instance, the actual optimal feedback rate will be no larger than the one determined by VISA assuming the Brownian motion model.

A similar argument can be made to motivate the second premise in Assumption 6.3.1, i.e., the independent models based on single-state environment elements. While real-wold environments are multidimensional, and states might be correlated with each other (e.g., velocity and position states), assuming independence among the tracked states also leads to information loss, and VISA will only overestimate the amount of information that should be shared over the network.

6.7.2 Value of Information Sharing in Vehicular Networks

So far, we have showcased how cooperation through VISA allows vehicles to improve their local environment estimates. We now examine how this improved accuracy benefits vehicles via a by simulating a specific scenario.

As previously argued, we envision that vehicles will already be able to drive safely in the environment, even without assistance from the network. However, opportunistically leveraging such assistance when it is available to provide vehicles with a better perception of their environment can be beneficial. We envision these benefits to translate to an increase in the vehicles' throughput. More specifically, we define the Safe Driving Throughput (SDT) to be the maximum mean vehicle rate (in vehicles/s) that can *safely* pass through a given cut of road of length l_{R}, see [164], and we shall show how col-

laboration can help increase this metric. From Little's formula [92], it should be clear that the following equality holds:

$$\text{SDT} = \frac{\eta}{l_{\text{R}}/v_{\text{S}}} \implies \text{SDT} \propto v_{\text{S}}. \tag{6.14}$$

where η represents the mean number of vehicles in the cut of road of length l_{R}. Hence, the safe driving velocity v_{S} can be seen as an equivalent efficiency metric as the SDT.

The challenge in assessing the value of information sharing is thus to characterize v_{S}. We shall consider a simple scenario, illustrated in Figure 6.5 and described below, with cooperating vehicles in an environment where the only source of uncertainty is the motion of a pedestrian.

- An *ego-vehicle* (vehicle of interest) is driving along a straight road (say the positive direction of the x-axis), with a breaking deceleration rate of a m/s^2, i.e., it can brake to slow down at a constant rate of a m/s every second. It follows that if it drives at velocity v_0, its breaking time t_b to reach a full stop is $t_b(v_0) = \frac{v_0}{a}$ and its breaking distance is $d_b(v_0) = \frac{v_0^2}{2a}$.

- A pedestrian is jaywalking on the road, and is detected/tracked by a set of n vehicles, including the ego-vehicle. The pedestrian's motion is modeled as a Brownian motion along the x-dimension with velocity ν^2 m^2/s for some time, and then leaves the network, e.g., it eventually crosses the road.

- While the pedestrian is on the road, a set of n vehicles, including the ego-vehicle, are detecting/tracking it using VISA, with heterogeneous sensing abilities characterized by the vectors $\boldsymbol{\sigma^2}$ and $\boldsymbol{\alpha}$. Without loss of generality, we use the convention that the ego-vehicle's sensing abilities are captured by the first entries of these vectors. We assume that all the vehicles share the same (feasible) error target β m^2.

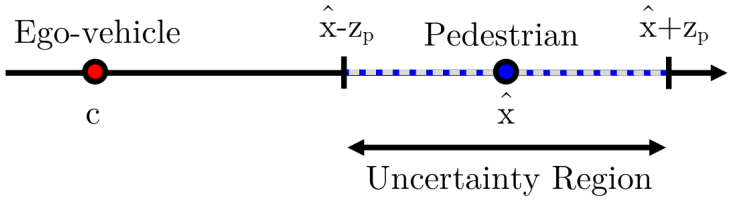


Figure 6.5: Pedestrian collision avoidance scenario

The ego-vehicle picks its driving velocity to be the largest one that ensures with high confidence $p < 1$ that it will not collide with the pedestrian, i.e., guaranteeing that it can completely stop before a potential collision. To that end, it constructs a worst-case *uncertainty region* around the estimated pedestrian location $\hat{x}$ with confidence level $2p-1$, of radius $z_p(\beta) = \sqrt{2\beta} \cdot \text{erf}(2p-1)$ meters.

The ego-vehicle at location c on the road can set its safe driving speed v_{S} so as to ensure that it can come to a full stop before entering the uncertainty region, while taking into account the motion of the pedestrian during the vehicle breaking time. It follows that v_{S} satisfies the following equation:

$$\hat{x} - z_p(\beta + \nu^2 t_b(v_{\text{S}})) - c = d_b(v_{\text{S}}) \tag{6.15}$$

where we equate the breaking distance (right-hand side) to the distance between c and the closest point in the uncertainty region (left-hand side), given that the pedestrian keeps moving during the breaking time t_b increasing the radius of the uncertainty region. This equation can be solved numerically and Figure 6.6 illustrates how the resulting v_{S} varies as a function of the target error $\sqrt{\beta}$ for different confidence levels, when the estimated distance between the ego-vehicle and the pedestrian is 15 meters.

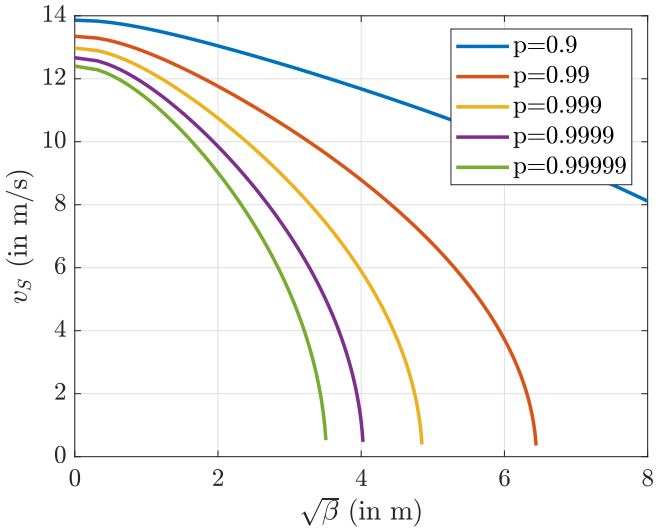


Figure 6.6: Figure of the maximum safe vehicle velocity as a function of the peak local estimation error standard deviation threshold $\sqrt{\beta}$, for $a = 7$ m/s^2, $\nu^2 = 0.5$ m^2/s, $c = \hat{x} - 15$ m.

This figure exhibits an increasing sensitivity of v_{S} to β as p increases, i.e., as the safety requirement becomes tighter. For β large enough, the pedestrian's location uncertainty becomes so large that the vehicle needs to fully stop to avoid a collision, and wait for the pedestrian to eventually cross the

road. When the ego-vehicle opportunistically benefits from our proposed information sharing framework, it can afford reducing its error target β, allowing it to considerably increase its safe driving velocity v_{S}, but at a cost of consuming additional communication resources. The value of information sharing can then be evaluated by studying how v_{S} increases as a function of the communication cost induced by the choice of β (recall that this relationship can be obtained by solving the VISP using VISA, while the one between v_{S} and β has been obtained in Figure 6.6), as depicted in Figure 6.7.

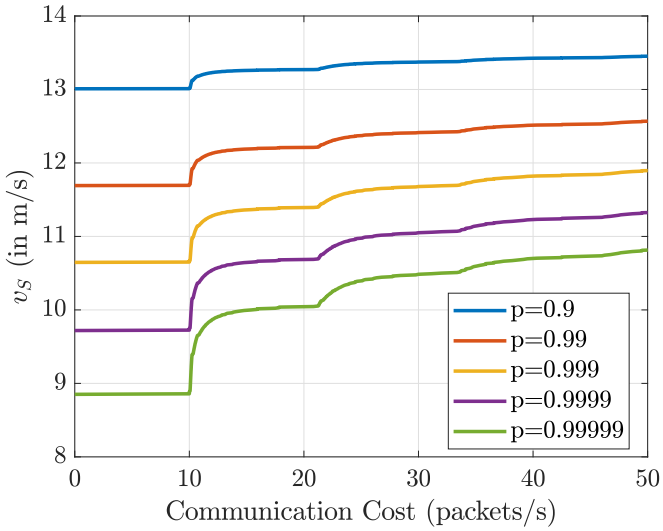


Figure 6.7: Figure of the maximum safe vehicle velocity as a function of the overall communication cost, for $a = 7$ m/s^2, $\nu^2 = 0.5$ m^2/s, $c = \hat{x} - 15$ m, $n = 4$ vehicles, $\sigma^2 = [3^2, 3^2, 3^2, 3^2]$ m^2, $\alpha = [0.75, 0.8, 0.82, 0.85]$, $\tau = 0.1$ s.

One can observe in this figure a general trend of diminishing marginal value (in terms of increasing v_{S}, or equivalently SDT) in the allocated network communication resources allocated for collaborative sensing. Multiple regimes

can be distinguished in this figure as the communication cost budget increases, corresponding to different numbers of contributors in $\mathcal{C}$. In the first regime, i.e., when the communication budget is below 10 packets/s for this example v_{S} is insensitive to an increased communication budget as any budget below $1/\tau$ packets/s would not allow any vehicle to belong to $\mathcal{C}$, making vehicle collaboration unnecessary. In the other regimes, v_{S} exhibits diminishing returns for any fixed $|\mathcal{C}|$. We also observe that most of the benefits of increasing the communication budget originate from the vehicles added earliest to $\mathcal{C}$, consistent with the supermodularity property of $P^*(\mathcal{C})$.

Another important observation is that, consistent with the findings in Figure 6.6, v_{S} is more sensitive to the communication budget for tighter safety constraint. This confirms the considerable benefits that wireless communication-assisted collaborative-sensing systems promise for the design of future vehicular networks.

6.8 Chapter Conclusion

This chapter introduced a novel procedure for collaborative sensing and information sharing in vehicular sensor networks. It distinguishes itself from others as it allows the vehicles to operate autonomously by default, and opportunistically improve their estimates about the environment by participating in the collaborative sensing process. We presented a set of three policies to determine the three key system parameters using the communication resources efficiently, namely the set of data recipients, the feedback rate from the base

station/edge server to the vehicles in that set, and the set of data contributors. We showed how network properties such as the monotonicity and supermodularity of the MF steady-state estimation error variance allows us to devise efficient algorithms with suboptimality guarantees such as VISA, allowing them to be ran in real-time and hence, to adapt to varying network conditions over time. Performance analysis results are promising and show that inter-vehicle collaboration can considerably improve the local vehicles' situational awareness, and hence the safe driving throughput on the roads, making the proposed procedure propitious for adoption in advanced driver-assistance/autonomous systems.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we exhibited how substantial gains can be achieved through efficient collaborative resource management and operations strategies in next generation wireless networks, using mathematical modeling and analysis tools. We focused particularly on two classes of network, namely collaborative vehicular ad-hoc networks, and networks supporting real-time collaborative applications wherein timely information sharing is key to high performance. We showed how different combinations of techniques, including (but not limited to) load balancing, opportunism, rate adaptation, fairness considerations, and suboptimal algorithm design, can be leveraged to improve the quality of service offered to the network users at lower capital and operating costs in a wide variety of networks and scenarios. We envision that the performance improvements suggested in this thesis will allow service providers to rely on these networks to satisfy their network users' ever-increasing demands, and embrace emerging trends in their lifestyle that might shape traffic patterns in future wireless networks (e.g., ride-sharing services, autonomous vehicles, edge-computing supported XR services, multi-user collaborative services).

Some valuable insights can be extracted from both parts of this thesis. In the first part, we demonstrated the benefits of jointly leveraging V2V and V2I connectivity to form vehicle clusters, able to route traffic among each other from/to the network infrastructure. We established that considerable gains are achievable in terms of (1) improved connection reliability, (2) reduced user shared-rate variability, (3) improved mean shared rate per vehicle, (4) improved mean shared rate for non-vehicle-bound users, (5) improved shared rate fairness among network users and (6) improved resilience to spatial traffic surges.

In the second part of this thesis, we showcased the power of effective information rate-adaptation in a wide variety of networks supporting real-time applications. We argued that "*more information is not always beneficial*" since besides the associated communication resource utilization costs, massive information sharing creates network congestion, delaying the time-sensitive information reception/integration, thereby reducing the value of the information being transmitted. We have shown how device-specific rate adaptation techniques can be used to negotiate the associated information timeliness tradeoff. The optimal rate-adaptation strategy is often non-trivially coupled with other decisions that need to be made in real-time, e.g., service placement, or sensor selection, but efficient (possibly suboptimal) algorithms have been be devised to address these problems jointly, while considering the network environment's stochasticity/temporal variability.

7.2 Future Work

The research conducted in this thesis lays the foundations for multiple research directions. We propose three future directions of interest.

First, cluster management mechanisms and protocols need to be devised to ensure the validity of the network performance benefits promised by V2V clustering architecture proposed in the first part of this thesis. Specific problems that need to be addressed include (1) interference management among the V2V links, (2) cluster breakup/merging detection, (3) cluster-head election, (4) incentive mechanisms associated with V2V clustering, and (5) packet delay management as clusters increase in size, possible associated with admission control policies.

Second, as the research conducted in this thesis essentially relies on mathematical modeling and analysis of wireless networks to extract insights and salient features of the studied systems, the presented results are intrinsically contingent on the modeling assumptions made in the successive chapters. Besides the validation of these assumptions in real-world settings, the algorithms devised and described throughout this thesis need to be implemented in real networks and proof-of-concept testbeds. We expect that additional insightful results on the behavior and characteristics of the networks studied can stem from the comparison between the theoretical and practical performances of these algorithms.

Third, this thesis presented network models that were constructed to be

general enough to provide high-level insight on their main features and underlying tradeoffs, yet specific enough to allow for tractable theoretical analysis. It follows that most of the models examined in this thesis can be extended to capture more general or more realistic scenarios. For instance, in the first part of this thesis, vehicular networks with more orderly infrastructure can be investigated (e.g., more regular base station deployment, road placement capturing structures in metropolitan areas, clustered model for mobile devices' placement, etc.). In the second part of the thesis, general graphs could be considered for service placement in the cloud-edge continuum instead of trees, multi-server multi-player games could be studied, and asynchronous vehicle sensing information sharing schemes for multiple correlated environment tracks estimation could be devised. While such systems might be too involved to be modeled and studied analytically, simulation-based experiments could be performed to study such networks and quantify the benefits associated with the features that were not captured in this thesis.

Appendices

Appendix A

Chapter 2 Definitions and Proofs

A.1 Stochastic Ordering Definitions

Definition A.1.1 (Stochastic/Increasing Convex Dominance)**.** *As in [116], we define stochastic dominance as*

$$X \leq^{st} Y \implies P(X > x) \leq P(Y > x),\ \forall x \tag{A.1}$$

and increasing convex dominance

$$X \leq^{icx} Y \implies \mathbb{E}[f(X)] \leq \mathbb{E}[f(Y)],\ \forall f \in \mathcal{F}, \tag{A.2}$$

where $\mathcal{F}$ is the set of increasingly convex functions for which the expected value is defined. Note further that if $X \leq^{icx} Y$ and $\mathbb{E}[X] = \mathbb{E}[Y]$ then $X \leq^{cx} Y$ for convex functions, e.g., $Var(X) \leq Var(Y)$.

A.2 Proof Lemma 2.4.2

Proof. We denote as φ the probability of not having any V2V capable vehicle in the communication range d ahead of a typical vehicle. From the Poisson assumption, the distance between vehicles follows an exponential distribution denoted by a random variable $E \sim \exp(\lambda_v)$. The probability of having one or more vehicles within the communication range d of a participant is then:

$F_E(d) = 1 - e^{-\lambda_v d}$. Since the market penetration is considered independent of the interarrival time, the probability of the next car being a V2V+V2I capable vehicle within the communication range d is given by $\gamma(1 - e^{-\lambda_v d})$, thus $\varphi = 1 - \gamma(1 - e^{-\lambda_v d})$.

Now since the number of users in a cluster is determined by the number of the successive V2V capable vehicles in range of each other, $p_N(n) = \varphi\,(1-\varphi)^{n-1}$, i.e., N is a geometric random variable with parameter φ, and mean $\mathbb{E}[N] = 1/\varphi$. □

A.3 Proof Lemma 2.4.3

Proof. From the analysis in [166], it is well known that the average cluster communication range is $\mathbb{E}[L] = {\lambda_v}^{-1} \cdot \left(e^{\lambda_v d} - \lambda_v d - 1\right)$ (defined as distance between the first and the last vehicle plus 2 times the communication range). However, the density function of the length has been only evaluated via simulations [136]. The length of the cluster, given that there are N vehicles, corresponds to $L = 2d + \sum_{i=1}^{N-1} T_i$, where T_i denotes the inter-spacing of *V2V* capable vehicles in the same cluster. Note that the distribution of T_i is that of an exponential conditioned on being smaller than d, thus

$$f_{T_i}(l) = \frac{\lambda e^{-\lambda_v l}}{1 - e^{-\lambda_v d}}, \quad 0 < l \leq d. \tag{A.3}$$

The moment generating function for T_i is thus

$$M_{T_i}(s) = \int_0^d e^{sl} \frac{\lambda e^{-\lambda_v l}}{1 - e^{-\lambda_v d}} dl = \frac{\lambda_v e^{d(s-\lambda_v)} - \lambda_v}{(s - \lambda_v)(1 - e^{-\lambda_v d})} \tag{A.4}$$

and consequently the conditional moment generating function of the length of a cluster, given its number of vehicles, denoted as $M_{L|N=n}(s)$ is given by:

$$M_{L|N=n}(s) = e^{2sd} \prod_{i=1}^{n-1} M_{T_i}(s) = e^{2sd} \left[\frac{\lambda_v e^{d(s-\lambda_v)} - \lambda_v}{(s-\lambda_v)(1-e^{-\lambda_v d})} \right]^{n-1}. \tag{A.5}$$

Given the conditional distribution, we can compute the moment generating function of L via:

$$M_L(s) = \sum_{n=1}^{\infty} M_{L|N=n}(s) p_N(n) = \frac{e^{2sd}\varphi}{1 - M_T(s) + \varphi M_T(s)}. \tag{A.6}$$

For the case of full market penetration, this simplifies to:

$$M_L(s) = \frac{e^{d(2s-\lambda_v)}\,(s-\lambda_v)}{s - \lambda_v e^{d(s-\lambda_v)}}. \tag{A.7}$$

The distributions $f_L(l)$ and $f_{L|N}(l)$ can be then obtained by the inverse Laplace transform of $M_L(-s)$ and $M_{L|N=n}(-s)$, respectively. □

A.4 Proof Lemma 2.4.4

Proof. The conditional c.d.f. of the number of RSUs M serving a cluster of length L is given by

$$F_{M|L}(m \mid L = l) = \begin{cases} 1 & \text{if } m\lambda_r^{-1} < l, \\ 1 - \frac{l}{m \cdot \lambda_r^{-1}} & \text{if } (m-1)\lambda_r^{-1} < l \leq m \cdot \lambda_r^{-1}, \\ 0 & \text{otherwise} \end{cases} \tag{A.8}$$

since the cluster process is stationary on the line and independent of the RSU locations. Given N it is direct to see, applying the chain rule that:

$$F^c_{M|N}(m \mid N = n) = \int_0^{\infty} F^c_{M|L}(m \mid L = l) f_{L|N}(l \mid N = n) dl. \tag{A.9}$$

where $F^c(\cdot)$ stands for the complementary c.d.f.. Substituting the distributions obtained above, the simplified expression is given by Equation 2.7. □

A.5 Proof Lemma 2.5.3

Proof. In order to derive an expression for the probability of coverage of a typical vehicle π_v, we will first relate the number of vehicles as seen in a typical cluster N to that seen by a typical vehicle in its cluster N_v :

$$p_{N_v}(n) = \frac{nP(N=n)}{\mathbb{E}[N]}, \tag{A.10}$$

represents the probability for a typical vehicle to be in a cluster of size n, where the $\frac{n}{\mathbb{E}[N]}$ biases the distribution of N as a typical vehicle is more likely to belong to larger clusters. Therefore,

$$\pi_v = \sum_{n=1}^{\infty} p_{N_v}(n) F^c_{M|N}\left(0 \mid N=n\right) = \varphi^2 \sum_{n=1}^{\infty} n(1-\varphi)^{n-1} F^c_{M|N}\left(0 \mid N=n\right). \tag{A.11}$$

In the case with networks with only *V2I* links enabled, the probability that a typical vehicle is connected corresponds to the probability that the vehicle lands in the fraction of the road covered by RSUs given by: $\pi_v^* = \frac{2d}{\lambda_r^{-1}}$. □

A.6 Proof Theorem 2.5.4

Proof. Under our sharing model, vehicles in a typical cluster with (N, M) users and RSU connections will see a shared rate no larger than $\frac{\rho^{RSU} M}{N}$. This is exact if the cluster does not share any of the RSUs with another cluster; otherwise this is an upper bound. Note that an RSU can be shared by two clusters,

each approaching from one side, and both not close enough to form one larger cluster. The mean rate seen by a typical vehicle R_v is then bounded by:

$$\mathbb{E}[R_v] \leq \mathbb{E}\left[\frac{N}{\mathbb{E}[N]}\frac{\rho^{RSU}}{N}M\right] = \frac{\rho^{RSU}\cdot\mathbb{E}[M]}{\mathbb{E}[N]} \tag{A.12}$$

where once again we have moved from the typical cluster shared rate to the typical vehicle shared rate by weighting by $N/\mathbb{E}[N]$.

In the V2I-only setting, the typical vehicle's rate is:

$$\mathbb{E}[R_v^*] = \mathbb{E}\left[\frac{\rho^{RSU}}{N^*+1} \mid I_v^*\right]\pi_v^* = \rho^{RSU}\frac{2d}{\lambda_r^{-1}}\mathbb{E}\left[\frac{1}{N^*+1}\right] \tag{A.13}$$

where I_v^* denotes the event of probability π_v^* that a vehicle is connected, and N^* denotes the number of (other) vehicles that a typical connected vehicle would see sharing its RSU. Note that the distribution of N^*, i.e., the reduced Palm distribution of the Poisson, is equal to its original distribution (Poisson($2d\gamma\lambda_v$)), given the Slivnyak's theorem [14]. Therefore $\mathbb{E}\left[\frac{1}{N^*+1}\right] = \sum_{n=0}^{\infty}\frac{P(N^*=n)}{n+1} = \frac{1-e^{-2\gamma\lambda_v d}}{2\gamma\lambda_v d}$ and $\mathbb{E}[R_v^*] = \rho^{RSU}\cdot\frac{1-e^{-2\gamma\lambda_v d}}{\lambda_v\gamma\lambda_r^{-1}}$.

Finally, by coupling the vehicle locations for the V2V+V2I network and V2I network without relaying it is easy to observe that the number of busy RSUs is the same, so the mean rate seen by a typical vehicle in this two settings is the same, i.e., $\mathbb{E}[R_v] = \mathbb{E}[R_v^*]$. □

A.7 Proof Theorem 2.5.5

Proof. Paralleling Theorem 2.5.4, an upper bound on the complementary CDF of the shared rate a typical vehicle sees in the V2V+V2I architecture, for $r > 0$

is given by:

$$F^c_{R_v}(r) = P(R_v > r) \leq \mathbb{E}\left[\frac{N}{\mathbb{E}[N]}\mathbb{E}\left[\mathbf{1}\left(\frac{M\rho^{\mathrm{RSU}}}{N} \geq r\right) \mid N\right]\right] \tag{A.14}$$

$$= \sum_{n=1}^{\infty} \frac{n}{\mathbb{E}[N]} \cdot p_N(n) \cdot F^c_{M|N}\left(\left\lceil \frac{rn}{\rho^{\mathrm{RSU}}} \right\rceil \mid N = n\right) \tag{A.15}$$

$$= \varphi^2 \sum_{n=1}^{\infty} n \cdot (1-\varphi)^{n-1} \cdot F^c_{M|N}\left(\left\lceil \frac{rn}{\rho^{\mathrm{RSU}}} \right\rceil \mid N = n\right), \tag{A.16}$$

where $F^c(\cdot)$ stands for the complementary c.d.f. Therefore, Equation 2.12 holds and $P(R_v = 0) = 1 - \pi_v$. Similarly the complementary c.d.f. for the shared rate for a typical vehicle in the V2I network, for $r > 0$ is

$$P(R^*_v > r) = P\left(\frac{\rho^{RSU}}{N^* + 1} > r\right)\frac{d}{\lambda_r^{-1}} \tag{A.17}$$

$$= P\left(\frac{\rho^{RSU} - r}{r} > N^*\right)\frac{d}{\lambda_r^{-1}} \tag{A.18}$$

$$= \frac{2d}{\lambda_r^{-1}} \sum_{i=0}^{\lfloor \frac{\rho^{RSU}-r}{r} \rfloor} \left(\frac{(2\gamma\lambda_v d)^i}{i!} e^{-2\gamma\lambda_v d}\right) \tag{A.19}$$

$$= \frac{2d}{\lambda_r^{-1}} \cdot Q\left(\frac{\rho^{RSU} - r}{r}, 2\gamma\lambda_v d\right) \tag{A.20}$$

where $Q(.)$ is the regularized gamma function. Consequently, Equation 2.13 holds and $P(R^*_v = 0) = 1 - \pi^*_v$. In order to prove the increasing convex dominance relation, we can use a coupling argument. We generate a single lane highway instance. It is clear that, for this instance, the number of vehicles and the total rate out of the network is the same, but the clusters are bigger in the V2V+V2I system (since the V2I only system only have clusters of one vehicle). It is proven in [129] that a max-min fairness allocation achieves the lexicographically minimum vector, i.e., for a max-min share rate allocation

$\hat{R}$ and any other shared rate allocation R then $\hat{R}$ *is majorized* by R [68] and further implies $\hat{R} \leq^{icx} R$. The proof is then completed by noticing that the max-min shared rate allocation of the V2I system R^* is a feasible rate allocation in the V2V+V2I system, so $R \leq^{icx} R^*$. □

A.8 Proof Theorem 2.6.3

Proof. The proof relies on constructing a coupling between a random process $\xi^{\mathcal{M}}$ denoting vehicle locations on a multilane highway $\mathcal{M}$ and an auxiliary process $\xi^{\mathcal{S}}$ denoting their locations on a single lane highway $\mathcal{S}$.

Let $(T_i, K_i)_{i\in\mathbb{N}}$ denote the sequence of locations of V2V+V2I capable vehicles on $\mathcal{M}$, where T_i denotes the location of the i^{th} vehicle and K_i its associated lane. We define the aggregated V2V+V2I capable vehicle intensity on highway $\mathcal{M}$ as λ^{V2V}, and their intensity on lane k as λ_k^{V2V}. Note that under our Poisson assumption $(T_i)_{i\in\mathbb{N}}$ is a PPP(λ^{V2V}) and $(K_i)_{i\in\mathbb{N}}$ is distributed as

$$p_{K_i}(k) = \left(\frac{\lambda_k^{\text{V2V}}}{\lambda^{\text{V2V}}} : k = 1, 2, \ldots, \eta\right), \forall i \in \mathbb{N}, \tag{A.21}$$

since aggregation of independent PPPs is also a PPP.

The first step of the coupled single-lane highway $\mathcal{S}$ construction consists in including V2V+V2I capable vehicles at locations $(T_i)_{i\in\mathbb{N}}$ in the auxiliary process $\xi^{\mathcal{S}}$.

Let us now consider the blocking vehicles in the multilane highway $\mathcal{M}$. These vehicles also correspond to PPPs of intensity λ_k^b on each lane k and independent of $(T_i, K_i)_{i\in\mathbb{N}}$. For a given realization $(t_i, k_i)_{i\in\mathbb{N}}$ let $B^{k_i}(t_i, t_{i+1}]$

denote a set of locations of blocking vehicles on lane k in the time interval $(t_i, t_{i+1}]$ in the multilane highway. Note that $B^{k_i}(t_i, t_{i+1}]$ for any i are mutually independent by the definition of PPP.

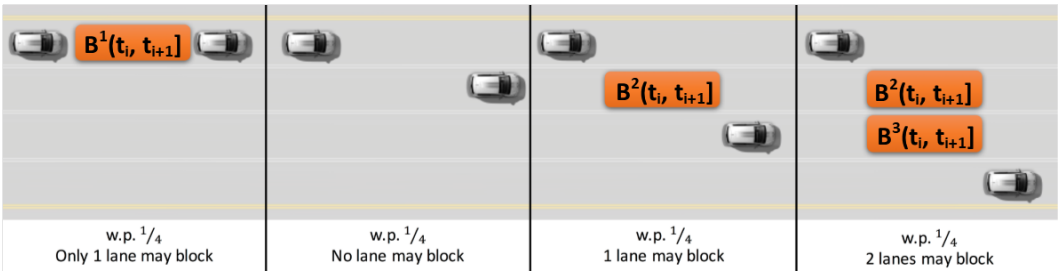


Figure A.1: Examples of configurations and their associated $B^i(.,.]$ sets, for $\eta = 4$ and $k_i \leq k_{i+1}$.

We shall let $B(t_i, t_{i+1}]$ denote blocking vehicles' locations that the process $\mathcal{M}$ will share with $\mathcal{S}$. Specifically, according to the blocking model in Definition 2.6.2 we let

$$B(t_i, t_{i+1}] = \begin{cases} \bigcup_{j=\mathrm{k}_i+1}^{\mathrm{k}_{i+1}-1} B^j(t_i, t_{i+1}] & \text{if } \mathrm{k}_{i+1} > j > \mathrm{k}_i, \\ B^j(t_i, t_{i+1}] & \text{if } \mathrm{k}_i = \mathrm{k}_{i+1} = j, \\ \bigcup_{j=\mathrm{k}_{i+1}+1}^{\mathrm{k}_i-1} B^j(t_i, t_{i+1}] & \text{if } \mathrm{k}_i > j > \mathrm{k}_{i+1}, \\ \emptyset & \text{otherwise.} \end{cases} \tag{A.22}$$

Note that in each interval $(t_i, t_{i+1}]$, $B(t_i, t_{i+1}]$ are Poisson process independent but with different intensities depending on k_i and k_{i+1}. Note also that given our blocking model $B(\cdot, \cdot]$ includes all vehicles that may block connectivity of V2V+V2I capable cars in $\mathcal{M}$. Figure A.1 shows examples of configurations and their associated $B(.,.]$. Finally, for each $i \in \mathrm{N}$ we define

$$B^{\mathcal{S}}(t_i, t_{i+1}] = B(t_i, t_{i+1}] \cup A(t_i, t_{i+1}] \tag{A.23}$$

where $A(t_i, t_{i+1}]$ is an independent PPP on $(t_i, t_{i+1}]$ with intensity needed to ensure that the overall intensity is equalized in all intervals; ensuring that $B^{\mathcal{S}}(t_i, t_{i+1}]$ is a PPP with intensity λ^b_{eff}. We shall introduce $B^{\mathcal{S}}(t_i, t_{i+1}]_{i\in\mathbb{N}}$ in each of the intervals in the process $\xi^{\mathcal{S}}$.

At this point, it is worth noting that given our construction,

$$\text{LoS interrupted in } \xi^{\mathcal{M}} \quad \begin{matrix} \Longrightarrow \\ \not\Longleftarrow \end{matrix} \quad \text{LoS interrupted in } \xi^{\mathcal{S}}. \tag{A.24}$$

and the distributions of $\xi^{\mathcal{M}} \sim \mathcal{M} = \mathcal{H}(\eta, \boldsymbol{\lambda}^{\text{V2V}},\ \boldsymbol{\lambda^b})$ and $\xi^{\mathcal{S}} \sim \mathcal{S} = \mathcal{H}(1,\ \gamma\lambda, \lambda^b_{\text{eff}})$ where $\lambda = \lambda^{\text{V2V}} + \lambda^b,\ \gamma = \frac{\lambda^{\text{V2V}}}{\lambda}$ and $\lambda^b_{\text{eff}} = \max(\lambda^b_0, \lambda^b_k, \sum_{i=2}^{\eta-1} \lambda^b_i)$.

This implies the following fact.

Fact A.8.1. *Based on the aforementioned coupling one can show that* $N_v^{\mathcal{M}} \geq^{st} N_v^{\mathcal{S}}, \quad L_v^{\mathcal{M}} \geq^{st} L_v^{\mathcal{S}} \quad$ *and* $\quad M_v^{\mathcal{M}} \geq^{st} M_v^{\mathcal{S}}$, *and,* $\pi_v^{\mathcal{M}} \geq \pi_v^{\mathcal{S}}, \quad R_v^{\mathcal{M}} \leq^{icx} R_v^{\mathcal{S}}$.

Proof. Note that by ergodicity of the cluster process, $P(N_v^{\mathcal{M}} > n)$ and $P(N_v^{\mathcal{S}} > n)$ correspond to:

$$P(N_v^{\mathcal{M}} > n) = \lim_{c\to\infty} \frac{1}{\sum_{i=1}^{c} N_i^{\mathcal{M}}} \sum_{i=1}^{c} N_i^{\mathcal{M}} \cdot \mathbb{1}(N_i^{\mathcal{M}} > n) \tag{A.25}$$

$$P(N_v^{\mathcal{S}} > n) = \lim_{c\to\infty} \frac{1}{\sum_{i=1}^{c} N_i^{\mathcal{M}}} \sum_{i=1}^{c} \sum_{j=1}^{Y_i} N_{i,j}^{\mathcal{S}} \cdot \mathbb{1}(N_{i,j}^{\mathcal{S}} > n), \tag{A.26}$$

where $N_i^{\mathcal{M}}$ is the number of vehicles in the i^{th} cluster in the multilane and Y_i is the number of subclusters in the single lane originated from the i^{th} cluster in the multilane. $N_{i,j}^{\mathcal{S}}$ denotes the number of vehicles in the j^{th} subcluster in the single lane process.

By noting that the clusters in $\mathcal{S}$ are created by splitting the clusters of $\mathcal{M}$, we can see that

$$\mathbb{1}(N_i^{\mathcal{M}} > n) \geq \mathbb{1}(N_{i,j}^{\mathcal{S}} > n), \quad \forall i, j, \tag{A.27}$$

and given the fact that $N_i^{\mathcal{M}} = \sum_{j=1}^{Y_i} N_{i,j}^{\mathcal{S}}$, we have that

$$P(N_v^{\mathcal{M}} > n) = \lim_{c\to\infty} \frac{1}{\sum_{i=1}^{c} N_i^{\mathcal{M}}} \sum_{i=1}^{c}\sum_{j=1}^{Y_i} N_{i,j}^{\mathcal{S}} \cdot \mathbb{1}(N_i^{\mathcal{M}} > n) \tag{A.28}$$

$$\geq \lim_{c\to\infty} \frac{1}{\sum_{i=1}^{c} N_i^{\mathcal{M}}} \sum_{i=1}^{c}\sum_{j=1}^{Y_i} N_{i,j}^{\mathcal{S}} \cdot \mathbb{1}(N_{i,j}^{\mathcal{S}} > n) \tag{A.29}$$

$$= P(N_v^{\mathcal{S}} > n) \tag{A.30}$$

and therefore $N_v^{\mathcal{M}} \geq^{st} N_v^{\mathcal{S}}$. Similarly, $L_v^{\mathcal{M}} \geq^{st} L_v^{\mathcal{S}}$ and $M_v^{\mathcal{M}} \geq^{st} M_v^{\mathcal{S}}$ by noting that $\mathbb{1}(L_{v,i}^{\mathcal{M}} > l) \geq \mathbb{1}(L_{v,i,j}^{\mathcal{S}} > l)$ and $\mathbb{1}(M_{v,i}^{\mathcal{M}} > m) \geq \mathbb{1}(M_{v,i,j}^{\mathcal{S}} > m)$ are direct implications of Equation A.24. Additionally it also has the implication that, within a cluster, if we denote as $\pi_{v,i}$ the probability that a typical vehicle in cluster ith is connected then $\pi_{v,i}^{\mathcal{M}} \geq \pi_{v,i,j}^{\mathcal{S}}$.

Noting that $N_i^{\mathcal{M}} = \sum_{j=1}^{Y_i} N_{i,j}^{\mathcal{S}}$ and observing that the expected shared rate per vehicle is equal in both systems we can directly infer the $R_v^{\mathcal{M}} \leq^{icx} R_v^{\mathcal{S}}$.

It is proven in [129] that a max-min fairness allocation achieves the lexicographically minimum vector, i.e., for a max-min share rate allocation $\hat{R}$ and any other shared rate allocation R then $\hat{R}$ *is majorized* by R [68] and further implies $\hat{R} \leq^{icx} R$. The max-min shared rate allocation of the single lane system $R^{\mathcal{S}}$ is always a feasible rate allocation in the multilane system; since the single lane system has the same number of vehicles and the same mean rate, but the ability for the vehicles to reach the RSUs is reduced and we have that $R_v^{\mathcal{M}} \leq^{icx} R_v^{\mathcal{S}}$. □

□

Appendix B

Chapter 3 Proofs

B.1 Proof Theorem 3.5.1

Proof. Figure B.1a exhibits the geometry of vehicular cluster-based opportunistic relaying. The distance D between a typical vehicle on the x-axis and its closest BS assumed without loss of generality to be at the origin follows the distribution in Equation 3.7. The typical vehicle belongs to a cluster of size Z^* vehicles whose distribution corresponds to the size biased distribution of the typical cluster length Z such that $p_{Z^*}(z) = \frac{z \cdot p_Z(z)}{\mathbb{E}[Z]}$, for $z \in \mathbb{N}$, i.e., typical vehicles are more likely to belong to longer clusters. The typical vehicle's cluster size induces a typical vehicle's cluster length L^* (in meters), such that $L^* = (Z^* - 1) \cdot d_{\mathrm{V}}$. The random orientation Θ of the typical vehicle's cluster (acute angle) is such that $\Theta \sim \mathrm{Unif}[0, \frac{\pi}{2}]$ which is independent of Z^*, L^* and D.

To prove Theorem 3.5.1, we note that the location of a typical vehicle within its cluster is uniformly distributed, and "breaks" its cluster into two fragments. We denote as $Z^{*,b}$ and $L^{*,b}$ the size and length of the typical vehicle's cluster fragment "pointing in the direction of the BS", where there may be candidate opportunistic relays with better channels to the BS at the

origin. The distribution of $Z^{*,b}$ is shown to be:

Lemma B.1.1 (Typical Cluster Size Distribution)**.** *Given a typical cluster size Z distribution $p_Z(\cdot)$, the distribution of $Z^{*,b}$ is*

$$p_{Z^{*,b}}(z) = \frac{\mathbb{P}(Z \geq z)}{\mathbb{E}[Z]}, \qquad z \in \mathbb{N} \tag{B.1}$$

Proof. As the typical vehicle can be any one in its cluster with the same probability, i.e., $p_{Z^{*,b}|Z^*}(z|Z^* = i) = 1/i, z = 1, \cdots, i$, the distribution of $Z^{*,b}$ is

$$p_{Z^{*,b}}(z) = \sum_{i=1}^{\infty} p_{Z^{*,b}|Z^*}(z|Z^* = i) \cdot p_{Z^*}(i), \ \forall z \in \mathbb{N} \tag{B.2}$$

$$= \sum_{i=z}^{\infty} \frac{1}{i} \cdot \frac{i \cdot p_Z(i)}{\mathbb{E}[Z]}, \ \forall z \in \mathbb{N} \tag{B.3}$$

$$= \frac{P(Z \geq z)}{\mathbb{E}[Z]}, \ \forall z \in \mathbb{N} \tag{B.4}$$

□

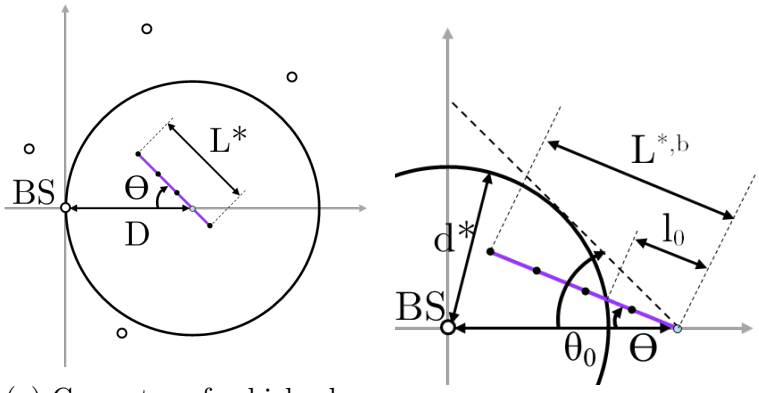
The distribution of $L^{*,b}$ directly follows from the relation $L^{*,b} = (Z^{*,b} - 1) \cdot d_{\mathrm{V}}$, giving

$$p_{L^{*,b}}(l) = \frac{P(Z \geq \frac{l}{d_{\mathrm{V}}} + 1)}{\mathbb{E}[Z]}, \quad l = 0, d_{\mathrm{V}}, 2 \cdot d_{\mathrm{V}}, \cdots. \tag{B.5}$$

We seek to determine the distribution of the minimum distance D^* between a relay vehicle on the cluster of length $L^{*,b}$ and the BS at the origin with the additional requirement that the relay vehicle also belongs to the typical vehicle's cell, conditional on the distance $D = d$ between the typical

vehicle and the BS. Note that $D^* \leq D$ almost surely, since a typical vehicle can of course receive data directly from its closest BS.

Figure B.1b exhibits the definition of two key functions of the geometry: (1) $\theta_0(d, d^*)$ the angle of the tangent to a disc of radius d^*, and (2) $l_0(d, d^*, \theta)$ which for $\Theta \leq \theta_0(d, d^*)$ is the length of the segment starting from $(d, 0)$ with angle θ to the disc of radius d^*.



(a) Geometry of vehicle cluster based opportunistic relaying.

(b) Illustration of d^*, $\theta_0(d, d^*)$, $l^{*,b}$ and $l_0(d, d^*, \theta)$.

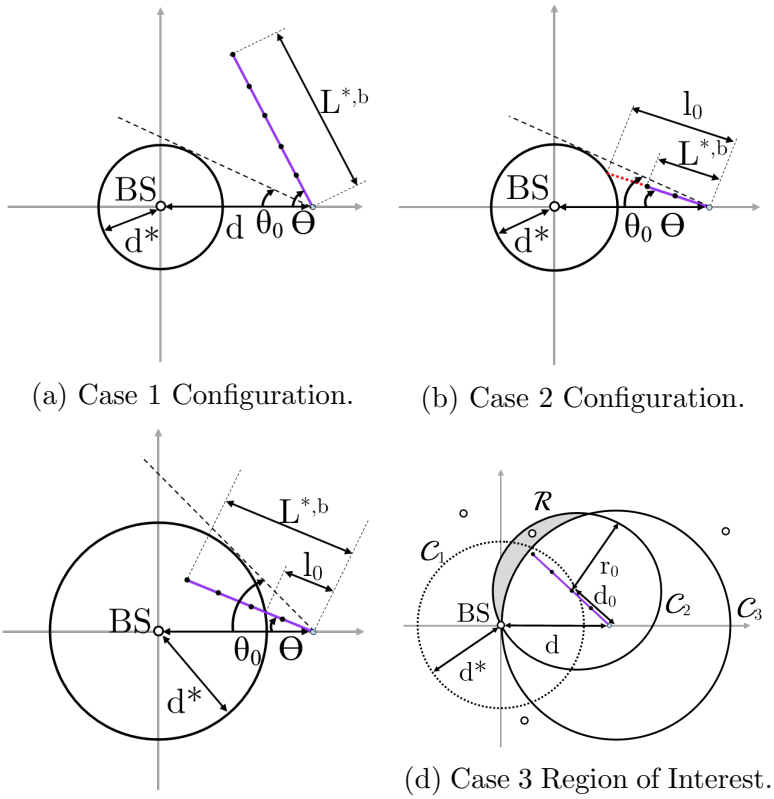
Figure B.1: Geometry of the typical vehicle's cluster and environment.

With these definitions one can evaluate $\mathbb{P}(D^* \leq d^* | D = d)$ by identifying a partition $\mathcal{E}_1, \mathcal{E}_2$ and $\mathcal{E}_3$ corresponding to the three cases/events exhibited in Figure B.2 and given by :

- Case 1: $\mathcal{E}_1 = \{\Theta > \theta_0(d^*, d)\}$
- Case 2: $\mathcal{E}_2 = \{\Theta \leq \theta_0(d^*, d),\ L^{*,b} < l_0(d, d^*, \Theta)\}$
- Case 3: $\mathcal{E}_3 = \{\Theta \leq \theta_0(d^*, d),\ L^{*,b} \geq l_0(d, d^*, \Theta)\}$

In general we have from independence of Θ and $L^{*,b}$:

$$\begin{aligned}&\mathbb{P}(D^* \leq d^* | D = d) \\ &= \int_0^{\frac{\pi}{2}} \sum_{i=0}^{\infty} \mathbb{P}(D^* \leq d^* | D = d, \Theta = \theta, L^{*,b} = i \cdot d_{\mathrm{V}}) p_{L^{*,b}}(i \cdot d_{\mathrm{V}}) f_{\Theta}(\theta) d\theta \end{aligned} \quad \text{(B.6)}$$



(a) Case 1 Configuration.

(b) Case 2 Configuration.

(c) Case 3 Configuration.

(d) Case 3 Region of Interest.

Figure B.2: Typical Vehicle's Cluster Configuration Analysis

We consider the three cases individually.

Case 1. For a given d, d^* the critical angle, i.e., the angle of the tangent line to the circle of radius d^*, is given by

$$\theta_0(d, d^*) \triangleq \sin^{-1}(d^*/d) \quad \text{(B.7)}$$

and note from Figure B.2a that if $\Theta \geq \theta_0(d, d^*)$ then the cluster does not hit the radius d^* disc, whence $D^* > d^*$, and $\mathbb{P}(D^* \leq d^* | \mathcal{E}_1, D = d) = 0$.

Case 2. For a given d, d^* and $\Theta = \theta < \theta_0(d, d^*)$ note from Figure B.2b that a cluster extending a length $L^{*,b} = l$ where

$$l \leq l_0(d, d^*, \theta) \triangleq d\cos(\theta) - \sqrt{d^{*2} - (d\sin(\theta))^2} \tag{B.8}$$

will not hit the disc of radius d^*, whence $D^* > d^*$. Here $l_0(d, d^*, \theta)$ is determined by studying the triangle of side lengths l_0, d and d^*, knowing θ. Therefore, here again, $\mathbb{P}(D^* \leq d^* | \mathcal{E}_2, D = d) = 0$.

Case 3. The last case corresponds to event $\mathcal{E}_3$ illustrated in Figure B.2c. Given $d, d^*, \Theta \leq \theta = \theta_0(d, d^*)$ and $L^{*,b} = l \geq l_0(d, d^*, \theta)$, the vehicle cluster extends into the circle $\mathcal{C}_1$ of radius d^*. In order for $D^* \leq d^*$, two conditions must be true: (1) at least one vehicle in the cluster must be in the disk of radius d^*, and (2) none of the vehicles within that disk must be closer to another BS than the one at the origin.

The first condition can be shown to be equivalent to

$$\left\lfloor \frac{l_0}{d_{\mathrm{V}}} \right\rfloor \neq \left\lfloor \frac{l_0 + 2\sqrt{d^{*2} - (d\sin(\theta))^2}}{d_{\mathrm{V}}} \right\rfloor, \tag{B.9}$$

i.e., when the two intersections between the cluster fragment and the circle of radius d^* occur between two different pairs of consecutive vehicles. If this condition does, not hold, then $\mathbb{P}(D^* \leq d^* | \mathcal{E}_2, D = d) = 0$.

The second condition is related to the scenario illustrated in Figure B.2d where we draw two additional circles. The first is $\mathcal{C}_2$ whose center the closest

vehicle from the one at $(d, 0)$ that lies within $\mathcal{C}_1$ and whose radius is its distance to the BS at the origin. The second, $\mathcal{C}_3$, is centered at the typical vehicle $(d, 0)$ and has radius d, i.e. also crosses the origin. Recalling that the origin is the location of the closest BS to $(d, 0)$, and thus $\mathcal{C}_3$ contains no other base stations. A necessary and sufficient condition to ensure that at least one vehicle in the cluster fragment within $\mathcal{C}_1$ is associated with the BS at the origin is that there are no BSs in the shaded region $\mathcal{R}(d, d^*, \theta)$ representing all locations that are closer to the first vehicle in $\mathcal{C}_1$ than to the origin. This follows because

- if there is a BS in $\mathcal{R}(d, d^*, \theta)$ then not only will the first cluster vehicle in $\mathcal{C}_1$ not be associated with b, but so will all the others, since the circle centered on each such vehicle and traversing the origin, will contain $\mathcal{R}(d, d^*, \theta)$. We can then conclude that $D^* > d^*$.

- if $\mathcal{R}(d, d^*, \theta)$ is empty, then at least one vehicle in the cluster is less than d^* meters away from b, and will associate with it, i.e. $D^* \leq d^*$.

Using basic algebra and the law of cosines one can show that the distance d_0 between the typical vehicle and the first cluster vehicle in $\mathcal{C}_1$ is given by

$$d_0(d, d^*, \theta) \triangleq d_{\mathrm{V}} \cdot \left\lceil \frac{l_0(d, d^*, \theta)}{d_{\mathrm{V}}} \right\rceil \tag{B.10}$$

and the distance r_0 from that vehicle to the origin (i.e., radius of $\mathcal{C}_2$) is given by

$$r_0(d, d^*, \theta) \triangleq \sqrt{d^2 + d_0^2 - 2d \cdot d_0 \cos(\theta)}. \tag{B.11}$$

where we have suppressed the arguments of d_0 for conciseness. Using the expression in [165] for the area of $\mathcal{C}_2 \cap \mathcal{C}_3$, as a function of d, d_0 and r_0, one can find an expression for the area a_0 of $\mathcal{R}(d, d^*, \theta)$:

$$a_0(d, d^*, \theta) \triangleq \pi r_0^2 - \left[r_0^2 \cos^{-1}\left(\frac{d_0^2 + r_0^2 - d^2}{2 d_0 r_0}\right) + d^2 \cos^{-1}\left(\frac{d_0^2 + d^2 - r_0^2}{2 d_0 d}\right) - \frac{\sqrt{(-d_0 + r_0 + d)(d_0 + r_0 - d)(d_0 - r_0 + d)(d_0 + r_0 + d)}}{2} \right] \tag{B.12}$$

where we have suppressed the arguments of d_0 and r_0 for conciseness. Since BSs form a PPP, the probability there is no BS in $\mathcal{R}$ is given by

$$\mathbb{P}(\Phi_{\mathrm{BS}} \cap \mathcal{R}(d, d^*, \theta) = \emptyset) = e^{-\lambda_{\mathrm{BS}} a_0(d, d^*, \theta)}. \tag{B.13}$$

Now considering the two necessary and sufficient conditions to ensure that at least one vehicle in the cluster fragment is within a distance d^* to the origin, we get

$$\begin{aligned} & e_0(d, d^*, \theta) \triangleq \mathbb{P}(D^* \leq d^* | \mathcal{E}_3, D = d) \\ & = e^{-\lambda_{\mathrm{BS}} a_0(d, d^*, \theta)} \cdot \mathbb{1}\left\{ \left\lfloor \frac{l_0}{d_{\mathrm{V}}} \right\rfloor \neq \left\lfloor \frac{l_0 + 2\sqrt{d^{*2} - (d \sin(\theta))^2}}{d_{\mathrm{V}}} \right\rfloor \right\} \end{aligned} \tag{B.14}$$

Therefore, Equation B.6 now reduces to

$$\begin{aligned} & \mathbb{P}(D^* \leq d^* | D = d) \\ & = \int_0^{\theta_0(d, d^*)} \sum_{i = \lceil \frac{l_0(d, d^*, \theta)}{d_{\mathrm{V}}} \rceil}^{\infty} e_0(d, d^*, \theta) p_{L^{*,b}}(i \cdot d_{\mathrm{V}}) f_{\Theta}(\theta) d\theta \end{aligned} \tag{B.15}$$

$$= \int_0^{\theta_0(d, d^*)} \mathbb{P}(L^{*,b} \geq d_0(d, d^*, \theta)) \cdot e_0(d, d^*, \theta) f_{\Theta}(\theta) d\theta \tag{B.16}$$

where $f_\Theta(\theta) = \frac{1}{\pi/2}$, for $\theta \in [0, \pi/2]$. The result follows from this expression and Lemma B.1.1. $\square$

B.2 Proof Theorem 3.6.1

Proof. The sequential algorithm is executed in $|\phi_{\mathrm{V},c}|$ steps, where one seeks to associate at step t one additional vehicle to one of the serving BSs that provides the largest marginal improvement in cluster utility given that $t-1$ vehicles have already been associated. We shall prove the theorem by contradiction. We shall need the following two definitions.

- Define $\Delta_c^b(\mathbf{n})$ to be the change in cluster utility if an additional vehicle in cluster c were to associate to BS b. More specifically, $\Delta_c^b(\mathbf{n}) = \mathcal{L}_{c,\alpha}(\mathbf{n} + \mathbf{e}_b) - \mathcal{L}_{c,\alpha}(\mathbf{n}) = (n^b+1)\cdot \mathcal{U}_\alpha\left(\frac{r_c^{b,*}}{(n^b+1)+k_c^b}\right) - n^b \cdot \mathcal{U}_\alpha\left(\frac{r_c^{b,*}}{n^b+k_c^b}\right)$. Note that $\Delta_c^b(\mathbf{n})$ is decreasing in the entries of $\mathbf{n}$ as $\mathcal{L}_{c,\alpha}(\mathbf{n})$ is concave in these entries.

- Similarly, define $\Delta_c^{-b}(\mathbf{n})$ to be the change in cluster utility if a vehicle in cluster c associated to BS b were to be removed. We have, $\Delta_c^{-b}(\mathbf{n}) = \mathcal{L}_{c,\alpha}(\mathbf{n} - \mathbf{e}_b) - \mathcal{L}_{c,\alpha}(\mathbf{n}) = (n^b-1)\cdot \mathcal{U}_\alpha\left(\frac{r_c^{b,*}}{(n^b-1)+k_c^b}\right) - n^b \cdot \mathcal{U}_\alpha\left(\frac{r_c^{b,*}}{n^b+k_c^b}\right)$. Note that $\Delta_c^{-b}(\mathbf{n}) = -\Delta_c^b(\mathbf{n} - \mathbf{e}_b)$.

Suppose $\mathcal{L}_{c,\alpha}(\mathbf{n}_c^*) > \mathcal{L}_{c,\alpha}(\tilde{\mathbf{n}}_c^{(|\phi_{\mathrm{V},c}|)})$. Then $\exists\ b_+, b_- \in \phi_{\mathrm{BS},c}$ such that $n_c^{b_+,*} > \tilde{n}_c^{b_+,(|\phi_{\mathrm{V},c}|)}$ and $n_c^{b_-,*} < \tilde{n}_c^{b_-,(|\phi_{\mathrm{V},c}|)}$. Let t_0 be the index of the last iteration of the sequential algorithm that associated an additional vehicle to

BS b_-. We seek to show that $\Delta_c^{b_-}(\mathbf{n}_c^*) + \Delta_c^{-b_+}(\mathbf{n}_c^*) \geq 0$. We have successively:

$$\Delta_c^{b_-}(\mathbf{n}_c^*) \geq \Delta_c^{b_-}(\tilde{\mathbf{n}}_c^{(t_0-1)}) \tag{B.17}$$

$$\geq \Delta_c^{b_+}(\tilde{\mathbf{n}}_c^{(t_0-1)}) \tag{B.18}$$

$$\geq \Delta_c^{b_+}(\mathbf{n}_c^* - \mathbf{e}_{b_+}) \tag{B.19}$$

$$= -\Delta_c^{-b_+}(\mathbf{n}_c^*) \tag{B.20}$$

Where inequality B.17 follows from the facts that $n_c^{b_-,*} \leq \tilde{n}_c^{b_-,(t_0-1)}$ and $\Delta_c^{b_-}$ is decreasing. Inequality B.18 is a necessary condition for the sequential algorithm to associate the vehicle to BS b_- at step t_0, and inequality B.19 follows from the facts that $n_c^{b_+,*} \geq 1 + \tilde{n}_c^{b_+,|\phi_{\mathrm{V},c}|} \geq 1 + \tilde{n}_c^{b_+,(t_0-1)}$ and $\Delta_c^{b_+}$ is decreasing in $n_c^{b_+}$. Finally equality B.20 follows from the definition of $\Delta_c^b(\mathbf{n})$. Therefore, we get $\Delta_c^{b_-}(\mathbf{n}_c^*) + \Delta_c^{-b_+}(\mathbf{n}_c^*) \geq 0$, hence $\mathbf{n}_c^*$ is not optimal. We conclude from this contradiction that the sequential algorithm finds an optimal association vector. □

Appendix C

Chapter 4 Proofs

C.1 Timeliness Metric Motivation

We show that the timeliness metric $\frac{1}{2\rho} + d^t + d^c$ is reasonable in the setting under study. In [172], the authors show in Theorem 3 that the mean AoI τ_m of device m is:

$$\tau_m = \frac{\mathbb{E}[I_m D_m] + \mathbb{E}[I_m^2]/2}{\mathbb{E}[I_m]} \tag{C.1}$$

where I_m represents the inter-arrival time between updates originating from m, and D_m is the system delay experienced by its updates. Intuitively, I_m and D_m are correlated, as a long inter-arrival time would be associated with the compute node having more time to process the tasks currently queued. More formally, we have the following result proved in Appendix C.2:

Lemma C.1.1. $\mathbb{E}[I_m D_m] \leq \mathbb{E}[I_m] \cdot \mathbb{E}[D_m]$

Therefore, for deterministic I_m , $\mathbb{E}[I_m] = \frac{1}{\rho}$, $\mathbb{E}[I_m^2] = \frac{1}{\rho^2}$ and $\mathbb{E}[D_m] = d^t + d^c$, we get $\tau_m \leq \frac{1}{2\rho} + d^t + d^c$. Now, with increasing number of devices (our regime of interest), the incremental impact of an individual device m on the delay experienced by its own packets becomes negligible. Hence, by separation

of time scales, one can conclude that this bound becomes tight in the limit, motivating our timeliness metric.

C.2 Proof of Lemma C.1.1:

Proof. We start by proving a useful result:

Lemma C.2.1 (Extension of the FKG inequality [58])**.** *Let f and g be respectively non-increasing and non-decreasing functions, then*

$$\mathbb{E}[f(X)g(X)] \leq \mathbb{E}[f(X)]\mathbb{E}[g(X)]. \tag{C.2}$$

Proof. Let X_1 and X_2 be two independent copies of the same random variable X. We have:

$$(g(X_1) - g(X_2))(f(X_1) - f(X_2)) \leq 0 \tag{C.3}$$

$$\iff \mathbb{E}[f(X_1)g(X_1)] + \mathbb{E}[f(X_2)g(X_2)] \leq \mathbb{E}[f(X_2)g(X_1)] + \mathbb{E}[f(X_1)g(X_2)] \tag{C.4}$$

$$\iff \mathbb{E}[f(X)g(X)] \leq \mathbb{E}[f(X)]\mathbb{E}[g(X)] \tag{C.5}$$

□

Let N_m be the number of arrivals to the queue during the inter-arrival time I_m, and let $\{R_i\}_i$ be the residual times of the update packets in the compute node queue upon arrival of the update from m. By the law of total covariance, we have:

$$\mathrm{Cov}(I_m, D_m) = \mathbb{E}[\mathrm{Cov}(I_m, D_m|\{R_i\}_i, N_m)]+$$

$$\mathrm{Cov}(\mathbb{E}[I_m|\{R_i\}_i, N], \mathbb{E}[D_m|\{R_i\}_i, N_m])$$

By observing that D_m is a deterministic function of $\{R_i\}_i$ and N_m, we conclude that the first term must be 0 as the covariance of two random variables is 0 if one of them is deterministic. Moreover, we note that $\mathbb{E}[I_m|\{R_i\}_i, N_m] = f(\{R_i\}_i, N_m)$ and $\mathbb{E}[D_m|\{R_i\}_i, N_m] = g(\{R_i\}_i, N_m)$, where f and g are deterministic functions. Clearly, f and g are respectively nonincreasing and nondecreasing in $\{R_i\}_i$ and N_m. Hence, from Lemma C.2.1,

$$\mathrm{Cov}(\mathbb{E}[I_m|\{R_i\}_i, N_m], \mathbb{E}[D_m|\{R_i\}_i, N_m]) \leq 0, \tag{C.6}$$

thus $\mathrm{Cov}(I_m, D_m) \leq 0$, hence $\mathbb{E}[I_m D_m] \leq \mathbb{E}[I_m]\mathbb{E}[D_m]$. □

C.3 Proof of Theorem 4.6.1

Proof. We start from Algorithm 4.1's decision policy. We have:

$$\arg\min_{s\in\tilde{\mathcal{S}}_a} \int_{u_s(t)}^{u'_{a,s}(t)} f(u)du \tag{C.7}$$

$$= \arg\min_{s\in\tilde{\mathcal{S}}_a} \int_{u_s(t)}^{u'_{a,s}(t)} \frac{1}{1-u}du \tag{C.8}$$

$$= \arg\max_{s\in\tilde{\mathcal{S}}_a} \log(1-u'_{a,s}(t)) - \log(1-u_s(t)) \tag{C.9}$$

$$= \arg\max_{s\in\tilde{\mathcal{S}}_a} \log(1-u'_{a,s}(t)) - \log(1-u_s(t)) \tag{C.10}$$

$$+ \sum_{s'\in\tilde{\mathcal{S}}_a} \log(1-u_{s'}(t)) \tag{C.11}$$

$$= \arg\max_{s\in\tilde{\mathcal{S}}_a} \log(1-u'_{a,s}(t)) + \sum_{s'\in\tilde{\mathcal{S}}_a\setminus s} \log(1-u_{s'}(t)) \tag{C.12}$$

where the third step consists in adding a constant w.r.t. s. □

C.4 Proof of Theorem 4.6.2

Proof. This lower-bound proof is a generalization of the one proposed in [6]. As in [6], we introduce a virtual overflow compute node s_o of infinite capacity hosting the VMs of devices that have been blocked. Define $\beta_a(t)$ to be the number of Type a customers that have been blocked, i.e., that have been hosted in s_o, in $[0, t)$, T_m to be the random holding time of device m, and $X_{a,s}(t)$ to be the state of the network at time t, i.e., the number of Type a customers served by s. We have successively:

$$\mathcal{C}_D(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) \tag{C.13}$$

$$= \sum_{a \in \mathcal{A}} w_a \mu_a^{-1} \lambda_a P(B_a; \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) \tag{C.14}$$

$$= \sum_{a \in \mathcal{A}} w_a \mu_a^{-1} \lambda_a \lim_{t \to \infty} \frac{\mathbb{E}[\beta_a(t)]}{t \lambda_a} \tag{C.15}$$

$$\overset{(a)}{=} \lim_{t \to \infty} \sum_{a \in \mathcal{A}} \frac{w_a \mu_a^{-1}}{t} \mathbb{E}[\sum_{m=1}^{\beta_a(t)} \frac{T_m}{\mu_a^{-1}}] \tag{C.16}$$

$$\overset{(b)}{\geq} \lim_{t \to \infty} \sum_{a \in \mathcal{A}} \frac{w_a}{t} \mathbb{E}[\int_0^t X_{a,s_o}(y) \, \mathrm{d}y] \tag{C.17}$$

$$= \lim_{t \to \infty} \sum_{a \in \mathcal{A}} \frac{w_a}{t} (\mathbb{E}[\int_0^t \sum_{s \in \mathcal{S} \cup s_o} X_{a,s}(y) \, \mathrm{d}y] \tag{C.18}$$

$$- \mathbb{E}[\int_0^t \sum_{s \in \mathcal{S}} w_a X_{a,s}(y) \, \mathrm{d}y]) \tag{C.19}$$

$$\overset{(c)}{=} \lim_{t \to \infty} \sum_{a \in \mathcal{A}} \frac{w_a \lambda_a}{\mu_a t} \int_0^t 1 - e^{-y\mu_a} \, \mathrm{d}y \tag{C.20}$$

$$- \lim_{t \to \infty} \frac{1}{t} \int_0^t \mathbb{E}[\sum_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}} w_a X_{a,s}(y)] \, \mathrm{d}y \tag{C.21}$$

$$\overset{\text{(d)}}{\geq} \sum_{a\in\mathcal{A}} \frac{w_a\lambda_a}{\mu_a} \lim_{t\to\infty} \frac{t + e^{-t\mu_a} - 1}{t} \tag{C.22}$$

$$- (\sum_{a\in\mathcal{A}} \sum_{s\in\mathcal{S}} w_a A^*_{a,s}) \lim_{t\to\infty} \frac{1}{t} \int_0^t 1 \,\mathrm{d}y) \tag{C.23}$$

$$= \sum_{a\in\mathcal{A}} w_a(\lambda_a\mu_a^{-1} - \sum_{s\in\mathcal{S}} A^*_{a,s}) \tag{C.24}$$

$$= \underline{\mathcal{C}}_D(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa}) \tag{C.25}$$

Step (a) follows from the algebraic limit theorem as the number of types is finite and from the fact that T_m are i.i.d. of mean μ_a^{-1}, i.e., $\frac{T_m}{\mu_a^{-1}}$ have unit mean. Step (b) is a bound because some customers that arrived to s_o before time t may still be in the system at time t. Step (c) follows from the idea that the augmented network can be viewed as an $M/M/\infty$ queue, using the expression of the mean number of users in such a system at time t starting from the empty state at $t = 0$, as well as the Fubini-Tonelli theorem. In step (d), we use the fact that at every time y, $\mathbb{E}[\sum_{a\in\mathcal{A}} \sum_{s\in\mathcal{S}} w_a X_{a,s}(y)] \leq \sum_{a\in\mathcal{A}} \sum_{s\in\mathcal{S}} w_a A^*_{a,s}$ by definition of LP-MKP$(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \Delta, \boldsymbol{\kappa})$. □

Appendix D

Chapter 5 Proofs

D.1 Proof Theorem 5.5.7

Proof. In this proof, we restrict our attention to a region $\mathcal{R}(\mathbf{x}, a_0)$, where $\mathcal{R}(\mathbf{x}, a_0) = \{\mathbf{y} \in \mathbb{R}^2 : \|\mathbf{y}\|_2 \leq \eta + \sigma(\mathbf{x})\}$, and η is such that $d^t(\eta) = a_0$. Defining $\mathcal{R}(\mathbf{x}, a_0)$ in this way leads to the following observation: $\mathbf{g} \notin \mathcal{R}(\mathbf{x}, a_0) \implies q(\mathbf{x}, \mathbf{g}) < 1 - \epsilon, \forall \mathbf{x} \in \mathbb{R}^{n\times 2}, \forall \epsilon \in [0, 1]$. Therefore, $\mathcal{F}_\epsilon(\mathbf{x}) \subset \mathcal{R}(\mathbf{x}, a_0)$, allowing us to study it by only considering points in $\mathcal{R}(\mathbf{x}, a_0)$. We now prove a useful lemma.

Lemma D.1.1 (Stochastic Majorization of Max Distance)**.** *Let* $\mathbf{x}$ *and* $\mathbf{x}' \in \mathbb{R}^{n\times 2}$ *be any two configurations of* n *players, where* $\sigma(\mathbf{x}) \geq \sigma(\mathbf{x}')$*, and let* $\mathbf{G}$ *be a random G-server coordinate vector uniformly distributed on* $\mathcal{R}(\mathbf{x}, a_0)$*. Define* $\mathbf{\Delta}, \mathbf{\Delta}' \in \mathbb{R}^n_+$ *to be the random vectors of induced distances between* $\mathbf{G}$ *and each point in* $\mathbf{x}$ *and* $\mathbf{x}'$*, respectively.*

If $\max_i \Delta_i \leq^{st} \max_i \Delta'_i$*, then under the JMRA algorithm*

$$|\mathcal{F}_\epsilon(\mathbf{x})| \geq |\mathcal{F}_\epsilon(\mathbf{x}')|, \forall \epsilon \in [0, 1]. \tag{D.1}$$

Proof. We start this proof by noting that

$$|\mathcal{F}_\epsilon(\mathbf{x})| = \iint_{\mathcal{R}(a_0,\mathbf{x})} \mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{g})) > 1 - \epsilon\} d\mathbf{g} \tag{D.2}$$

$$= |\mathcal{R}(a_0, \mathbf{x})| \cdot \mathbb{E}_{\mathbf{G}}[\mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{G})) > 1 - \epsilon\}] \tag{D.3}$$

Similarly, $|\mathcal{F}_\epsilon(\mathbf{x}')| = \mathbb{E}_{\mathbf{G}}[\mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}', \mathbf{G})) > 1 - \epsilon\}]$. Hence $|\mathcal{F}_\epsilon(\mathbf{x})| \geq |\mathcal{F}_\epsilon(\mathbf{x}')|, \forall \epsilon \in [0,1] \iff \mathbb{E}_{\mathbf{G}}[\mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{G})) > 1 - \epsilon\}] \geq \mathbb{E}_{\mathbf{G}}[\mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{G})) > 1 - \epsilon\}], \forall \epsilon \in [0,1]$.

Furthermore, we note that $\mathbb{1}\{q(\mathbf{d}^t) > 1 - \epsilon\} = \mathbb{1}\{\max_{\boldsymbol{\rho}} \left\{ \mathbb{P}(A_{\mathbf{D}^t_{\boldsymbol{\delta}}, \boldsymbol{\rho}} \leq a_0 \mid \mathbf{D}^t_{\boldsymbol{\delta}} = \mathbf{d}^t) \ : \ d^c(\sum_j \rho_j) \leq \tau \right\} > 1 - \epsilon\}$ is a symmetric function of the delay vector $\mathbf{d}^t$, see Equation 5.5, and decreasing in each of the components of this random vector. Besides, the indicator function returns non-negative values, less than or equal to 1. Therefore, $\mathbb{1}\{q(\mathbf{D}^t_{\boldsymbol{\Delta}}) > 1 - \epsilon\}$ is a symmetric joint survival function of the random delay vector $\mathbf{D}^t_{\boldsymbol{\Delta}}$, hence of the random distance vector $\boldsymbol{\Delta}$. Now we have:

$$\max_i \Delta_i \leq^{\text{st}} \max_i \Delta'_i \tag{D.4}$$

$$\iff \mathbb{P}(\max_i \Delta_i \leq t) \geq \mathbb{P}(\max_i \Delta'_i \leq t), \forall t \in \mathbb{R} \tag{D.5}$$

$$\iff \mathbb{P}(\Delta_1 \leq t, \cdots, \Delta_n \leq t) \geq \mathbb{P}(\Delta'_1 \leq t, \cdots, \Delta'_n \leq t), \forall t \in \mathbb{R} \tag{D.6}$$

$$\iff \max_i \Delta_i \leq^{\text{slo}} \max_i \Delta'_i \tag{D.7}$$

$$\iff \mathbb{E}[\psi(\boldsymbol{\Delta})] \geq \mathbb{E}[\psi(\boldsymbol{\Delta}')], \forall \psi \in \mathcal{C}. \tag{D.8}$$

where $\mathcal{C}$ is the class of symmetric joint survival functions. The definition of the *symmetric lower orthant* ordering and its properties can be found in [141, 142]. The result follows from the fact that $\mathbb{1}\{q(\bar{\mathbf{d}}^t(\mathbf{x}, \mathbf{G})) > 1 - \epsilon\} \in \mathcal{C}, \forall \epsilon \in [0,1]$. $\square$

We now proceed to prove the theorem. The proof is subdivided in two parts: we first show that for any player configuration $\mathbf{x}$ in a disk of radius $\sigma(\mathbf{x})$

moving the players to the boundary of the disk reduces $|\mathcal{F}_\epsilon(\mathbf{x})|$; we then show that equispacing the players on the boundary of the disk minimizes this area.

Part 1: Equalizing the radial coordinate components. In this part, we construct a coupling between any configuration of players $\mathbf{x}$, of geographical spread $\sigma(\mathbf{x})$ and the configuration $\mathbf{x}'$ of players having the same polar angular coordinates, as in $\mathbf{x}$, but all the polar radial coordinate components equal to $\sigma(\mathbf{x})$, i.e., all the players are located on the boundary of the circle centered at the origin and of radius $\sigma(\mathbf{x})$. We observe that under configuration $\mathbf{x}'$ region $\mathcal{R}(\mathbf{x}, a_0)$ can be partitioned into n sectors, where sector $\mathcal{R}'_k(\mathbf{x}, a_0)$ is defined to be the region of points such that player k is the furthest player, or equivalently, $\mathcal{R}'_k(\mathbf{x}, a_0) = \{\mathbf{g} \in \mathcal{R}(\mathbf{x}, a_0) : \arg\max_i \Delta'_i = k\}$. Similarly, we define $\mathcal{R}_k(\mathbf{x}, a_0) = \{\mathbf{g} \in \mathcal{R}(\mathbf{x}, a_0) : \arg\max_i \Delta_i = k\}$. Since no adjacent players are separated by an angle larger than π, by construction of the circle of radius $\sigma(\mathbf{x})$ to be the circle of smallest radius encompassing all the players, it is clear that $\langle \mathbf{x}_k, \mathbf{g}\rangle \leq 0, \forall \mathbf{g} \in \mathcal{R}_k(\mathbf{x}, a_0), \forall k$ and $\langle \mathbf{x}'_k, \mathbf{g}\rangle \leq 0, \forall \mathbf{g} \in \mathcal{R}'_k(\mathbf{x}, a_0), \forall k$. Now we have: $\max_i \delta_i = \delta_k = \|\mathbf{x}_k - \mathbf{g}\|_2 = \sqrt{\|\mathbf{x}_k\|_2^2 + \|\mathbf{g}\|_2^2 - 2\langle \mathbf{x}_k, \mathbf{g}\rangle} \leq \sqrt{\|\mathbf{x}'_j\|_2^2 + \|\mathbf{g}\|_2^2 - 2\langle \mathbf{x}'_j, \mathbf{g}\rangle} = \delta'_j = \max_i \delta'_i$, where the inequality follows from the facts that $\|\mathbf{x}_k\|_2 \leq \|\mathbf{x}'_j\|_2$, $\langle \mathbf{x}_k, \mathbf{g}\rangle \leq 0$, $\langle \mathbf{x}'_j, \mathbf{g}\rangle \leq 0$, and the angle between $\mathbf{x}_k$ and $\mathbf{g}$ being equal to the one between $\mathbf{x}'_j$ and $\mathbf{g}$, by construction. Therefore, for any realization $\mathbf{g} \in \mathcal{R}(\mathbf{x}, a_0)$ we have $\max_i \delta_i \leq \max_i \delta'_i$, thus $\max_i \Delta_i \leq \max_i \Delta'_i$, almost surely. It follows that $\max_i \Delta_i \leq^{\text{st}} \max_i \Delta'_i$, hence we get from Lemma D.1.1, $|\mathcal{F}_\epsilon(\mathbf{x})| \geq |\mathcal{F}_\epsilon(\mathbf{x}')|, \forall \epsilon \in [0, 1]$.

Part 2: Equalizing the angular coordinate components. In this

part, we prove that for any configuration $\mathbf{x}$ such that all the players are on the boundary of a circle of radius $\sigma(\mathbf{x})$, spacing the players regularly on the boundary minimizes $|\mathcal{F}_\epsilon(\mathbf{x})|$. In this setting, the player configuration can be parametrized by $\boldsymbol{\theta}$ the vector of differential angles between adjacent players on the circle.

We start by deriving an expression for $\mathbb{P}(\max_i \Delta_i \geq t | \|\mathbf{G}\|_2, \boldsymbol{\theta})$, the conditional c.d.f. of $\max_i \Delta_i$, given $\|\mathbf{G}\|_2 = r$ and parametrized by $\boldsymbol{\theta} \in [0, 2\pi]^n$, where $\sum_i \theta_i = 2\pi$. One can show that:

$$\mathbb{P}(\max_i \Delta_i \geq t | \|\mathbf{G}\|_2 = r, \boldsymbol{\theta}) = \frac{\sum_k \min[\gamma(t, r), \theta_k/2]}{\pi} \tag{D.9}$$

where $\gamma(t, r) = \pi - \cos^{-1}(\max[\min[\frac{\sigma(\mathbf{x})^2 + r^2 - t^2}{2r\sigma(\mathbf{x})}, 1], -1])$.

We observe that $\mathbb{P}(\max_i \Delta_i \geq t | \|\mathbf{G}\|_2 = r, \boldsymbol{\theta})$ is symmetric and concave in $\boldsymbol{\theta}$, it is therefore Schur-concave in $\boldsymbol{\theta}$. Let $\boldsymbol{\theta}'$ parametrize the equispaced configuration, i.e., $\theta'_i = \frac{2\pi}{n}, \forall i$, then clearly $\boldsymbol{\theta}' \prec \boldsymbol{\theta}, \forall \boldsymbol{\theta} \in [0, 2\pi]^n$, where $\sum_i \theta_i = 2\pi$. We say that $\boldsymbol{\theta}'$ is *majorized* by $\boldsymbol{\theta}$. Therefore, from Schur-concavity, we have

$$\mathbb{P}(\max_i \Delta'_i \geq t | \|\mathbf{G}\|_2 = r, \boldsymbol{\theta}') \geq \mathbb{P}(\max_i \Delta_i \geq t | \|\mathbf{G}\|_2 = r, \boldsymbol{\theta}), \forall r, \tag{D.10}$$

which implies that

$$\mathbb{P}(\max_i \Delta'_i \geq t) \geq \mathbb{P}(\max_i \Delta_i \geq t), \forall \boldsymbol{\theta} \in [0, 2\pi]^n, \tag{D.11}$$

by integrating over all values of r so as to span $\mathcal{R}(\mathbf{x}, a_0)$. It follows that $\max_i \Delta_i \leq^{\text{st}} \max_i \Delta'_i$, hence we get from Lemma D.1.1,

$$|\mathcal{F}_\epsilon(\mathbf{x})| \geq |\mathcal{F}_\epsilon(\mathbf{x}')|, \forall \epsilon \in [0, 1]. \tag{D.12}$$

□

D.2 Proof of Theorem 5.6.3

Proof. We know that $\bar{q}(\boldsymbol{\delta}) = \mathbb{P}(q(\mathbf{D}^t_{\boldsymbol{\delta}}) > \epsilon)$ is a Schur-concave function in $\boldsymbol{\delta}$ as $q(\mathbf{d}^t)$ is Schur-concave in $\mathbf{d}^t$, see [112]. The Schur-concavity property of $q(\mathbf{d}^t)$ directly follows from the fact that the function is symmetric in the entries of $\mathbf{d}^t$, and concave in $\mathbf{d}^t$, see section 3.2.5 in [24]. In addition, from Assumption 5.5.1, we know that $\forall \mathbf{z} \in \mathbb{R}^n_+,\ \mathbf{D}^t_{\boldsymbol{\delta}} \leq^{\text{icx}} \mathbf{D}^t_{\boldsymbol{\delta}+\mathbf{z}}$. Since $q(\mathbf{d}^t)$ is decreasing and concave in $\mathbf{d}^t$, we have

$$\bar{q}(\boldsymbol{\delta}) = \mathbb{P}(q(\mathbf{D}^t_{\boldsymbol{\delta}}) > 1-\epsilon) \geq \mathbb{P}(q(\mathbf{D}^t_{\boldsymbol{\delta}+\mathbf{z}}) > 1-\epsilon) = \bar{q}(\boldsymbol{\delta}+\mathbf{z}), \tag{D.13}$$

i.e., $\bar{q}(\boldsymbol{\delta})$ is decreasing in $\boldsymbol{\delta}$. Therefore, $\bar{q}(\boldsymbol{\delta})$ is a Schur-concave decreasing function in $\boldsymbol{\delta}$, thus given $\boldsymbol{\delta}$ and $\boldsymbol{\delta}' \in \mathbb{R}^n_+$ be two distance vectors induced by two feasible game servers, $\boldsymbol{\delta} \prec_w \boldsymbol{\delta}' \implies \bar{q}(\boldsymbol{\delta}) \geq \bar{q}(\boldsymbol{\delta}')$, as argued in [112]. □

Appendix E

Chapter 6 Supplementary Material and Proofs

E.1 Petovello's Method Overview

Consider the following general linear dynamic system:

$$\begin{cases} \mathbf{x}_{k+1} = \Phi \mathbf{x}_k + \Gamma \mathbf{w}_k \\ \mathbf{z}_k = H\mathbf{x}_k + \mathbf{v}_k \\ \mathbf{v}_{k+1} = \Psi \mathbf{v}_k + \boldsymbol{\zeta}_k, \end{cases} \tag{E.1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the true state at time k, and $\mathbf{z}_k \in \mathbb{R}^m$ is the vector of m measurements at time k, such that $\mathbf{w}_k \sim \mathcal{N}(0, Q)$, $\boldsymbol{\zeta}_k \sim \mathcal{N}(0, R)$, and $\mathbb{E}[\mathbf{w}_k \boldsymbol{\zeta}_l^\top] = \mathbf{0}_{n \times m}$, for all k, l.

Similarly to the classical Kalman filter, the filtering procedure at any time k can be decomposed into a prediction and measurement update steps. In Petovello's method [126], the prediction step is performed at time k as

$$\hat{\mathbf{x}}_{k|k-1} = \Phi \hat{\mathbf{x}}_{k-1|k-1} \tag{E.2}$$

$$P_{k|k-1} = \Phi P_{k-1|k-1} \Phi^\top + Q \tag{E.3}$$

while the measurement update step is performed at time k as

$$\tilde{\mathbf{z}}_k = \mathbf{z}_k - \Phi \mathbf{z}_{k-1} \tag{E.4}$$

$$\tilde{H} = H - \Psi H \Phi^{-1} \tag{E.5}$$

$$S = Q\Gamma^\top(\Phi^{-1})^\top H^\top \Psi^\top \tag{E.6}$$

$$\tilde{R} = \Psi H \Phi^{-1}\Gamma Q \Gamma^\top (\Phi^{-1})^\top H^\top \Psi^\top + R \tag{E.7}$$

$$K_k = [P_{k|k-1}\tilde{H}^\top + S][\tilde{H}P_{k|k-1}\tilde{H}^\top + \tilde{R} + \tilde{H}S + S^\top \tilde{H}^\top]^{-1} \tag{E.8}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k[\tilde{\mathbf{z}}_k - \tilde{H}\hat{\mathbf{x}}_{k|k-1}] \tag{E.9}$$

$$P_{k|k} = P_{k|k-1} - K_k[\tilde{H}P_{k|k-1}\tilde{H}^\top + \tilde{R} + \tilde{H}S + S^\top \tilde{H}^\top]K_k^\top \tag{E.10}$$

E.2 Proof Theorem 6.5.1

Proof. To study the steady-state error variance of the DKF tracking the environment characterized by Equation 6.4, we invoke a result in [156] showcasing the equivalence between centralized and decentralized Kalman Filtering when the MF performs Track-to-Track Fusion with Memory. We can therefore study the steady-state error variance of the associated Centralized Kalman Filter, whose equations are provided in Appendix E.1 with the following simplifications: $\Psi = \text{diag}(\boldsymbol{\alpha})$, $R = \text{diag}(\boldsymbol{\sigma^2})$, $\Phi = 1$, $\Gamma = 1$, $H = \mathbf{1}_{|\mathcal{C}|\times 1}$.

Lemma E.2.1. *Let* $W \in \mathbb{R}_+^*, X \in \mathbb{R}_+^{*,1\times n}, Y \in \mathbb{R}_+^{*,n\times 1}, Z \in \mathbb{R}_+^{*,n\times n}$ *and* Z *invertible. We have:*

$$W = X[YX + Z]^{-1}X^\top \iff W = XZ^{-1}[X^\top - YW] \tag{E.11}$$

Proof.

$$W = X[YX + B]^{-1}X^\top \tag{E.12}$$

$$\iff YW = YX[YX + Z]^{-1}X^\top + Z[YX + Z]^{-1}X^\top - Z[YX + Z]^{-1}X^\top \tag{E.13}$$

$$\iff YW + Z[YX + Z]^{-1}X^\top = X^\top \tag{E.14}$$

$$\iff X^\top = [YX + Z]Z^{-1}[X^\top - YW] \tag{E.15}$$

$$\iff X^\top = YXZ^{-1}[X^\top - YW] + X^\top - YW \tag{E.16}$$

$$\iff Y^\top YW = Y^\top YXZ^{-1}[X^\top - YW] \tag{E.17}$$

$$\iff W = XZ^{-1}[X^\top - YW] \tag{E.18}$$

□

From Equations E.3, E.10 and E.8, we know that P^* must satisfy the following fixed point equation:

$$\begin{aligned} P^* = P^* + Q - [P^*\tilde{H}^\top + Q\tilde{H}^\top + S]\times \\ [\tilde{H}P^*\tilde{H}^\top + \tilde{H}Q\tilde{H}^\top + \tilde{R} + \tilde{H}S + S^\top\tilde{H}^\top]^{-1}[P^*\tilde{H}^\top + Q\tilde{H}^\top + S]^\top \end{aligned} \tag{E.19}$$

$$\begin{aligned} \iff Q = [P^*\tilde{H}^\top + QH^\top]\times \\ [\tilde{H}(P^*\tilde{H}^\top + QH^\top) + \tilde{R} + S^\top\tilde{H}^\top]^{-1}[P^*\tilde{H}^\top + QH^\top]^\top \end{aligned} \tag{E.20}$$

$$\iff Q = [P^*(\mathbf{1} - \boldsymbol{\alpha}) + Q\mathbf{1}]^\top[Q\boldsymbol{\alpha}\mathbf{1}^\top + R]^{-1}[P^*(\mathbf{1} - \boldsymbol{\alpha}) + Q\boldsymbol{\alpha}] \tag{E.21}$$

where the last step directly follows from Lemma E.2.1. Also, we have from the matrix inversion lemma:

$$[Q\boldsymbol{\alpha}\mathbf{1}^\top + R]^{-1} = R^{-1} - R^{-1}Q\boldsymbol{\alpha}(1 + Q\sum_i \frac{\alpha_i}{\sigma_i^2})^{-1}\mathbf{1}^\top R^{-1}. \tag{E.22}$$

It follows that for A, B, C, D, E and F defined in Theorem 6.5.1:

$$Q = [P^*(\mathbf{1} - \boldsymbol{\alpha}) + Q\mathbf{1}]^\top R^{-1}[P^*(\mathbf{1} - \boldsymbol{\alpha}) + Q\boldsymbol{\alpha}]$$

$$- \frac{Q}{1+QA}[P^*(\mathbf{1}-\boldsymbol{\alpha}) + Q\mathbf{1}]^\top R^{-1}\boldsymbol{\alpha}\mathbf{1}^\top R^{-1}[P^*(\mathbf{1}-\boldsymbol{\alpha}) + Q\boldsymbol{\alpha}] \quad \text{(E.23)}$$

Now given that $R = \text{diag}(\boldsymbol{\sigma^2})$, expanding the above expression combining and simplifying the terms in $(P^*)^2$, P^* and independent ones lead to the following quadratic equation:

$$\begin{aligned}
&(B - \frac{QCD}{1+QA}) \cdot (P^*)^2 + (\frac{QE}{1+QA}) \cdot P^* - (Q^2A - \frac{Q^3A^2}{1+QA} - Q) = 0 \\
&\iff (B + Q(BA - CD)) \cdot (P^*)^2 + QE \cdot P^* - Q = 0 \qquad \text{(E.24)}
\end{aligned}$$

Now given that $P^* \geq 0$, the unique solution of this quadratic equation is

$$P^* = \frac{-QE + \sqrt{Q^2FB + 4QB}}{2(B + Q(BA - CD))}. \qquad \text{(E.25)}$$

□

E.3 Proof Theorem 6.5.5

Proof. Consider the feedback rate selection policy that triggers a reset signal to vehicle i whenever the estimation error variance reaches β_i. This policy is clearly optimal as it minimizes number of feedback signals transmitted to any LF i, while ensuring that the error constraint is slack. We now show that this policy sends the same number of feedback signals as the one described in Theorem 6.5.5.

Let t be one of those trigger times, in general t can be expressed as $t = k\tau + \delta$ for some $k \in \mathbb{N}$ and $0 \leq \delta < \tau$. Then the estimation error variance of LF i would drop at time t to $P^* + \nu^2\delta$ and would evolve independently

from there until it reaches β_i again. Now consider that the feedback signal is transmitted at time $k\tau$ instead of time t. Then the estimation error variance of LF i successively drops to P^* at time $k\tau$, rises to $P^* + \nu^2\delta$ at time t and evolves independently from there until it reaches β_i again at the same time as it would have without the time shift. Therefore, shifting the feedback time to the immediate preceding multiple of τ has no effect on the future trigger times. Thus, all the feedback times can be shifted without affecting the total number of transmitted feedback signals. □

In retrospect, this is not surprising as the information contained in the feedback signal at time $k\tau$ and $t = k\tau + \delta$ is the same.

E.4 Proof Theorem 6.5.6

Proof. To prove this theorem, we prove first two useful lemmas.

Lemma E.4.1. *The function $T_i(x)$ is increasing in x, for any i in $\mathcal{V}$.*

Proof. We have $T_i(x) = \frac{x(\sigma_i^2+Q\alpha_i^2)+Q\sigma_i^2}{x(a-\alpha_i)^2+\sigma_i^2+Q}$, and we verify that its derivative is $\frac{\partial T_i(x)}{\partial x} = \frac{(\sigma_i^2+Q\alpha_i)^2}{(x(1-\alpha_i)^2+\sigma_i^2+Q)^2} > 0, \forall x.$ □

Lemma E.4.2. *$\sum_{i\in\mathcal{R}^*} \rho_i^*$ is a non-decreasing function of P^*.*

Proof. Let $x_1, x_2 \in \mathbb{R}+$, such that $x_1 < x_2$. From Lemma E.4.1, we deduce that $T_k(x_1) < T_k(x_2)$ and hence

$$T_i^{(n)}(x_1) < T_i^{(n)}(x_2), \forall n \in \mathbb{N} \tag{E.26}$$

Now let $\gamma_1^* = \arg\max_\gamma\{\gamma : T_i^{(\gamma-1)}(x_1) + Q \leq \beta_i\}$ and $\gamma_2^* = \arg\max_\gamma\{\gamma : T_i^{(\gamma-1)}(x_2) + Q \leq \beta_i\}$. Then it must be that $\gamma_1^* \geq \gamma_2^*$. Indeed, suppose $\gamma_1^* < \gamma_2^*$, then from Equation E.26, $T_i^{(\gamma_2^*-1)}(x_1) + Q < T_i^{(\gamma_2^*-1)}(x_2) + Q \leq \beta_i$ which, given the definition of γ_1^*, contradicts the premise. It follows from Equation 6.11 that $\rho_i^*(x_1) = (\tau\gamma_1^*)^{-1} \leq (\tau\gamma_2^*)^{-1} = \rho_i^*(x_2)$. It follows that $\sum_{i\in\mathcal{R}^*}\rho_i^*$ is non-decreasing in P^* as a sum of non-decreasing functions is non-decreasing. $\square$

Let $\mathcal{C}_{m,1}^* = \arg\min_{\mathcal{C}}\{P^*(\mathcal{C}) : |\mathcal{C}| = m\}$ and
$\mathcal{C}_{m,2}^* = \arg\min_{\mathcal{C}}\left\{\sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}) : |\mathcal{C}| = m\right\}$.

First, we know that $\sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,2}^*) \leq \sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,1}^*)$ from the definition of $\mathcal{C}_{m,2}^*$. Now assume that $\sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,2}^*) < \sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,1}^*)$, then it must be that $P^*(\mathcal{C}_{m,2}^*) < P^*(\mathcal{C}_{m,1}^*)$ from Lemma E.4.2, which contradicts the definition of $\mathcal{C}_{m,1}^*$. Therefore, it must be that $\sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,2}^*) = \sum_{i\in\mathcal{R}^*}\rho_i^*(\mathcal{C}_{m,1}^*)$. $\square$

Bibliography

[1] 5GPPP. 5G Automotive Vision. 2015.

[2] 5GPPP. Cloud-Native and Verticals' services. 2019.

[3] K. Abboud, H. Omar, and W. Zhuang. Interworking of DSRC and Cellular Network Technologies for V2X Communications: A Survey. *IEEE Transactions on Vehicular Technology*, 2016.

[4] A. Abdrabou and W. Zhuang. Probabilistic Delay Control and Road Side Unit Placement for Vehicular Ad Hoc Networks with Disrupted Connectivity. *IEEE Journal on Selected Areas in Communications*, 2011.

[5] J. Abou Rahal, G. de Veciana, T. Shimizu, and H. Lu. Optimizing Timely Coverage in Communication Constrained Collaborative Sensing Systems. *18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, 2020.

[6] M. Alanyali and B. Hajek. Analysis of simple algorithms for dynamic load balancing. *Mathematics of Operations Research*, 1997.

[7] NGMN Alliance. 5G white paper. *Next generation mobile networks, white paper*, 2015.

[8] Amazon. Amazon EC2 Pricing. *[Online] www.aws.amazon.com, Accessed on 2019-10-05.*, 2019.

[9] Amazon. Amazon Luna. *[Online] https://www.amazon.com/luna/landing-page, Accessed on 2020-11-15.*, 2020.

[10] J. Andrews, T. Bai, M. Kulkarni, A. Alkhateeb, A. Gupta, and R. Heath. Modeling and Analyzing Millimeter Wave Cellular Systems. *IEEE Transactions on Communications*, 2017.

[11] J. Andrews, A. Gupta, and H. Dhillon. A primer on cellular network analysis using stochastic geometry. *arXiv preprint arXiv:1604.03183*, 2016.

[12] J. Andrews, X. Zhang, G. Durgin, and A. Gupta. Are we approaching the fundamental limits of wireless network densification? *IEEE Communications Magazine*, 2016.

[13] R. Atallah, M. Khabbaz, and C. Assi. Multihop V2I Communications: A Feasibility Study, Modeling, and Performance Analysis. *IEEE Transactions on Vehicular Technology*, 2017.

[14] F. Baccelli and B. Błaszczyszyn. *Stochastic geometry and wireless networks*, volume 1. Now Publishers Inc, 2009.

[15] P. Bahl, R. Chandra, P. Lee, V. Misra, J. Padhye, D. Rubenstein, and Y. Yu. Opportunistic Use of Client Repeaters to Improve Performance of WLANs. *IEEE/ACM Transactions on Networking*, 2009.

[16] Y. Bejerano and S. Han. Cell Breathing Techniques for Load Balancing in Wireless LANs. *IEEE Transactions on Mobile Computing*, 2009.

[17] S. Berezner, A. Krzesinski, and P. Taylor. On the inverse of Erlang's function. *Journal of Applied Probability*, 1998.

[18] D. Bertsekas, R. Gallager, and P. Humblet. *Data networks*. Prentice-hall Englewood Cliffs, NJ, 1987.

[19] H. Beytur and E. Uysal-Bıyıkoglu. Minimizing age of information on multi-flow networks. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 2018.

[20] S. Bhoi and P. Khilar. Vehicular communication: a survey. *Institution of Engineering and Technology Networks*, September 2014.

[21] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. Sukhavasi, C. Patel, and S. Geirhofer. Network densification: the dominant theme for wireless evolution into 5G. *IEEE Communications Magazine*, 2014.

[22] F. Boccardi, R. Heath, A. Lozano, T. Marzetta, and P. Popovski. Five disruptive technology directions for 5G. *IEEE Communications Magazine*, 2014.

[23] B. Boudreau. Global Bandwidth & IP Pricing Trends. *[Online] www.telegeography.com, Accessed on 2019-10-13.*, 2017.

[24] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[25] P. Caballero, A. Banchs, G. de Veciana, and X. Costa-Pérez. Network Slicing Games: Enabling Customization in Multi-Tenant Mobile Networks. *IEEE/ACM Transactions on Networking*, 2019.

[26] W. Cai, R. Shea, C. Huang, K. Chen, J. Liu, V. Leung, and C. Hsu. A Survey on Cloud Gaming: Future of Computer Games. *IEEE Access*, 2016.

[27] A. Cailean, B. Cagneau, L. Chassagne, V. Popa, and M. Dimian. A survey on the usage of DSRC and VLC in communication-based vehicle safety applications. In *2014 IEEE 21st Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, 2014.

[28] A. Cailean and M. Dimian. Current Challenges for Visible Light Communications Usage in Vehicle Applications: A Survey. *IEEE Communications Surveys Tutorials*, May 2017.

[29] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri. Distributed Kalman filtering based on consensus strategies. *IEEE Journal on Selected Areas in Communications*, 2008.

[30] F. Castro, A. Martins, N. Capela, and S. Sargento. Multihoming for uplink communications in vehicular networks. In *Wireless Days*, 2017.

[31] J. Chen, G. Mao, C. Li, A. Zafar, and A. Zomaya. Throughput of Infrastructure-Based Cooperative Vehicular Networks. *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[32] S. Chen, J. Hu, Y. Shi, and L. Zhao. LTE-V: A TD-LTE-Based V2X Solution for Future Vehicular Network. *IEEE Internet of Things Journal*, December 2016.

[33] Y. Chen, J. Liu, and Y. Cui. Inter-player Delay Optimization in Multiplayer Cloud Gaming. In *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, 2016.

[34] V. Chetlur and H. Dhillon. Coverage Analysis of a Vehicular Network Modeled as Cox Process Driven by Poisson Line Process. *arXiv preprint:1709.08577*, 2017.

[35] M. Chiang and T. Zhang. Fog and IoT: An Overview of Research Opportunities. *IEEE Internet of Things Journal*, 2016.

[36] J. Cho and Z. Haas. On the throughput enhancement of the downstream channel in cellular radio networks through multihop relaying. *IEEE Journal on Selected Areas in Communications*, 2004.

[37] C. Choi and F. Baccelli. Poisson Cox Point Processes for Vehicular Networks. *IEEE Transactions on Vehicular Technology*, 2018.

[38] J. Choi, V. Va, N. Gonzalez-Prelcic, R. Daniels, C. Bhat, and R. Heath. Millimeter-Wave Vehicular Communication to Support Massive Automotive Sensing. *IEEE Communications Magazine*, December 2016.

[39] S. Chuah, C. Yuen, and N. Cheung. Cloud gaming: a green solution to massive multiplayer online games. *IEEE Wireless Communications*, 2014.

[40] M. Claypool and K. Claypool. Latency and Player Actions in Online Games. *Communications of the ACM*, 2006.

[41] M. Costa, M. Codreanu, and A. Ephremides. Age of information with packet management. In *2014 IEEE International Symposium on Information Theory*, 2014.

[42] S. Das, S. Sen, and R. Jayaram. A dynamic load balancing strategy for channel assignment using selective borrowing in cellular mobile environment. *Wireless Networks*, 1997.

[43] S. Das, H. Viswanathan, and G. Rittenhouse. Dynamic load balancing through coordinated scheduling in packet data systems. In *IEEE INFOCOM 2003*, 2003.

[44] O. Delalleau, E. Contal, E. Thibodeau-Laufer, R. Ferrari, Y. Bengio, and F. Zhang. Beyond Skill Rating: Advanced Matchmaking in Ghost Recon Online. *IEEE Transactions on Computational Intelligence and AI in Games*, 2012.

[45] L. Deng, Y. He, Y. Zhang, M. Chen, Z. Li, J. Lee, Y. Zhang, and L. Song. Device-to-Device Load Balancing for Cellular Networks. *IEEE Transactions on Communications*, 2019.

[46] R. Deng, R. Lu, C. Lai, T. Luan, and H. Liang. Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption. *IEEE Internet of Things Journal*, 2016.

[47] Y. Deng, Y. Li, R. Seet, X. Tang, and W. Cai. The Server Allocation Problem for Session-Based Multiplayer Cloud Gaming. *IEEE Transactions on Multimedia*, 2018.

[48] K. Dey, A. Rayamajhi, M. Chowdhury, P. Bhavsar, and J. Martin. Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network–Performance evaluation. *Transportation Research Part C: Emerging Technologies*, 2016.

[49] E. Dhib, K. Boussetta, N. Zangar, and N. Tabbane. Modeling Cloud gaming experience for Massively Multiplayer Online Games. In *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2016.

[50] A. D'Costa and A. Sayeed. Collaborative signal processing for distributed classification in sensor networks. *Information Processing in Sensor Networks*, 2003.

[51] M. Elbamby, C. Perfecto, M. Bennis, and K. Doppler. Toward Low-Latency and Ultra-Reliable Virtual Reality. *IEEE Network*, 2018.

[52] A. ElGamal, J. Mammen, B. Prabhakar, and D. Shah. Throughput-delay trade-off in wireless networks. In *IEEE INFOCOM 2004*, 2004.

[53] ETSI. 5G; Study on channel model for frequencies from 0.5 to 100 GHz (3GPP TR 38.901 v14.3.0 Release 14). Technical report, ETSI, 2018.

[54] C. Evci and B. Fino. Spectrum management, pricing, and efficiency control in broadband wireless communications. *Proceedings of the IEEE*, 2001.

[55] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 1998.

[56] Z. Feng, S. Jaewon, and J. Reich. Information-driven dynamic sensor collaboration. *IEEE Signal Processing Magazine*, 2002.

[57] Z. Feng, L. Jie, L. Juan, L. Guibas, and J. Reich. Collaborative signal and information processing: an information-directed approach. *Proceedings of the IEEE*, 2003.

[58] C. Fortuin, P. Kasteleyn, and J. Ginibre. Correlation inequalities on some partially ordered sets. *Communications in Mathematical Physics*, 1971.

[59] X. Foukas, G. Patounas, A. Elmokashfi, and M. Marina. Network Slicing in 5G: Survey and Challenges. *IEEE Communications Magazine*, 2017.

[60] C. Fraleigh, F. Tobagi, and C. Diot. Provisioning IP backbone networks to support latency sensitive traffic. In *IEEE INFOCOM 2003*, 2003.

[61] A. Fréville. The multidimensional 0–1 knapsack problem: An overview. *European Journal of Operational Research*, 2004.

[62] Y. Gao, L. Wang, and J. Zhou. Cost-Efficient and Quality of Experience-Aware Provisioning of Virtual Machines for Multiplayer Cloud Gaming in Geographically Distributed Data Centers. *IEEE Access*, 2019.

[63] M. 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*, 2021.

[64] German Aerospace Center (DLR). Simulation of Urban MObility. *[Online] https://sumo.dlr.de/docs/, Accessed on 2020-03-01.*, 2020.

[65] Google. Google Stadia. *[Online] https://stadia.google.com/, Accessed on 2020-11-15.*, 2020.

[66] M. Gramaglia, P. Serrano, J. Hernández, M. Calderon, and C. Bernardos. New insights from the analysis of free flow vehicular traffic in highways. In *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2011.

[67] B. Hajek. Balanced loads in infinite networks. *Annals of Applied Probability*, 1996.

[68] G. Hardy, J. Littlewood, and G. Polya. Some simple inequalities satisfied by convex functions. *Messenger of Mathematics*, 1929.

[69] H. Hashemipour, S. Roy, and A. Laub. Decentralized structures for parallel Kalman filtering. *IEEE Transactions on Automatic Control*, 1988.

[70] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal. NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC). *IEEE Network*, 2014.

[71] H. Hong, D. Chen, C. Huang, K. Chen, and C. Hsu. Placing Virtual Machines to Optimize Cloud Gaming Experience. *IEEE Transactions on Cloud Computing*, 2015.

[72] K. Hong, J. Kenney, V. Rai, and K. Laberteaux. Evaluation of multi-channel schemes for vehicular safety communications. In *2010 IEEE 71st Vehicular Technology Conference*, 2010.

[73] W. Hongyi, Q. Chunming, S. De, and O. Tonguz. Integrated cellular and ad hoc relaying systems: iCAR. *IEEE Journal on Selected Areas in Communications*, 2001.

[74] C. Huang, Y. Fallah, R. Sengupta, and H. Krishnan. Adaptive inter-vehicle communication control for cooperative safety systems. *IEEE Network*, 2010.

[75] L. Huang and E. Modiano. Optimizing age-of-information in a multi-class queueing system. In *2015 IEEE International Symposium on Information Theory (ISIT)*, 2015.

[76] R. Hussain and S. Zeadally. Autonomous Cars: Research Results, Issues, and Future Challenges. *IEEE Communications Surveys Tutorials*, 2019.

[77] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hoßfeld. An Evaluation of QoE in Cloud Gaming Based on Subjective Tests. In *2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2011.

[78] S. Jawaid and S. Smith. On the submodularity of sensor scheduling for estimation of linear dynamical systems. *2014 American Control Conference*, 2014.

[79] N. Jindal, J. Andrews, and S. Weber. Optimizing the SINR operating point of spatial networks. *arXiv preprint cs/0702030*, 2007.

[80] C. Karakus and S. Diggavi. Enhancing Multiuser MIMO Through Opportunistic D2D Cooperation. *IEEE Transactions on Wireless Communications*, 2017.

[81] A. Karamoozian, A. Hafid, and E. Aboulhamid. On the Fog-Cloud Cooperation: How Fog Computing can address latency concerns of IoT applications. In *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, 2019.

[82] S. Kasiviswanathan, S. Eidenbenz, and G. Yan. Geography-based analysis of the internet infrastructure. In *IEEE INFOCOM 2011*, 2011.

[83] S. Kassir, P. Caballero Garces, G. de Veciana, N. Wang, X. Wang, and P. Palacharla. An Analytical Model and Performance Evaluation of Multihomed Multilane VANETs. *IEEE/ACM Transactions on Networking*, 2021.

[84] S. Kassir, G. de Veciana, N. Wang, X. Wang, and P. Palacharla. Enhancing Cellular Performance via Vehicular-based Opportunistic Relaying and Load Balancing. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, 2019.

[85] S. Kassir, G. de Veciana, N. Wang, X. Wang, and P. Palacharla. Service Placement for Real-Time Applications: Rate-Adaptation and Load-Balancing at the Network Edge. In *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 2020.

[86] S. Kassir, G. de Veciana, N. Wang, X. Wang, and P. Palacharla. Joint Update Rate Adaptation in Multiplayer Cloud-Edge Gaming Services:

Spatial Geometry and Performance Tradeoffs. In *Proceedings of the Twenty-Second International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, MobiHoc '21, 2021.

[87] S. Kaul, M. Gruteser, V. Rai, and J. Kenney. Minimizing age of information in vehicular networks. *8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2011.

[88] S. Kaul, R. Yates, and M. Gruteser. Real-time status: How often should one update? In *IEEE INFOCOM 2012*, 2012.

[89] J. Kenney. Dedicated Short-Range Communications (DSRC) Standards in the United States. *Proceedings of the IEEE*, July 2011.

[90] Z. Khan, A. Vasilakos, B. Barua, S. Shahabuddin, and H. Ahmadi. Cooperative content delivery exploiting multiple wireless interfaces: methods, new technological developments, open research issues and a case study. *Wireless networks*, 2016.

[91] H. Kim, G. de Veciana, X. Yang, and M. Venkatachalam. Distributed α -Optimal User Association and Cell Load Balancing in Wireless Networks. *IEEE/ACM Transactions on Networking*, 2012.

[92] L. Kleinrock. *Theory, Volume 1, Queueing Systems*. Wiley-Interscience, 1975.

[93] A. Kosta, N. Pappas, and V. Angelakis. Age of information: A new concept, metric, and tool. *Foundations and Trends in Networking*, 2017.

[94] S. Kwon, Y. Kim, and N. Shroff. Analysis of Connectivity and Capacity in 1-D Vehicle-to-Vehicle Networks. *IEEE Transactions on Wireless Communications*, 2016.

[95] T. Lan, D. Kao, M. Chiang, and A. Sabharwal. An Axiomatic Theory of Fairness in Network Resource Allocation. In *IEEE INFOCOM 2010*, 2010.

[96] S. LaValle, A. Yershova, M. Katsev, and M. Antonov. Head tracking for the Oculus Rift. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[97] L. Le and E. Hossain. Multihop Cellular Networks: Potential Gains, Research Challenges, and a Resource Allocation Framework. *IEEE Communications Magazine*, 2007.

[98] J. Lee and J. H. Lee. Performance Analysis and Resource Allocation for Cooperative D2D Communication in Cellular Networks With Multiple D2D Pairs. *IEEE Communications Letters*, 2019.

[99] F. Li and Y. Wang. Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine*, June 2007.

[100] Z. Li, C. Wang, and C. Jiang. User Association for Load Balancing in Vehicular Networks: An Online Reinforcement Learning Approach. *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[101] Y. Lin and H. Shen. CloudFog: Leveraging Fog to Extend Cloud Gaming for Thin-Client MMOG with High Quality of Service. *IEEE Transactions on Parallel and Distributed Systems*, 2017.

[102] J. Liu, Y. Kawamoto, H. Nishiyama, N. Kato, and N. Kadowaki. Device-to-device communications achieve efficient load balancing in LTE-advanced networks. *IEEE Transactions on Wireless Communications*, 2014.

[103] Y. Liu, J. Ma, J. Niu, Y. Zhang, and W. Wang. Roadside units deployment for content downloading in vehicular networks. In *2013 IEEE International Conference on Communications (ICC)*, 2013.

[104] T. Losev, S. Storteboom, S. Carpendale, and S. Knudsen. Distributed Synchronous Visualization Design: Challenges and Strategies. In *2020 IEEE Workshop on Evaluation and Beyond - Methodological Approaches to Visualization (BELIV)*, 2020.

[105] P. Luoto, M. Bennis, P. Pirinen, S. Samarakoon, K. Horneman, and M. Latva-aho. Vehicle clustering for improving enhanced LTE-V2X network performance. In *2017 European Conference on Networks and Communications (EuCNC)*, 2017.

[106] P. Mach and Z. Becvar. Mobile Edge Computing: A Survey on Architecture and Computation Offloading. *IEEE Communications Surveys Tutorials*, 2017.

[107] P. Mach, T. Spyropoulos, and Z. Becvar. Incentive-Based D2D Relaying in Cellular Networks. *IEEE Transactions on Communications*, 2021.

[108] A. Maia, Y. Ghamri-Doudane, D. Vieira, and M. de Castro. Optimized Placement of Scalable IoT Services in Edge Computing. In *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2019.

[109] S. Mangiante, G. Klas, A. Navon, Z. GuanHua, J. Ran, and M. Silva. VR is on the Edge: How to Deliver 360° Videos in Mobile Networks. In *ACM SIGCOMM 2017*, 2017.

[110] G. Mao and B. Anderson. Graph Theoretic Models and Tools for the Analysis of Dynamic Wireless Multihop Networks. In *2009 IEEE Wireless Communications and Networking Conference*, April 2009.

[111] Y. Mao, C. You, J. Zhang, K. Huang, and K. Letaief. A Survey on Mobile Edge Computing: The Communication Perspective. *IEEE Communications Surveys Tutorials*, 2017.

[112] A. Marshall, I. Olkin, and B. Arnold. *Inequalities: theory of majorization and its applications*. Springer, 1979.

[113] X. Masip-Bruin, E. Marín-Tordera, G. Tashakor, A. Jukan, and G. Ren. Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems. *IEEE Wireless Communications*, 2016.

[114] Microsoft. Project xCloud. *[Online] https://www.xbox.com/en-US/xbox-game-pass/cloud-gaming/home, Accessed on 2020-11-15.*, 2020.

[115] N. Mohammadi and J. Taylor. Smart city digital twins. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017.

[116] A. Müller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks*. Wiley Series in Probability and Statistics. Wiley, 2002.

[117] NGMN Alliance. V2X white paper. 2018.

[118] Y. Niu, Y. Li, D. Jin, L. Su, and A. Vasilakos. A survey of millimeter wave communications (mmWave) for 5G: opportunities and challenges. *Wireless Networks*, 2015.

[119] D. Niyato and E. Hossain. A Unified Framework for Optimal Wireless Access for Data Streaming Over Vehicle-to-Roadside Communications. *IEEE Transactions on Vehicular Technology*, 2010.

[120] Nvidia. Nvidia GeForce. *[Online] https://www.nvidia.com/en-us/geforce-now/, Accessed on 2020-11-15.*, 2020.

[121] R. Olfati-Saber. Distributed Kalman filtering for sensor networks. *46th IEEE Conference on Decision and Control*, 2007.

[122] A. Osseiran, S. Parkvall, P. Persson, A. Zaidi, S. Magnusson, and K. Balachandran. 5G wireless access: an overview. *White Paper*, April 2010.

[123] B. Pan and H. Wu. Performance Analysis of Connectivity Considering User Behavior in V2V and V2I Communication Systems. In *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017.

[124] S. Pasteris, S. Wang, M. Herbster, and T. He. Service Placement with Provable Guarantees in Heterogeneous Edge Computing Systems. In *IEEE INFOCOM 2019*, 2019.

[125] P. Pathak, X. Feng, P. Hu, and P. Mohapatra. Visible light communication, networking, and sensing: A survey, potential and challenges. *IEEE Communications Surveys Tutorials*, 2015.

[126] M. Petovello, K. O'Keefe, G. Lachapelle, and E. Cannon. Consideration of time-correlated errors in a Kalman filter applicable to GNSS. *Journal of Geodesy*, 2009.

[127] Z. Pi, J. Choi, and R. Heath. Millimeter-wave gigabit broadband evolution toward 5G: fixed access and backhaul. *IEEE Communications Magazine*, 2016.

[128] H. Qi, Y. Xu, and X. Wang. Mobile-agent-based collaborative signal and information processing in sensor networks. *Proceedings of the IEEE*, 2003.

[129] B. Radunovic and J. Le Boudec. A Unified Framework for Max-Min and Min-Max Fairness With Applications. *IEEE/ACM Transactions on Networking*, 2007.

[130] R. Ramaswamy, N. Weng, and T. Wolf. Characterizing network processing delay. In *Globecom '04.*, 2004.

[131] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y.aniv Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez. Millimeter Wave Mobile Communications for 5G Cellular: It Will Work! *IEEE Access*, 2013.

[132] T. Rappaport, Y. Xing, G. MacCartney, A. Molisch, E. Mellios, and J. Zhang. Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless Networks—With a Focus on Propagation Models. *IEEE Transactions on Antennas and Propagation*, 2017.

[133] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh. NB-IoT system for M2M communication. In *IEEE Wireless Communications and Networking Conference*, 2016.

[134] G. Rawat and K. Singh. Joint beacon frequency and beacon transmission power adaptation for internet of vehicles. *Transactions on Emerging*

Telecommunications Technologies, 2020.

[135] E. Reingold, J. Nievergelt, and N. Deo. *Combinatorial Algorithms: Theory and Practice*. Prentice Hall College Div, 1977.

[136] A. Reis, S. Sargento, F. Neves, and O. Tonguz. Deploying Roadside Units in Sparse Vehicular Networks: What Really Works and What Does Not. *IEEE Transactions on Vehicular Technology*, 2014.

[137] C. Saha and H. Dhillon. D2D Underlaid Cellular Networks with User Clusters: Load Balancing and Downlink Rate Analysis. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017.

[138] A. Samuylov, M. Gapeyenko, D. Moltchanov, M. Gerasimenko, S. Singh, N. Himayat, S. Andreev, and Y. Koucheryavy. Characterizing Spatial Correlation of Blockage Statistics in Urban mmWave Systems. In *2016 IEEE Globecom Workshops (GC Wkshps)*, 2016.

[139] A. Sang, X. Wang, M. Madihian, and R. Gitlin. A load-aware handoff and cell-site selection scheme in multi-cell packet data systems. In *IEEE Global Telecommunications Conference, 2004. GLOBECOM '04.*, 2004.

[140] A. Santoyo González and C. Cervelló Pastor. Edge Computing Node Placement in 5G Networks: A Latency and Reliability Constrained Framework. In *2019 6th IEEE International Conference on CSCloud/ 2019 5th IEEE EdgeCom*, 2019.

[141] M. Shaked and G. Shanthikumar. Supermodular stochastic orders and positive dependence of random vectors. *Journal of Multivariate Analysis*, 1997.

[142] M. Shaked and G. Shanthikumar. *Stochastic orders*. Springer Science & Business Media, 2007.

[143] M. Shamaiah, S. Banerjee, and H. Vikalo. Greedy sensor selection: Leveraging submodularity. *49th IEEE Conference on Decision and Control (CDC)*, 2010.

[144] C. Shao, S. Leng, Y. Zhang, A. Vinel, and M. Jonsson. Analysis of connectivity probability in platoon-based Vehicular Ad Hoc Networks. In *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2014.

[145] C. Shao, S. Leng, Y. Zhang, A. Vinel, and M. Jonsson. Performance Analysis of Connectivity Probability and Connectivity-Aware MAC Protocol Design for Platoon-Based VANETs. *IEEE Transactions on Vehicular Technology*, 2015.

[146] A. Sharma, A. Trivedi, and N. Roberts. Efficient Load Balancing Using D2D Communication and Biasing in LTE-Advance Het-Nets. In *Proceedings of the Sixth ACM International Conference on Computer and Communication Technology 2015*, 2015.

[147] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal*, 2016.

[148] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner. Optimized IoT service placement in the fog. *Service Oriented Computing and Applications*, 2017.

[149] B. Sliwa, R. Falkenberg, and C. Wietfeld. A Simple Scheme for Distributed Passive Load Balancing in Mobile Ad-hoc Networks. *arXiv preprint arXiv:1702.05235*, 2017.

[150] K. Son, S. Chong, and G. de Veciana. Dynamic association for load balancing and interference avoidance in multi-cell networks. *IEEE Transactions on Wireless Communications*, 2009.

[151] Sony. PlayStation Now. *[Online] https://www.playstation.com/en-us/explore/playstation-now/, Accessed on 2020-09-15.*, 2020.

[152] B. Soret, P. Mogensen, K. Pedersen, and M. Aguayo-Torres. Fundamental tradeoffs among reliability, latency and throughput in cellular networks. In *Globecom 2014 Workshops*, 2014.

[153] S. Sou and O. Tonguz. Enhancing VANET Connectivity Through Roadside Units on Highways. *IEEE Transactions on Vehicular Technology*, 2011.

[154] M. Stecklein, H. Beytur, G. de Veciana, and H. Vikalo. Optimizing Resource Constrained Distributed Collaborative Sensing. *IEEE International Conference on Communications Workshops*, 2021.

[155] C. Storck and F. Duarte-Figueiredo. A survey of 5G technology evolution, standards, and infrastructure associated with vehicle-to-everything communications by internet of vehicles. *IEEE Access*, 2020.

[156] X. Tian, T. Yuan, and Y. Bar-Shalom. Track-to-track fusion in linear and nonlinear systems. *Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control*, 2012.

[157] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein. Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information. *IEEE Transactions on Vehicular Technology*, 2009.

[158] V. Tzoumas, A. Jadbabaie, and G. Pappas. Sensor placement for optimal Kalman filtering: Fundamental limits, submodularity, and algorithms. *2016 American Control Conference (ACC)*, 2016.

[159] M. Uysal, Z. Ghassemlooy, A. Bekkali, A. Kadri, and H. Menouar. Visible Light Communication for Vehicular Networking: Performance Study of a V2V System Using a Measured Headlamp Beam Pattern Model. *IEEE Vehicular Technology Magazine*, 2015.

[160] T. Van and N. Location-aware and load-balanced data delivery at roadside units in vehicular Ad hoc networks. In *IEEE International Sympo-*

sium on Consumer Electronics (ISCE 2010), 2010.

[161] D. Viegas, P. Batista, P. Oliveira, and C. Silvestre. Discrete-time distributed Kalman filter design for formations of autonomous vehicles. *Control Engineering Practice*, 2018.

[162] L. Vigneri, G. Paschos, and P. Mertikopoulos. Large-Scale Network Utility Maximization: Countering Exponential Growth with Exponentiated Gradients. In *IEEE INFOCOM 2019*, 2019.

[163] The Void. The Void. *[Online] https://www.thevoid.com/, Accessed on 2020-11-15.*, 2020.

[164] Y. Wang and H. Wei. Road Capacity and Throughput for Safe Driving Autonomous Vehicles. *IEEE Access*, 2020.

[165] E. Weisstein. Circle-circle intersection. *[Online] MathWorld – http:// mathworld.wolfram.com /Circle-CircleIntersection.html, Accessed on 2022-02-21.*, 2021.

[166] N. Wisitpongphan, F. Bai, P. Mudalige, V. Sadekar, and O. Tonguz. Routing in Sparse Vehicular Ad Hoc Wireless Networks. *IEEE Journal on Selected Areas in Communications*, October 2007.

[167] D. Wu, J. Luo, R. Li, and A. Regan. Geographic load balancing routing in hybrid Vehicular Ad Hoc Networks. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2011.

[168] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. Wong, and X. Gao. A Survey of Physical Layer Security Techniques for 5G Wireless Networks and Challenges Ahead. *IEEE Journal on Selected Areas in Communications*, 2018.

[169] L. Yang, J. Cao, G. Liang, and X. Han. Cost Aware Service Placement and Load Dispatching in Mobile Cloud Systems. *IEEE Transactions on Computers*, 2016.

[170] S. Yang, F. Li, M. Shen, X. Chen, X. Fu, and Y. Wang. Cloudlet Placement and Task Allocation in Mobile Edge Computing. *IEEE Internet of Things Journal*, 2019.

[171] Y. Yang, Z. Mi, J. Yang, and G. Liu. A Model Based Connectivity Improvement Strategy for Vehicular Ad hoc Networks. In *2010 IEEE 72nd Vehicular Technology Conference - Fall*, 2010.

[172] R. Yates and S. Kaul. The Age of Information: Real-Time Status Updating by Multiple Sources. *IEEE Transactions on Information Theory*, 2019.

[173] R. Yates, M. Tavan, Y. Hu, and D. Raychaudhuri. Timely cloud gaming. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017.

[174] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. Andrews. User Association for Load Balancing in Heterogeneous Cellular

Networks. *IEEE Transactions on Wireless Communications*, 2013.

[175] J. Yoo, B. Sung C. Choi, and M. Gerla. An opportunistic relay protocol for vehicular road-side access with fading channels. In *The 18th IEEE International Conference on Network Protocols*, 2010.

[176] F. Yousaf, M. Bredel, S. Schaller, and F. Schneider. NFV and SDN—Key Technology Enablers for 5G Networks. *IEEE Journal on Selected Areas in Communications*, 2017.

[177] C. Yu and O. Tirkkonen. Opportunistic multiple relay selection with diverse mean channel gains. *IEEE Transactions on Wireless Communications*, 2012.

[178] R. Yu, G. Xue, and X. Zhang. Application Provisioning in FOG Computing-enabled Internet-of-Things: A Network Perspective. In *IEEE INFOCOM 2018*, 2018.

[179] F. Zabini, A. Bazzi, B. Masini, and R. Verdone. Optimal Performance Versus Fairness Tradeoff for Resource Allocation in Wireless Systems. *IEEE Transactions on Wireless Communications*, 2017.

[180] H. Zhang, L. Song, and Y. Zhang. Load Balancing for 5G Ultra-Dense Networks Using Device-to-Device Communications. *IEEE Transactions on Wireless Communications*, 2018.

[181] L. Zhang and S. Valaee. Safety context-aware congestion control for vehicular broadcast networks. *IEEE 15th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2014.

[182] J. Zhao, Y. Chen, and Y. Gong. Study of Connectivity Probability of Vehicle-to-Vehicle and Vehicle-to-Infrastructure Communication Systems. In *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016.

[183] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou. Heterogeneous Vehicular Networking: A Survey on Architecture, Challenges, and Solutions. *IEEE Communications Surveys Tutorials*, 2015.

[184] S. Zhong, J. Chen, and Y. Yang. Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *IEEE INFOCOM 2003*, 2003.

[185] A. Zhou, S. Wang, B. Cheng, Z. Zheng, F. Yang, R. Chang, M. Lyu, and R. Buyya. Cloud Service Reliability Enhancement via Virtual Machine Placement Optimization. *IEEE Transactions on Services Computing*, 2017.

[186] J. Zhou, G. Gu, and X. Chen. Distributed Kalman Filtering Over Wireless Sensor Networks in the Presence of Data Packet Drops. *IEEE Transactions on Automatic Control*, 2019.

[187] Y. Zhu, Z. You, J. Zhao, K. Zhang, and R. Li. The optimality for the distributed Kalman filtering fusion with feedback. *Automatica*, 2001.

Vita

Saadallah Kassir earned his Scientific Baccalaureate in July 2013 from the Lycée La Favorite Sainte-Thérèse in Lyon, France. He then enrolled at the American University of Beirut, Lebanon, from where he graduated with a B.Sc. in Computer and Communication Engineering with High-Distinction in May 2017. In August 2017, he joined the Wireless Networking and Communications Group (WNCG) at the University of Texas at Austin, TX, USA, to pursue his M.Sc., earned in December 2019, and Ph.D. degrees in Electrical and Computer Engineering under the supervision of Prof. Gustavo de Veciana. During his graduate studies, he performed summer internships with Fujitsu Networks Communication as a Strategic Planning Intern in the summer of 2018 and with Qualcomm as a Wireless R&D Intern in the summers of 2019, 2020 and 2021. His main research interests include design and performance analysis of wireless networks, applied in particular to vehicular and cloud/edge computing networks.

Permanent address: skassir@utexas.edu

This dissertation was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.